Kepler Codebook

Junrong Lian [*1] **Ziyue Dong** [*2] **Pengxu Wei** [1] **Wei Ke** [2] **Chang Liu** [3] **Qixiang Ye** [4] **Xiangyang Ji** [3] **Liang Lin** [1 5]

Abstract

A codebook designed for learning discrete distributions in latent space has demonstrated state-of-the-art results on generation tasks. This inspires us to explore what distribution of codebook is better. Following the spirit of Kepler's Conjecture, we cast the codebook training as solving the sphere packing problem and derive a Kepler codebook with a compact and structured distribution to obtain a codebook for image representations. Furthermore, we implement the Kepler codebook training by simply employing this derived distribution as regularization and using the codebook partition method. We conduct extensive experiments to evaluate our trained codebook for image reconstruction and generation on natural and human face datasets, respectively, achieving significant performance improvement. Besides, our Kepler codebook has demonstrated superior performance when evaluated across datasets and even for reconstructing images with different resolutions. Codes and pre-trained weights are available at https://github.com/banianrong/KeplerCodebook.

1. Introduction

Vector quantization (VQ) (Gray, 1984) is a foundational algorithm in the field of machine learning, extensively utilized in deep learning for various domains including audio (Baevski et al., 2019; Wang et al., 2021; Wu et al., 2020), language (Roy & Grangier, 2019; Chen et al., 2023) and vision tasks (Van Den Oord et al., 2017; Razavi et al., 2019; Esser et al., 2021). Among these, its application in image generation/synthesis has been particularly notable in recent years, especially with the prevalence of pre-quantizing images into discrete latent variables and modeling them autoregressively, *e.g.*, VQVAE (Van Den Oord et al., 2017), DALL-E (Ramesh et al., 2021a), VQGAN (Esser et al., 2021), and ViT-VQGAN (Yu et al., 2021). Those approaches follow a two-stage generation routine, including a codebook learning by image quantization for image reconstruction in the first stage and vector-quantized image modeling based on the learned codebook for image generation in the second stage.

Nevertheless, codebook learning always bears the brunt of codebook collapse (Yu et al., 2021; Zhang et al., 2023; Ramesh et al., 2021a), indicating that a large portion of tokens in a learned codebook have not been fully used with a rather low codebook usage[1] (Yu et al., 2021; Zhang et al., 2023), *e.g.*, *35.9%* for VQGAN, shown in Fig. 2. This raises an issue: *for a learned codebook, its low codebook usage is bad for image reconstruction.* Subsequently, several methods have been proposed to address this issue. One effective method is the application of Gumbel-Softmax (Jang et al., 2016; Ramesh et al., 2021b), which employs stochastic quantization by random sampling to select a token from a predicted token distribution. Reg-VQ (Zhang et al., 2023) uses a stochastic mask regularization to balance VQGAN and Gumbel-VQ quantization method. However, this apparent improvement in codebook usage is affected by stochastic quantization, which essentially leads to unreliable training with limited quality of reconstructed images and generalization of image representation. This raises another issue: *higher codebook usage does not promise an excellent reconstruction capability.* For instance, in Fig. 2, Reg-VQ has a higher codebook usage, but its active frequency variance[2], is

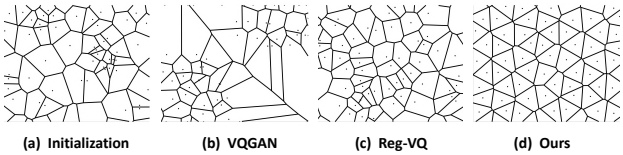

Figure 1. Codebook distribution. Tokens are denoted by dots. The codebook is initialized with uniform distribution (a). After training, VAGAN and Reg-VQ maintain unordered in (b) and (c), impairing their performance in reconstructing and generating images. Our Kepler codebook with an ordered and compact distribution (d).

[*]Equal contribution [1]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China [2]School of Software Engineering, Xi'an Jiaotong University, Xi'an, China [3]Department of Automation, Tsinghua University, Beijing, China [4]School of Electronic, Electrical and Communication Engineering, University of Chinese Academy of Sciences, Beijing, China [5]Peng Cheng Laboratory, Shenzhen, China. Correspondence to: Pengxu Wei <weipx3@mail.sysu.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

1. *"Codebook usage"* means the percentage of how many tokens in a codebook are used for image reconstruction.

2. *Active frequency variance* measures the difference indicator

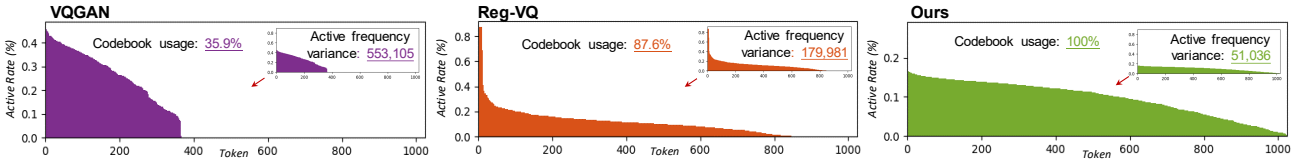

Figure 2. The codebook usage, active frequency variance, and active rate statistics of tokens for different models trained on ADE20K. Those models are trained with the same epochs for statistic comparison. VQGAN focuses on a very limited number of tokens that have been trained many times with a high active rate, and thus it has a low codebook usage (35.9%) and high active frequency variance. Even though Reg-VQ has a high codebook usage (87.6%) and declines the active frequency variance, it still exhibits a training bias to some tokens trained many times, but some tokens have not trained. Instead, our Kepler codebook tends to a balanced training for each token; thus it significantly improves the codebook usage to 100% and remarkably declines the active frequency variance.

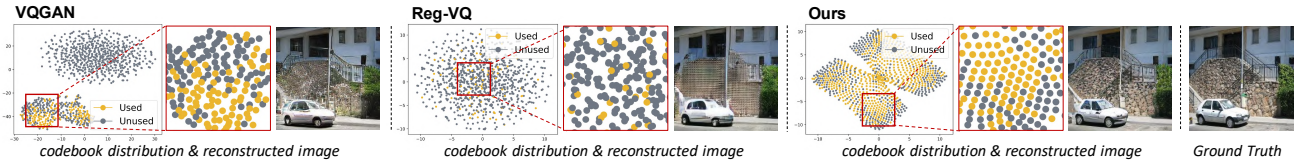

Figure 3. The active tokens are highlighted for image reconstruction in the whole token embedding space (t-SNE). One dot denotes one token in a trained codebook. Take one image for example: for reconstructing an image in a street scene, active tokens are in yellow; not-used (*i.e.*, dead) tokens are in gray. VQGAN and Reg-VQ utilize a rather smaller part of tokens than our method to reconstruct images. Thus, they possibly fail to exploit the VQ representations to produce more image details effectively, and thus the quality of their reconstructed images is negatively influenced.

also large, indicating that their tokens are not well-balanced for training. Thus, a small part of tokens are used to reconstruct images, and its reconstructed images present limited texture details and even obvious artifacts, as shown in Fig. 3. This possibly causes a limited codebook generalization, extending it to other datasets.

The above problems are relevant to the distribution of the codebook, as shown in Fig. 1. In VQGAN, during early training, only the tokens closest to the feature will be activated. As the update progresses, this small subset of tokens gradually drifts from the center of the initial distribution, leading to *low usage*, distorted structure, and many wrong details. Reg-VQ regularizes the training process with a uniform distribution to make all tokens be used uniformly. Despite most tokens being activated, Reg-VQ still favors certain tokens to an extreme, potentially leading to *high active frequency variance*, obvious artifacts (*e.g.*, duplicate wall textures), and limited codebook generalization. In Fig. 3, the codebook distributions of both VQGAN and Reg-VQ show a disordered pattern. This implies that certain tokens used in almost every reconstruction may represent multiple different features, leading to artifacts and inaccurate details in reconstruction.

To address the aforementioned issues, we propose developing a compact and structured codebook for improving discrete representation. This approach draws inspiration from Kepler's Conjecture, suggesting that codebook training can be likened to solving a sphere packing problem. Building on two key preconditions, we argue that the compact and structured distribution can be effectively modeled by the Irwin-Hall distribution (Hall, 1927). This is a probability distribution for a random variable defined as the sum of many independent random variables, each having a uniform distribution. Using Irwin-Hall distribution as the ideal prior, we apply it to regularize the codebook's posterior distribution via KL divergence. Based on the preconditions, we further argue that the distribution of each dimension within the codebook follows the independent and identically distributed (i.i.d.) distribution. Consequently, we group the encoder tokens, called codebook partition, to simplify the complex dense distribution. This allows the codebook to better capture the distribution, thus improving the ability of reconstruction and generalization. Moreover, we conduct reconstruction and generation experiments on the ADE20K and CelebA-HQ datasets and obtain superior performance. We additionally perform cross-domain experiments on three other datasets to validate our model's generalization and downstream super-resolution based on the latent diffusion model on DRealSR dataset.

In a nutshell, our contributions are summarized below:

- Following the spirit of Kepler's Conjecture, we propose to combine codebook training with solving the sphere packing problem. We introduce the Kepler codebook, which features a compact and structured distribution, to achieve enhanced discrete representation.

of *"Active frequency"* that denotes how many times one token has been trained during codebook training. *"Active rate"* is the percentage of active frequency, indicating whether codebook tokens have been well trained.

- We employ the derived Irwin-Hall distribution to regularize the codebook optimization process and propose a codebook partitioning method to reduce the codebook distribution's complexity. These two strategies effectively restructure the codebook distribution, improving the codebook usage and decreasing the active frequency variance with balanced codebook training.
- Comprehensive experiments have demonstrated the superiority of our method in reconstruction, generation, cross-domain reconstruction, and downstream super-resolution tasks.

2. Related Work

2.1. Tokenized Image Synthesis

Many prevailing approaches for learning discrete representations employ VQ-based methodologies, typically following a two-step training procedure. The first step trains a well-structured codebook, considered as a discrete representation. In the second step, networks are trained to predict token indices to approximate this discrete space. VQ-VAE (Van Den Oord et al., 2017) initially demonstrated strong generation capabilities with PixelCNN (Van Den Oord et al., 2016), while VQGAN (Esser et al., 2021) later excelled at synthesizing high-resolution images using auto-regressive transformers. ViT-VQGAN (Yu et al., 2021) improved the tokenization phase by introducing a VIT-based (Dosovitskiy et al., 2020) encoder-decoder setup. RQ-VAE (Lee et al., 2022) made the code sequence more manageable by encoding images as discrete stack sequences. DQ-VAE (Huang et al., 2023) generated images progressively by assigning varying code lengths to different parts of the image. HQ-VAE (You et al., 2022) used a two-tiered discrete coding approach with differing spatial resolutions. In contrast to adding complexity to network architectures, our work concentrates on refining the codebook itself. We aim to heighten reconstruction quality by optimizing codebook usage and attaining a more condensed distribution.

2.2. Codebook Usage Optimization

There have been several methods to improve codebook usage by various ideas. VQGAN (Van Den Oord et al., 2017; Esser et al., 2021) with narrow interval codebook initialization and without regularization often leaves many tokens untrained throughout the training process, resulting in their usage falling below 40%. In ViT-VQGAN (Yu et al., 2021), factorized codes and L2-norm are used to enhance codebook usage. Reg-VQ (Zhang et al., 2023)combines deterministic and stochastic quantization to activate tokens via Gumbel sampling. Additionally, HVQ-VAE (Williams et al., 2020) and Jukebox (Dhariwal et al., 2020) implement codebook reset strategies, randomly re-initializing unused or low-used codebook tokens. Building upon the concept, CVQ-VAE (Zheng & Vedaldi, 2023) further refines the approach by clustering anchors online to unoptimized tokens, thereby waking up inactive tokens. However, these methods do not answer what a good codebook distribution looks like. Following the spirit of Kepler's Conjecture, we propose a solution involving a compact and ordered distribution to tackle these challenges effectively.

3. Kepler Codebook

As outlined in Sec. 1, two primary concerns on codebook training pose significant challenges: ***1)*** *a learned codebook often exhibits low codebook usage, which is detrimental to image reconstruction;* ***2)*** *even a higher codebook usage does not necessarily guarantee superior reconstruction capability.* Those two issues would result in low-quality image reconstruction and limited codebook generalization. Thus, it is essential to investigate the characteristics of an effective codebook and the optimal methods to train such a codebook. In our study, we attempt to address this question by deriving the structure and distribution of the codebook for its training.

3.1. Codebook Training is Kepler's Conjecture

Learning discrete representations is closely related to a codebook for vector quantization. Assuming the representation space is bounded, ***1)*** *a good codebook with N tokens is expected, whose space spanned by all tokens is as large as possible;* ***2)*** *the distance between each token is relatively far apart, resulting in a relatively balanced probability of each token being trained.*

With both preconditions, we will derive the distribution or structure of the codebook. In particular, to ensure as identical training as possible for each token, it can be cast as a problem of codebook token packing, inspired by Kepler's Conjecture (Kepler, 1966), which was proposed to exploit the problem of the sphere packing problem. Thus, we specifically establish our strategy of codebook training, following two principles of Kepler's Conjecture (Hales, 1998).

Formally, a codebook is denoted as $\mathcal{Z} = \{z_k\}_{k=1}^{K} \subset \mathbb{R}^{n_z}$, consisting of K tokens in n_z dimensions. For an image $x \in \mathbb{R}^{H \times W \times 3}$, it is represented by a set of codebook entries $z_{\boldsymbol{q}} \in \mathbb{R}^{h \times w \times n_z}$, where $(h, w) = (H/f, W/f)$, and f is the down-sampling factor. In line with the VQGAN model, the codebook is learned via a convolutional model comprised of an encoder E and a decoder D. During training, a given image x is approximated into $\hat{x} = D(z_{\boldsymbol{q}})$ for image reconstruction. This process is subject to two preconditions aforementioned, which are elaborated as follows.

1) Making the space spanned by all the tokens as **large**

as possible, indicating that a token $z_{\boldsymbol{q}} \in \mathbb{R}^{h \times w \times n_z}$ in the codebook should cover continuous latent space $\hat{z}_{\boldsymbol{q}} = E(x) \in \mathbb{R}^{h \times w \times n_z}$ as extensively as possible. $|\hat{z}|$ is the size of the continuous latent space $\hat{z}_{\boldsymbol{q}}$.

2) Making the distance between each token in the codebook relatively **far**, indicating maximizing the minimum of d_i, *i.e.*, $\max_d \min_i d_i$. d_i is the minimum distance between the i-th token and all the other tokens.

Lemma 1. *Considering that a token vector z_k is in the rational number field, according to the countability of rational numbers (Sagan, 1991), there exists a set of basis vector (a.k.a., basis matrix) B that satisfies $z_k = Bm$, where the n_z-dimensional vector m denotes integral coefficients.*

Based on Lemma 1, we reformulate the codebook $\mathcal{Z}$ by a basis matrix $B = \{b_k\}_{k=1}^{n_z} \subset \mathbb{R}^{n_z}$, where b_k is a basis vector. That is, **each token vector can be represented by this basis matrix** B. The spanned space of z_k can be regarded as an n_z-dimensional sphere with the radius r_k and then its volume is in direct proportion to ${r_k}^{n_z}$ (its coefficient is constant for all the tokens and thus is ignored for simplicity in the following sections). The space of each token has no overlap. Accordingly, the whole volume of the spanned space by all the tokens is constrained to $|\hat{z}|$, namely, $\sum_{i=1}^{K} {r_i}^{n_z} \leq |\hat{z}|$. Then, based on the two preconditions, our objective of codebook training can be formally formulated as follows,

$$\begin{cases} arg \max_d \min_i d_i, \\ s.t. \ \sum_{i=1}^{K} {r_i}^{n_z} \leq |\hat{z}|, \end{cases} \tag{1}$$

Remark 1. *In Equ. 1, the objective optimization for codebook training essentially follows the spirit of Kepler's Conjecture (Kepler, 1966), which indicates a problem of the closest packing of spheres to achieve the maximum packing density of spheres in n_z-dimensional space.*

Remark 2. *With the Lagrange multiplier technique, the optimal objective is achieved when $d_1 = d_2 = ... = d_K$ under the condition of the spheres of two adjacent tokens being tangent. The detailed proof is given in Appendix A.1.*

In this fashion, an optimally trained codebook exhibits a tight and structured distribution. Compared to the loose disorder of merely narrowing the upper and lower bounds of the initial uniform distribution in VQGAN and the disorder of Gaussian distribution initialization in Reg-VQ, this compactness is conducive to improving the usage of the codebook. At the same time, the orderliness makes the probability of each token being trained more balanced, reducing the active frequency variance.

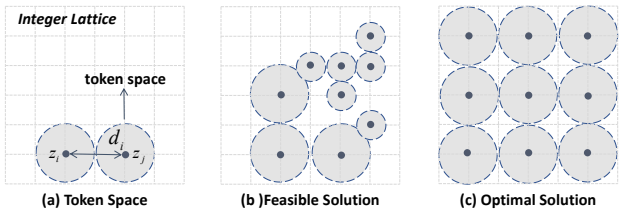

Figure 4. (a) The distance between two tokens. The grey area represents the token space, which can be approximately measured by the number of fundamental domains it occupies. (b)-(c) Visualization for a two-dimensional solution for Equ. 1. A dark black point represents a token and the grey area represents the corresponding spanned space of each token. (b) illustrates a feasible solution, where the space of all tokens is small and the minimum distance between some tokens is close. This does not meet two of our preconditions. (c) illustrates the optimal solution, where each token is relatively far and the space of all tokens is large. This presents the compact and ordered properties.

3.2. Hexagonal Distribution for Codebook

Based on the analyses in Sec. 3.1, training a good codebook problem is regarded as the sphere packing problem in Kepler's Conjecture. The sphere packing problem considers different distributions of equal spheres in the space. Its target is to maximize the packing density of packing spheres (Hales, 1998; Bernal, 1959). This just indicates that an optimal codebook has tokens with identical distances to the nearest token, but does not suggest the specific distribution of the codebook. In this part, we will explore this issue based on Kepler's Conjecture.

Remark 3. *In Lemma 1, $z_k = Bm$ with integral items of m also **follows the definition of integer lattice**[3] (Maehara, 2018), and m indicates the locations in the integer lattice. Thus, we derive the codebook distribution in the integer lattice.*

Specifically, in Kepler's Conjecture, the packing density is the ratio between the volume of spheres and the volume of total space. Similarly, we denote the codebook (token) density as η, measuring the ratio between the token space and the fundamental domain (Beardon, 1983) in the integer lattice. Thus, training a good codebook equals maximizing η. η is defined as

$$\eta = \frac{\pi^{\frac{n_z}{2}}}{\Gamma(\frac{n_z}{2}+1)} \cdot \frac{\sum_{i=1}^{K} r_i^{n_z}}{K \det(B)} \tag{2}$$

where $\det(B)$ is the volume of a fundamental domain of B. However, this is an NP-hard problem, and it's intractable to optimize this objective directly. In our work, we relax the

3. A formal definition of integer lattice is that given n linearly independent vectors $b_1, b_2, ..., b_m \in \mathbb{R}^n$ and $m \times n$ basis matrix B whose columns are $b_1, b_2, ..., b_n$, then the lattice generated by B is $\mathcal{L}(B) = \{Bx | x \in \mathbb{Z}^n\}$, where $\mathbb{Z}^n$ denotes the integer.

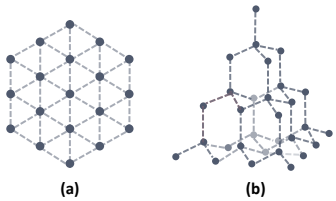

Figure 5. Kepler codebook in the 2D (a) and 3D (b) space.

original problem to maximize η with a constraint that the angle between any two basis vectors in B is equal. Specifically, we can build the codebook structure B as follows:

$$\begin{cases} b_1 = (1, 0, 0, \cdots, 0), \|b_i\| = 1 & (i) \\ b_i \cdot b_{i-1} = \cos\theta & (ii) \\ b_{i,j} = b_{i-1,j} \ (1 \leq j \leq i-2, i \geq 3) & (iii) \end{cases} \tag{3}$$

where $b_{i,j}$ is the j-th element in basis vector b_i, θ is the angle between two basis vectors. Rule (i) presents the initial condition and the constraint for the unit vector. Rule (ii) measures the distance between two vectors. Rule (iii) indicates starting from the i-th $(i \geq 3)$ basis vector, the elements in its first $(i-2)$-rows must be equal to the $(i-1)$-th vector, to ensure that the angle between it and the previous vectors remains constant. For example, the previous third vectors of B are $b_1 = (1, 0, 0, \cdots, 0), b_2 = (\frac{1}{2}, \frac{\sqrt{3}}{2}, 0, \cdots, 0), b_3 = (\frac{1}{2}, \frac{\sqrt{3}}{6}, \frac{\sqrt{6}}{12}, \cdots, 0)$.

Derived from the given codebook structure in Equ. 3, the optimization of the codebook density η becomes the following form:

$$\arg\max_{\theta} q^n(\theta)/\det(B), \tag{4}$$

where $q(\theta)$ means the maximum radium of sphere in the fundamental domain (Beardon, 1983) of B.

Lemma 2: *When $\theta = 60°$, the maximum codebook density is attained. The proof is provided in the Appendix.2.*

Remark 4: *Based on Lemma 2, one conclusion has been drawn that the codebook structure or distribution in two-dimensional space is **hexagonal**. This derived codebook with the hexagonal structure or distribution is named the **Kepler codebook**. Fig. 5 illustrates examples of the derived structures in 2D/3D.*

For a token, $z_k = Bm, m \in \mathbb{Z}^{n_z}$, and the i-th element is calculated as $z_{ki} = \sum_{j=1}^{n_z} b_{ij} m_j$, where for each dimension of m, it can be approximated to follow $m_j \sim U, i \in [1, n_z]$. That is, the i-th dimension of one token is approximately equivalent to the sum of n_z independent uniform distributions since the basis matrix B can be rotated that none entries in B are zero, which is mathematically regarded as a *Irwin-Hall distribution* (Hall, 1927),

$$z_{ki} \sim U^{n_z}(0, 1), \tag{5}$$

where U^{n_z} represents the sum of n_z independent uniform distributions. On one hand, every dimension z_{ki} of each token follows an n_z-dimensional Irwin-Hall distribution as depicted in Equ. 5. On the other hand, if the distribution of each codebook token entry is n_z-dimensional Irwin-Hall distribution, it will reach the previous target where making the space of all tokens larger and the distance between each token relatively further. Meanwhile, it also means the tokens in the codebook are compact and ordered, while the compact property will bring the high codebook usage potential and the ordered property will balance the train times of each token to solve the problem proposed in the title and finally improve the quality of image in both reconstruction and generalization task.

4. Kepler Codebook Training

4.1. Irwin-Hall Distribution Regularization

In Sec. 3.2, we conclude that each dimension of every token conforms to independent and identical n_z-dimensional Irwin-Hall distribution. Thus, we follow the principle of the Kepler codebook and propose an Irwin-Hall Distribution (IHD) regularization to constrain the training of the codebook. Specifically, we take the distribution of Kepler codebook as prior for codebook training. The prior distribution $P_{\text{prior}} = U^{n_z}(0, 1) = [p_1, p_2, ..., p_k]$, where p_i is a vector sampled from n_z-dimensional Irwin-Hall distribution, is utilized to regularize the vector quantization. The posterior distribution can be approximated $P_{\text{post}} = [z_1, z_2, ..., z_K]$. Accordingly, our Irwin-Hall distribution regularization is calculated by the distance between the prior and the predicted codebook distribution, to constrain the codebook training. It aims to facilitate the model to learn a compact codebook space with ordered token distribution to improve codebook usage and balance the training for each token. With the Kullback–Leibler (KL) divergence as the distance measure, it is defined as follows,

$$\mathcal{L}_{\text{IHD}} = \text{KL}(P_{\text{post}}, P_{\text{prior}}) = -\sum_{i=1}^{K} p_i \log \frac{z_i}{p_i}. \tag{6}$$

4.2. Codebook Partition

Due to the curse of dimensionality, the token distribution in the codebook becomes sparse, which can lead to inaccurate probability distribution estimation and then make it hard to calculate accurately the KL divergence in Equ. 6. Thus, we aim to obtain a lower-dimensional codebook distribution to reduce training complexity, by partitioning the encoder output and then quantifying the elements within each partition. This partitioning strategy is supported by the conclusion that every dimension of tokens conforms to Equ. 5 in an independent identical distribution. Precisely, we can shuffle the encoder output $\hat{z}_{\boldsymbol{q}}$ to low dimension $\hat{z}_q/d$, where d is the number of partitions in the quantization process, and then unshuffle to $z_{\boldsymbol{q}}$ for decoder to reconstruct the image.

Table 1. Quantitative comparison of image reconstruction and generation tasks with VQGAN (Esser et al., 2021), Reg-VQ (Zhang et al., 2023), FSQ (Mentzer et al., 2023) on the ADE20K and CelebA-HQ datasets. The notation [R] and [G] indicate whether the metric is for image reconstruction or generation.

Method	ADE20K			CelebA-HQ		
	PSNR[R]↑	rFID[R]↓	FID[G]↓	PSNR[R]↑	rFID[R]↓	FID[G]↓
VQ-VAE	19.95	49.21	60.29	23.39	28.38	39.57
VQGAN	18.89	28.17	38.53	22.44	12.74	17.42
Reg-VQ	18.44	23.69	34.47	22.05	10.09	15.34
FSQ	20.31	18.30	35.03	-	-	-
Ours	**21.34**	**16.87**	**33.84**	**24.85**	**8.59**	**14.96**

Table 2. Ablation study for our proposed method on ADE20K. The codebook partition number is $d = 4$. n_z is the codebook dimension. The codebook vector number K is set as 1024 in all experiments.

Method	d	n_z	rFID↓	usage↑
Baseline (VQGAN)	1	256	28.17	35%
+ IHD	1	256	26.28	40%
+ Partition	4	64	19.34	100%
+ IHD&Partition (Ours)	**4**	**64**	**16.87**	**100%**

For its implementation, VQ-GAN flattens the encoder output $\hat{z}_{\boldsymbol{q}}$ to $hw \times n_z$ for quantization, while we reshape it to $hwd \times n_z/d$. Correspondingly, the dimension of the codebook n_z reduces to n_z/d to mitigate the effects of the curse of dimensionality. After quantization, the flattened $z_{\boldsymbol{q}}$ is reshaped back to $h \times w \times n_z$. This ensures 0imension consistency in the network between the output of encoder and the input of decoder, aligning with VQGAN. Similarly, this strategy can also be used in the dimension hw, by flattening hw to $d \times hw/d$ during the quantization process and reshaping it back to hw after quantization.

5. Experiments

5.1. Experimental Settings

Datasets. For empirical comparison with existing methods, we conduct the codebook training on ADE20K (Zhou et al., 2017) and CelebA-HQ (Liu et al., 2015) datasets, respectively, in two tasks of image reconstruction and semantic image synthesis. The evaluation results are reported on the validation sets of these two datasets, respectively. To further demonstrate the generalization capabilities of our method, we extend our evaluation to cross-domain datasets, namely training on the ADE20K dataset and testing on another three datasets, MS-COCO (Lin et al., 2014), LSDIR (Li et al., 2023), and DIV2K (Timofte et al., 2017). Additionally, we undertake a downstream application of the learned codebooks to the image super-resolution task. Specifically, we train the latent diffusion model (Rombach et al., 2022) on a large real-world image super-resolution dataset DRealSR (Wei et al., 2020), whose autoencoder is replaced by our models trained on the ImageNet dataset.

Implementation details. Our model follows the similar architecture of VQGAN (Esser et al., 2021), which compresses 256×256 images into 16×16 tokens (where $f = 16$). We utilize the proposed Irwin-Hall distribution regularization for the reconstruction training to optimize the codebook, which has a $K \times n_z/d$ size. We set $d = 4$ in all experiments. All the experiments are conducted on 8 NVIDIA Tesla A100-40G GPUs. The model optimization is performed using the AdamW optimizer (Loshchilov & Hutter, 2017) with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.95$, and a base learning rate of 4.5×10^{-6}. The batch size is 96 for the reconstruction and 64 for the generation. Detailed efficiency analyses can be found in the Appendix.

Metrics. In the image reconstruction and semantic image synthesis tasks, following VQGAN, we use FID (Heusel et al., 2017) and PSNR for comparison. For cross-domain evaluation and downstream application, we adopt PSNR, SSIM, LPIPS, and FID to evaluate the model's capabilities from various aspects.

5.2. Quantitative Evaluation

For the **image reconstruction task**, the experiments are conducted on the ADE20K and CelebA-HQ datasets, respectively. The comparative results are detailed in Tab. 1. It is noteworthy that our model outperforms VQ-VAE (Van Den Oord et al., 2017), VQGAN (Esser et al., 2021), and Reg-VQ (Zhang et al., 2023) by 1.39, 2.45, 2.9 dB in PSNR and by 32.34, 11.3, 6.82 in rFID on ADE20K, respectively. On CelebA-HQ, compared to VQ-VAE, VQGAN, and Reg-VQ, there is a 1.46, 2.41, 2.8 dB improvement in PSNR and 19.79, 4.15, 1.5 performance gains in rFID. Besides, the comparison with the state-of-the-art method, FSQ, In comparison with FSQ (Mentzer et al., 2023) trained from scratch on ADE20K using its official codes, our method outperforms FSQ by 1.03 dB in PSNR and 1.43 in rFID. These significant performance improvements demonstrate that our model performs better than existing works, *i.e.*, VQ-VAE, VQGAN and Reg-VQ, on two different datasets. One main difference between our method and those existing works is the introduction of IHD regulation for the codebook training. Thus, these improvements can be attributed to the consideration of the codebook distribution, which is vital for effectively reconstructing image details and reducing artifacts.

For the **semantic image synthesis task**, the experimental results on ADE20K and CelebA-HQ datasets are provided in Tab. 1. Our model also achieves a remarkable improvement on these two different datasets. For example, in comparison with VQ-VAE, VQGAN, and Reg-VQ, our model

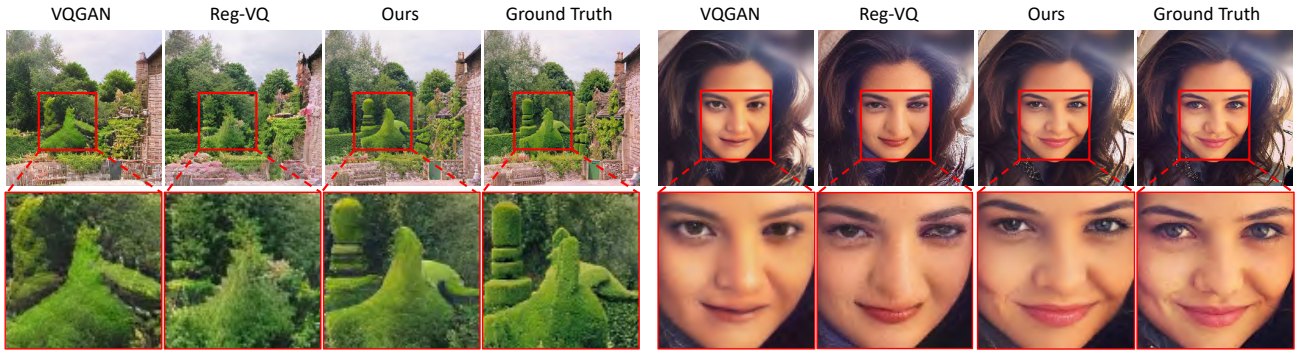

Figure 6. Reconstruction results on ADE20K and CelebA-HQ from different models.

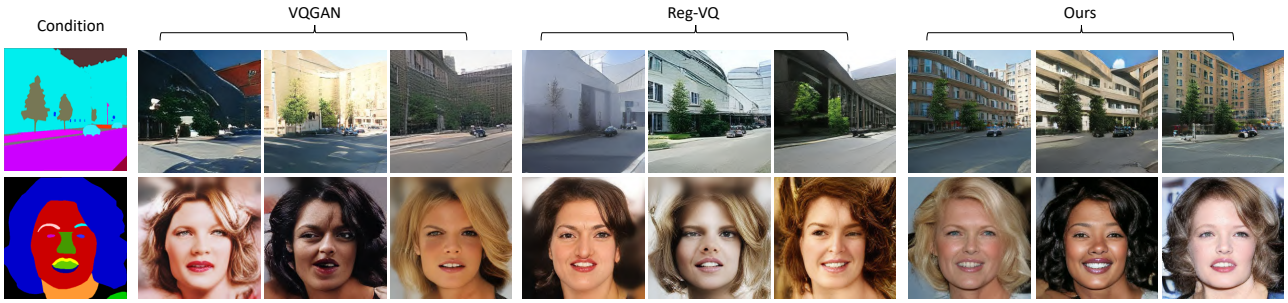

Figure 7. Semantic segmentation synthesis on ADE20K and CelebA-HQ. The semantic segmentation map in the first column is the condition for generation.

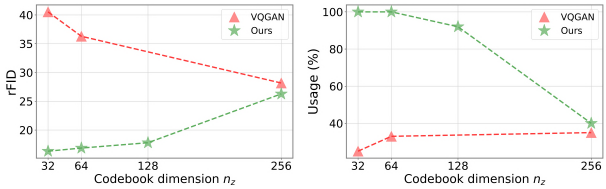

Figure 8. Ablation study on the effect of codebook dimension n_z on ADE20K. In our method, $n_z = 256$ means no codebook partition technique is utilized, $n_z = 128$ means 2 partitions, and $n_z = 64$ means 4 partitions, and so on. For VQGAN, the codebook dimension is reduced directly. The codebook vector number K is set as 1024 in all experiments.

outperforms them by 26.45, 4.69, and 1.19 in FID on the ADE20K dataset. It indicates that our Kepler codebook with a compact and ordered distribution is beneficial in producing conditional images for the autoregressive model.

5.3. Qualitative Evaluation

We present a comparison of reconstruction visualizations for the ADE20K and CelebA-HQ datasets, as illustrated in Fig. 6. Our model demonstrates superior performance in image reconstruction. For instance, our model reproduces the shape of the haystack more accurately, and the woman's eyes and lips are more faithful. Compared with VQGAN and Reg-VQ, our model preserves details and structure without distortions. Fig. 7 shows the results generated by each model using the same semantic segmentation map. Our model produces images corresponding to the semantic map while retaining reasonable details in natural scenes, indoor

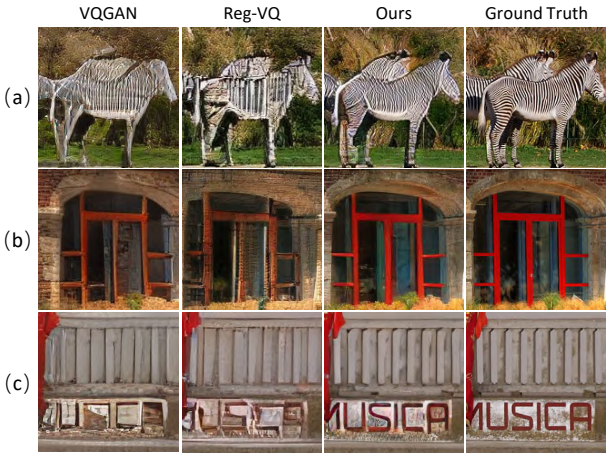

Figure 9. Cross-domain experiment. All models are trained on the ADE20K dataset and tested on (a) MS-COCO, (b) LSDIR, and (c) DIV2K.

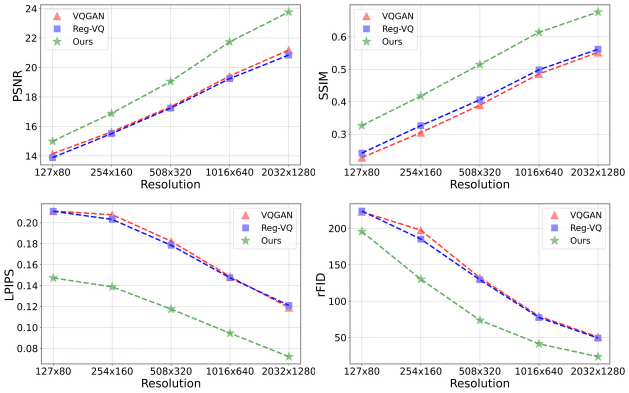

Figure 10. Multi-resolution cross-domain evaluation on DIV2K validation set. We train the three models on ADE20K with 256×256 resolution and test them on five different resolutions from low to high on the DIV2K dataset.

Table 3. Quantitative reconstruction and SR comparison on ImageNet and DRealSR validation sets.

Method	Task	Dataset	PSNR↑	SSIM↑	LPIPS↓	rFID/FID↓
LDM	Rec.	ImageNet	27.48	0.826	0.024	0.42
Ours			**28.00**	**0.837**	**0.019**	**0.33**
LDM	SR	DRealSR	24.86	0.715	0.080	35.66
Ours			**26.15**	**0.750**	**0.065**	**26.11**

scenes, and faces.

5.4. Ablation Study and Evaluation

IHD regularization & Codebook partition. The ablation study on those two strategies is provided in Tab. 2. With the proposed IHD regularization based on the baseline model VQGAN, the performance of both rFID and codebook usage is improved. With the codebook partition, there is an improvement of about 31% in rFID, and the usage rate is directly increased from 35% to 100%. When both strategies are used, our model makes an improvement of 40% in rFID compared to VQGAN and the usage rate is still 100%. Since the dimension of tokens in codebook is so high that it is difficult to compute the KL divergence accurately in Equ. 6, only using IHD regularization improves performance slightly. With the codebook partition, though there is a remarkable improvement, it cannot confirm the well-ordered property of codebook distribution. Thus the collaboration of the codebook partition to lower the dimension of tokens and the IHD regularization to confirm the well-ordered property of the distribution, our model improves the image reconstruction by an even larger margin.

Codebook dimension. The use of the codebook partition technique naturally alters the dimensionality of the codebook. To explore the effect on the quality of image reconstruction when only reducing the dimension but no partition performed, we conduct a set of ablation experiments. As shown in Fig. 8, when the token number K remains unchanged, the quality of image reconstruction gradually deteriorates and the codebook usage decreases as the dimension decreases. In contrast, our model achieves a significant improvement in image reconstruction quality and codebook utilization as the dimension decreases (which implies a corresponding increase in the partition number n_z). This implies that there is a fundamental difference between codebook partition and dimensionality reduction.

Balance evaluation of codebook distribution. To fully verify our method, we measure the balance of the codebook distribution before and after training. The distribution balance implies that the number of other tokens within the neighborhood of each token is roughly equal, avoiding situations where certain tokens have disproportionately many or few neighboring tokens within their hollow neighborhoods. Based on this principle, we set a radius for the hollow neighborhoods and evaluate the balance of the distribution by calculating the variance of the number of other tokens within each token's hollow neighborhood. If the variance is large, it suggests that some tokens in the distribution are either overly dense or sparse, whereas a small variance indicates a relatively even distribution among the tokens.

Specifically, in Tab. 4, the parameter d_{rank} represents that we select the d_{rank}-th distance as the hollow neighborhood radius when arranging all the distances between any two tokens in the codebook in ascending order. We present evaluation results on the ADE20K dataset with a codebook dimensionality K=1024. As shown in Tab. 4, during the early stages of training, all models exhibit varying degrees of imbalance in their distribution states. However, after applying our optimized training strategy, we observe a significant decrease in the variance values, with an average reduction approximately twice that of the initial state. In contrast, other models show minimal improvement in their distribution balance after training, still closely resembling their initial distribution states. Notably, regardless of the d_{rank} value chosen, our method consistently reduces the variance to about half that achieved by alternative methods, further substantiating the superior performance of our model in improving the balance of distributions.

5.5. Cross-Domain Evaluation

The generalization performance of VQGAN, Reg-VQ, and our model is examined by training them on ADE20K and testing on MS-COCO, LSDIR, and DIV2K, respectively. Specifically, these models are trained on ADE20K with a resolution of 256×256 and tested on the other three datasets. As shown in Fig. 9, our model produces impressive visual

Table 4. Balance evaluation of the codebook distribution trained on the ADE20K dataset. The variance of the number of other tokens within each token's hollow neighborhood, explained in Sec.5.4, is employed to measure the balance degree.

d_{rank}	2048			5120			10240			20480		
	VQGAN	Reg-VQ	Ours	VQGAN	Reg-VQ	Ours	VQGAN	Reg-VQ	Ours	VQGAN	Reg-VQ	Ours
Variance(Before training)	54.2	66.3	54.1	248.6	286	252.3	765.2	837.5	768.1	2269.3	2377.4	2241.9
Variance(After training)	50.6	67.3	**23.5**	242.6	282.4	**113.7**	798.7	837.7	**410.1**	2635.3	2398.5	**1505.6**

Figure 11. Multi-resolution cross-domain visualization on DIV2K validation set with five resolutions from high to low. Notably, the models are trained on ADE20K with 256×256 resolution. Please zoom in for a better view.

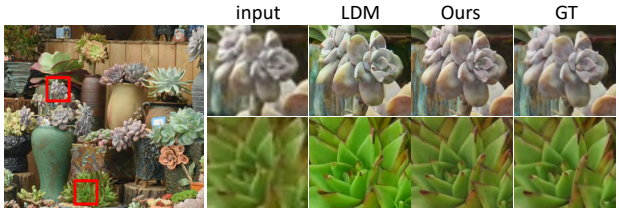

Figure 12. Evaluation on the super-resolution task on the DRealSR datset.

results. This demonstrates the accuracy of our codebook in modeling discrete representations and the ability to generalize to cross-domain images while still achieving accurate reconstruction compared to other models. Furthermore, we conduct a multi-resolution cross-domain experiment on the DIV2K dataset, using trained models on ADE20K. As shown in Fig. 10, our model achieves superior results on the four metrics. The visualization for five resolution images is shown in Fig. 11. Additional visualization results and detailed metrics can be found in the Appendix.

5.6. Downstream Application: Super-Resolution

We present a downstream application to image super-resolution (SR). We first train our model as an autoencoder from scratch on the ImageNet dataset, employing the same downsample factor ($f = 4$) and codebook configurations (VQ, $K = 8192, n_z = 3$) with the autoencoder in Latent Diffusion Model (LDM) (Rombach et al., 2022). We then apply our model to the LDM to train an SR model on the DRealSR dataset. All configurations are consistent with the SR tasks in LDM. The quantitative result for reconstruction and SR is shown in Tab. 3 and the qualitative result for SR is shown in Fig. 12. Our model has significant advantages for both the reconstruction and SR tasks, especially in terms of rFID/FID enhancement, indicating that Kepler Codebook is also beneficial for enhancing downstream tasks. Additional visualization results can be found in the Appendix.

6. Conclusion

In this paper, we make a theoretical and technical attempt to explore the codebook to address the typical codebook collapse and ensure full training of codebook tokens for high codebook usage. The codebook distribution is formulated and derived in conjunction with Kepler's Conjecture in a principle way. To constrain the distribution of tokens, the derived Irwin-Hall distribution regularization for Kepler codebook training is conducted together with a codebook partition strategy to improve codebook usage. Extensive experiments have been conducted to evaluate our trained codebook for image reconstruction and generation on natural and human face datasets, respectively, demonstrating a remarkable performance in these tasks. Moreover, the proposed Kepler codebook has been further evaluated across datasets and even for reconstructing images with different resolutions, demonstrating a promising codebook generalization. Our main contributions, including the mathematical derivation of the codebook distribution from Kepler's Conjecture perspective and the proposed Kepler codebook together with its training manner, are expected to be useful for further insightful research.

Acknowledgements

This work is supported in part by National Natural Science Foundation of China (NSFC) under Grant No.62376292, 62376209, U21A20470, 62325605, China Postdoctoral Science Foundation under Grant No. 2023M731964, Guangzhou Science and Technology Program (No.2024A04J6365), and Guangdong Province Key Laboratory of Information Security Technology.

Impact Statement

Our work aims to explore the problem of codebook collapse for its training and learn discrete representations with vector quantization. The trained codebook is a precondition for generative models and is the base for visual content generation. The main contribution is casting codebook training as the densest sphere packing and providing a principle solution to derive a compact and structured codebook distribution, which presents a promising potential to extend to the learning visual representation. Ethical considerations are crucial, as generative models can be misused to create misleading content. This paper highlights the significance of responsible use of technology to ensure that technological advancements benefit our society.

References

Baevski, A., Schneider, S., and Auli, M. vq-wav2vec: Self-supervised learning of discrete speech representations. *arXiv preprint arXiv:1910.05453*, 2019.

Beardon, A. F. *Fundamental Domains*, pp. 204–252. Springer New York, New York, NY, 1983.

Bernal, J. D. A Geometrical Approach to the Structure Of Liquids. , 183(4655):141–147, 1959.

Chen, Y., Yuan, J., Tian, Y., Geng, S., Li, X., Zhou, D., Metaxas, D. N., and Yang, H. Revisiting multimodal representation in contrastive learning: from patch and token embeddings to finite discrete tokens. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15095–15104, 2023.

Dhariwal, P., Jun, H., Payne, C., Kim, J. W., Radford, A., and Sutskever, I. Jukebox: A generative model for music, 2020.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Esser, P., Rombach, R., and Ommer, B. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12873–12883, 2021.

Gray, R. Vector quantization. *IEEE Assp Magazine*, 1(2): 4–29, 1984.

Hales, T. C. An overview of the Kepler conjecture. *arXiv Mathematics e-prints*, 1998.

Hall, P. The distribution of means for samples of size n drawn from a population in which the variate takes values between 0 and 1, all such values being equally probable. *Biometrika*, pp. 240–245, 1927.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Proceedings of Advances in Neural Information Processing Systems*, 2017.

Huang, M., Mao, Z., Chen, Z., and Zhang, Y. Towards accurate image coding: Improved autoregressive image generation with dynamic vector quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22596–22605, 2023.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Kepler, J. *The Six-Cornered Snowflake*. The Six-Cornered Snowflake, 1966.

Lee, D., Kim, C., Kim, S., Cho, M., and Han, W.-S. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11523–11532, 2022.

Li, Y., Zhang, K., Liang, J., Cao, J., Liu, C., Gong, R., Zhang, Y., Tang, H., Liu, Y., Demandolx, D., Ranjan, R., Timofte, R., and Van Gool, L. Lsdir: A large scale dataset for image restoration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1775–1787, 2023.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft coco: Common objects in context. In *Proceedings of European Conference on Computer Vision*, pp. 740–755, 2014.

Liu, Z., Luo, P., Wang, X., and Tang, X. Deep learning face attributes in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3730–3738, 2015.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Maehara, Hiroshi Martini, H. Elementary geometry on the integer lattice. *Aequationes mathematicae*, 92(4), 2018.

Mentzer, F., Minnen, D., Agustsson, E., and Tschannen, M. Finite scalar quantization: Vq-vae made simple. *arXiv preprint arXiv:2309.15505*, 2023.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *Proceedings of International Conference on Machine Learning*, pp. 8821–8831, 2021a.

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., and Sutskever, I. Zero-shot text-to-image generation. In *Proceedings of International Conference on Machine Learning*, pp. 8821–8831, 2021b.

Razavi, A., Van den Oord, A., and Vinyals, O. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., and Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

Roy, A. and Grangier, D. Unsupervised paraphrasing without translation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pp. 6033–6039, 2019.

Sagan, H. Some reflections on the emergence of space-filling curves: the way it could have happened and should have happened, but did not happen. *Journal of the Franklin Institute*, 328(4):419–430, 1991. ISSN 0016-0032.

Timofte, R., Agustsson, E., Gool, L. V., Yang, M.-H., and Zhang, L. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 1110–1121, 2017.

Van Den Oord, A., Kalchbrenner, N., and Kavukcuoglu, K. Pixel recurrent neural networks. In *Proceedings of International Conference on Machine Learning*, pp. 1747–1756, 2016.

Van Den Oord, A., Vinyals, O., et al. Neural discrete representation learning. In *Proceedings of Advances in Neural Information Processing Systems*, 2017.

Wang, D., Deng, L., Yeung, Y. T., Chen, X., Liu, X., and Meng, H. Vqmivc: Vector quantization and mutual information-based unsupervised speech representation disentanglement for one-shot voice conversion. *arXiv preprint arXiv:2106.10132*, 2021.

Wei, P., Xie, Z., Lu, H., Zhan, Z., Ye, Q., Zuo, W., and Lin, L. Component divide-and-conquer for real-world image super-resolution. In *Proceedings of European Conference on Computer Vision*, pp. 101–117, 2020.

Williams, W., Ringer, S., Ash, T., MacLeod, D., Dougherty, J., and Hughes, J. Hierarchical quantized autoencoders. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, pp. 4524–4535, 2020.

Wu, D.-Y., Chen, Y.-H., and Lee, H.-Y. Vqvc+: One-shot voice conversion by vector quantization and u-net architecture. *arXiv preprint arXiv:2006.04154*, 2020.

You, T., Kim, S., Kim, C., Lee, D., and Han, B. Locally hierarchical auto-regressive modeling for image generation. In *Proceedings of Advances in Neural Information Processing Systems*, pp. 16360–16372, 2022.

Yu, J., Li, X., Koh, J. Y., Zhang, H., Pang, R., Qin, J., Ku, A., Xu, Y., Baldridge, J., and Wu, Y. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.

Zhang, J., Zhan, F., Theobalt, C., and Lu, S. Regularized vector quantization for tokenized image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18467–18476, 2023.

Zheng, C. and Vedaldi, A. Online clustered codebook. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 22798–22807, 2023.

Zhou, B., Zhao, H., Puig, X., Fidler, S., Barriuso, A., and Torralba, A. Scene parsing through ade20k dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 633–641, 2017.

A. Proof Details

A.1. Codebook Training is Kepler's Conjecture Proof Details

To ensure the establishment of a representative feature space, two essential conditions are posited. Under the assumption that the representation space is bounded (a reasonable consideration for deterministic data within a given training set), the conditions are delineated as follows:

1) *A well-constructed codebook comprising N tokens is requisite, with the space spanned by all tokens maximizing its expansiveness.* ***2)*** *The distance between each token should be relatively large, leading to a relatively uniform probability distribution for the training of each token.*

Further more, according to the first precondition, the space of each token is contact instead of separation to get larger spanned space. According to the based model VQGAN, it usually use the nearest neighbour encoding resulting the spanned space of any two tokens will contact in their midpoint. Thus we can get the relation between r_i and d_i of i-th token that $2r_i = d_i$. In other words, the constrained condition can be transformed into $g(d_1, d_2, ..., d_K) = \sum_{i=1}^{K} d_i^{n_z} \leq |\hat{z}| / 2^{n_z}$. Formally, these conditions can be expressed as follows:

$$\begin{cases} \quad arg\max f(d_1, d_2, ..., d_K) \\ s.t. \;\; g(d_1, d_2, ..., d_K) = \sum_{i=1}^{K} d_i^{n_z} \leq |\hat{z}| / 2^{n_z} \end{cases} \tag{7}$$

where d_i is the minimum distance between the i-th token and all the other tokens, K is the number of tokens in codebook, n_z is the dimension of the token and $|\hat{z}|$ is the space of $\hat{z} = E(x)$. Since $f(d_1, d_2, ..., d_K) \doteq \min(d_1, d_2, ..., d_K)$, we may infer that d_1 represents the minimum distance within $d_1, d_2, ..., d_K$. Consequently, we can reformulate Equation 7 into an equivalent expression as follows:

$$\begin{cases} arg\min \quad -d_1 \\ s.t. \quad g_1(d_1, d_2, ..., d_K) = g(d_1, d_2, ..., d_K) - |\hat{z}| / 2^{n_z} \leq 0 \\ \qquad g_2(d_1, d_2, ..., d_K) = d_1 - d_2 \leq 0 \\ \qquad g_3(d_1, d_2, ..., d_K) = d_1 - d_3 \leq 0 \\ \qquad\qquad \vdots \\ \qquad g_K(d_1, d_2, ..., d_K) = d_K - d_2 \leq 0 \end{cases} \tag{8}$$

It is apparent that the set G becomes nonlinear when $G = \{\nabla g_i(\boldsymbol{d}^*) : i = 1, 2, ..., K\}$ serves as the optimal solution for Equ. 8. Initially, we compute each $\nabla g_i(\boldsymbol{d}^*)$ as outlined below:

$$\begin{aligned} &\nabla g_1(\boldsymbol{d}^*) = (\tfrac{\partial g_1}{\partial d_1}, \tfrac{\partial g_1}{\partial d_2}, ..., \tfrac{\partial g_1}{\partial d_K}) = (n_z d_1^{n_z-1}, n_z d_2^{n_z-1}, ..., n_z d_K^{n_z-1}) \\ &\nabla g_2(\boldsymbol{d}^*) = (\tfrac{\partial g_1}{\partial d_1}, \tfrac{\partial g_1}{\partial d_2}, ..., \tfrac{\partial g_1}{\partial d_K}) = (1, -1, 0, 0, ..., 0) \\ &\nabla g_3(\boldsymbol{d}^*) = (\tfrac{\partial g_1}{\partial d_1}, \tfrac{\partial g_1}{\partial d_2}, ..., \tfrac{\partial g_1}{\partial d_K}) = (1, 0, -1, 0, ..., 0) \\ &\nabla g_4(\boldsymbol{d}^*) = (\tfrac{\partial g_1}{\partial d_1}, \tfrac{\partial g_1}{\partial d_2}, ..., \tfrac{\partial g_1}{\partial d_K}) = (1, 0, 0, -1, ..., 0) \\ &\qquad\qquad \vdots \\ &\nabla g_K(\boldsymbol{d}^*) = (\tfrac{\partial g_1}{\partial d_1}, \tfrac{\partial g_1}{\partial d_2}, ..., \tfrac{\partial g_1}{\partial d_K}) = (1, 0, 0, 0, ..., -1) \end{aligned} \tag{9}$$

It is evident that $\nabla g_i(\boldsymbol{d}^*)$ is linearly independent for $2 \leq i \leq K$. Simultaneously, considering that the distance between any two tokens is greater than zero ($d_1, d_2, ..., d_K > 0$), a linear combination of $\nabla g_2(\boldsymbol{d}^*), \nabla g_3(\boldsymbol{d}^*), ..., \nabla g_K(\boldsymbol{d}^*)$ cannot represent $\nabla g_1(\boldsymbol{d}^*)$. Consequently, Equ. 8 adheres to the regularity conditions.

We consider using the Lagrange Multiplier Method to solve the problem. We transform Equ. 8 into the Lagrange function as follows:

$$L(\boldsymbol{d}, \mu_1, \mu_2, ..., \mu_K) = -d_1 + \sum_{i=1}^{K} \mu_i g_i(\boldsymbol{d}) \tag{10}$$

where $\boldsymbol{d}$ represents $(d_1, d_2, ..., d_K)$ and μ_i is the Lagrange Multiplier. Using the KKT conditions, potential optimal solutions can be found:

$$\begin{cases} \nabla_{\boldsymbol{d}} L = 0 \\ \mu_i g_i(\boldsymbol{d}) = 0 & i = 1, 2, ..., K \\ \mu_i \geq 0 & i = 1, 2, ..., K \\ g_i(\boldsymbol{d}) \leq 0 & i = 1, 2, ..., K \end{cases} \tag{11}$$

One such optimal solution is:

$$\begin{cases} d_1 = d_2 = ... = d_K \\ g_1(\boldsymbol{d}) = 0 \end{cases} \tag{12}$$

The expression corresponding to Equ. 12 maximizes the space occupied by all tokens when the distances d_i of each token are equal within the constraint space. Simultaneously, this expression is analogous to Kepler's Conjecture, which seeks to determine the maximum density of sphere packing.

A.2. Hexagonal Distribution is Good for Codebook Proof Details

To demonstrate the existence of the basis matrix B when $\theta \leq 90°$ in any dimension, consider $B = (\overrightarrow{b_1}, \overrightarrow{b_2}, ..., \overrightarrow{b_{n_z}})$. It is evident that the matrix $B^T B$ is semi-positive definite, indicating that $x^T B^T B x \geq 0$. For $x = (1, 1, ..., 1)$, the following expression can be derived:

$$x^T B^T B x = \sum_{ij} b_i^T b_j = \sum_{i \neq j} b_i^T b_j + n_z \geq 0 \tag{13}$$

The following form can be derived from Equ. 13:

$$n_z(n_z - 1) \max_{i \neq j} (\overrightarrow{b_i} \cdot \overrightarrow{b_j}) \geq \sum_{i \neq j} b_i^T b_j \geq -n_z \tag{14}$$

Consequently, we can draw the conclusion from Equ. 14:

$$\max_{i \neq j} (\overrightarrow{b_i} \cdot \overrightarrow{b_j}) \geq -\frac{1}{n-1} \tag{15}$$

Given that both b_i and b_j are unit vectors, it follows that $\cos(\theta) \geq -\frac{1}{n_z - 1}$ in n_z-dimensional space. To elaborate further, the range of angles between basis vectors in B is $0 \leq \theta \leq \arccos(-\frac{1}{n-1})$ in n_z dimensions, implying that the basis matrix can be constructed using the method outlined in the paper. As the dimension n_z approaches infinity, the range of angles θ approximates the interval from 0 to 90 degrees. Exploiting this property of angle range in sufficiently high dimensions and the associated symmetry, we maximize the following expression within the angle range from 0 to 90 degrees:

$$\max_{\theta} \frac{q^n(\theta)}{\det(B)} \tag{16}$$

Before addressing Equ. 16, let us elucidate the choice of $q(\theta)$. Without loss of generality, the basis matrix B can be constructed in the following manner:

$$\begin{cases} b_1 = (1, 0, 0, ..., 0), \|b_i\| = 1 \\ b_i \cdot b_{i-1} = \cos\theta \\ b_{i,j} = b_{i-1,j} (1 \leq j \leq i-2, i \geq 3) \end{cases} \tag{17}$$

Here, the angles between and basis vectors in B are θ. Moreover, the basis matrix B is a nonnegative matrix. Then $q(\theta)$ can be described as follows:

$$q(\theta) = \min_{\boldsymbol{\alpha} \neq \boldsymbol{\beta}} \|B\boldsymbol{\alpha} - B\boldsymbol{\beta}\| \tag{18}$$

Here, $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ belong to $\{0,1\}^{n_z}$. Given that the basis matrix B is a nonnegative matrix, the possible values for $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are constrained to

$$\{(0,0,...,0),(1,0,...,0),(0,1,...,0),...,(0,0,...,1)\}$$

. In other words, if $\boldsymbol{\alpha} = (1,1,0,...,0)$ and $\boldsymbol{\beta} = (0,0,0,...,1)$, the distance between $B\boldsymbol{\alpha}$ and $B\boldsymbol{\beta}$ is greater than when $\boldsymbol{\alpha} = (1,0,0,...,0)$ and $\boldsymbol{\beta} = (0,0,0,...,1)$. Meanwhile, Equ. 18 can be accurately transformed into the following form:

$$q(\theta) = \min(1, \min_{i \neq j} \|b_i - b_j\|) \tag{19}$$

where the value 1 represents the distance between $\boldsymbol{\alpha} = (0,0,...,0)$ and $\boldsymbol{\beta}$. And b_i represents $B\boldsymbol{\alpha}$ where only i-th entry in $\boldsymbol{\alpha}$ is 1, other entries are zero. It's easy to find that $\|b_i - b_{i+1}\| = \|b_i - b_j\|, j = i+1, i+2, ..., K$. Therefore, Equ. 19 can be transformed to the following form.

$$q(\theta) = \min(1, \min_{i} \|b_i - b_{i+1}\|) \tag{20}$$

In order to calculate $\|b_i - b_{i+1}\|$, here we first derive the relationship between adjacent diagonal elements in the basis matrix B. The relationship between b_i and b_{i+1} can be derived from Equ. 17 as follows.

$$\begin{cases} b_i \cdot b_{i+1} = \cos\theta \\ \|b_i\| = \|b_{i+1}\| = 1 \end{cases} \tag{21}$$

From Equ. 21, we can get the relationship between b_{ii} and b_{i+1i+1} as follows.

$$b_{i+1i+1}^2 = 2(1-\cos\theta) - \frac{(1-\cos\theta)^2}{b_{ii}^2} \tag{22}$$

While for the relationship between b_{ii} and b_{i+1i} is $b_{i+1i} = \frac{b_{ii}^2 - 1 + \cos\theta}{b_{ii}}$. Thus, $\|b_i - b_{i+1}\|$ can be calculated as follows.

$$\begin{aligned} \|b_i - b_{i+1}\| &= ((b_{ii} - b_{i+1i})^2 + b_{i+1i+1}^2)^{\frac{1}{2}} \\ &= ((b_{ii} - b_{ii} + \tfrac{1-\cos\theta}{b_{ii}})^2 + 2(1-\cos\theta) - \tfrac{(1-\cos\theta)^2}{b_{ii}^2})^{\frac{1}{2}} \\ &= (2(1-\cos\theta))^{\frac{1}{2}} \\ &= 2\sin\tfrac{\theta}{2} \end{aligned} \tag{23}$$

It follows that $q(\theta) = \min(1, 2\sin\frac{\theta}{2})$. When $\theta > 60°$, it is evident that $q(\theta) = \frac{1}{2}$, and $\prod_{i=1}^{k} b_{i,i}$ increases. This indicates that the result for $\theta = 60°$ cannot be surpassed if $\theta > 60°$ in Equation 16. Therefore, we can narrow down the range of angles under consideration to $0 \leq \theta \leq 60°$.

To simplify the discussion, let us use the following symbols to represent the optimization target in Equ. 16:

$$h(n,\theta) = \frac{(2\sin\frac{\theta}{2})^n}{\prod_{i=1}^{n} b_{ii}} \tag{24}$$

For the 2-dimensional case, Equ. 24 takes the following form:

$$h(2,\theta) = \frac{2\sin(\frac{\theta}{2})}{\sin\theta} = \frac{1}{\cos(\frac{\theta}{2})} \tag{25}$$

Clearly, when the angle θ equals 60 degrees, $h(2,\theta)$ attains its maximum.

Similarly, we can demonstrate that for $n = 3$, $\frac{2\sin(\frac{\theta}{2})}{b_{22}}$ also achieves its maximum at $\theta = 60°$. This implies that at an angle of 60 degrees, $h(3,\theta)$ reaches its maximum because both $h(2,\theta)$ and $\frac{2\sin(\frac{\theta}{2})}{b_{22}}$ attain their maxima at $\theta = 60°$.

Assuming that $n = 2, 3, ..., k$ supports the conclusion that $\frac{\sin(\frac{\theta}{2})}{b_{nn}}$ reaches its maximum at $\theta = 60°$, then when $n = k+1$, we can draw the following conclusions:

$$\begin{aligned}
\frac{\sin(\frac{\theta}{2})}{b_{k+1k+1}} &= \frac{\sin(\frac{\theta}{2})}{\sqrt{2(1-\cos\theta) - \frac{(1-\cos\theta)^2}{b_{kk}^2}}} \\
&= \frac{\sin(\frac{\theta}{2})}{\sqrt{4\sin^2(\frac{\theta}{2}) - \frac{4\sin^4(\frac{\theta}{2})}{b_{kk}^2}}} \\
&= \frac{1}{2}(1 - \frac{\sin^2(\frac{\theta}{2})}{b_{kk}^2})^{-1}
\end{aligned} \tag{26}$$

In the above assumption, we deduce that when $\theta = 60°$, $\frac{\sin(\frac{\theta}{2})}{b_{kk}}$ is the largest. Thus, when $\theta = 60°$, $\frac{\sin(\frac{\theta}{2})}{b_{k+1k+1}}$ reaches its maximum. More to specifically, that implies that $h(k+1,\theta)$ is the largest at $\theta = 60°$.

To sum up, when $\theta = 60°$, the maximum codebook density $h(n_z,\theta)$ is attained.

B. Further Evaluation for Codebook Partition

We conclude that each entry in the codebook follows an independent identical distribution. Consequently, we employ the codebook partition to enhance the image quality in the reconstruction and generalization processes. Specifically, this implies that there can be multiple variations of the codebook partition method. As described in the following Table 5, reshaping the model's encoder output $\hat{z}_{\boldsymbol{q}}$ from hwn_z to $hwd \times n_z/d$ or from $n_z hw$ to $d \times n_z hw/d$ contributes to improving the image quality.

Table 5. Ours reshapes encoder output $\hat{z}_{\boldsymbol{q}}$ from hwn_z to $hwd \times n_z/d$ where d means the number of partitions. Ours(w/o permute) is reshapes encoder output $\hat{z}_{\boldsymbol{q}}$ from $n_z hw$ to $d \times n_z hw/d$. It shows better reconstruction image quality in both reshape methods which further proves each entry in codebook is an independent identical distribution, thus it can be used in any reshape methods in the quantized process.

Model	PSNR↑	rFID↓
Reg-VQ	18.44	23.69
Ours(w/o permute)	20.31	20.43
Ours	21.71	16.39

C. Efficiency analysis

The main modifications of our method to the baseline are a KL regularization-based loss and the codebook partition which both bring negligible computations. A comparison in terms of the parameters can be found in Tab. 6.

With almost the same parameter size and FLOPS, VQGAN, Reg-VQ, and Ours require almost the same training hours, as shown in Tab. 7.

Table 6. The efficiency analysis comparison on ADE20K.

Model	#Param	FLOPS
VQGAN	376.4M	264.1G
Reg-VQ	376.1M	264.2G
Ours	377.2M	264.3G

Table 7. The details about batch, epoch and training time set in the training process.

Task	Dataset	Batch	Epoch	Training time
Reconstruction	ADE20K	96	100	14h
Reconstruction	Celeb-HQ	96	100	14h
Generation	ADE20K	64	50	18h
Generation	Celeb-HQ	64	50	18h

D. More Visualization Results

D.1. More reconstruction and generation results

We provide additional reconstruction and semantic segmentation synthesis results on ADE20K and CelebA-HQ datasets in Fig.13, Fig.14 and Fig.15, respectively.

D.2. More cross-domain results

We provide additional cross-domain visualization comparison on MS-COCO LSDIR and DIV2K datasets in Fig.16, along with multi-resolution results for DIV2K in Fig.17. More detailed metrics are shown in Tab. 8. In the comparison of cross-domain datasets with identical resolution, our model outperforms others in reconstructing various elements such as animals, architecture, text, landscapes, etc. When comparing cross-domain multi-resolution reconstruction, our model demonstrates a more favorable visualization effect compared to the other two models. These results highlight the potential of the Kepler codebook distribution in cross-domain and multi-resolution. The tight and ordered properties of Kepler codebook distribution improve the ability to capture more details for codebook tokens.

D.3. More super-resolution results

We additionally provide downstream super-resolution visualization comparison on DRealSR validation set in Fig. 18. Whether faced with text, buildings, or natural scenes, our models accurately reproduce GT, whereas LDM suffers from significant color aberration, producing artifacts and false details.

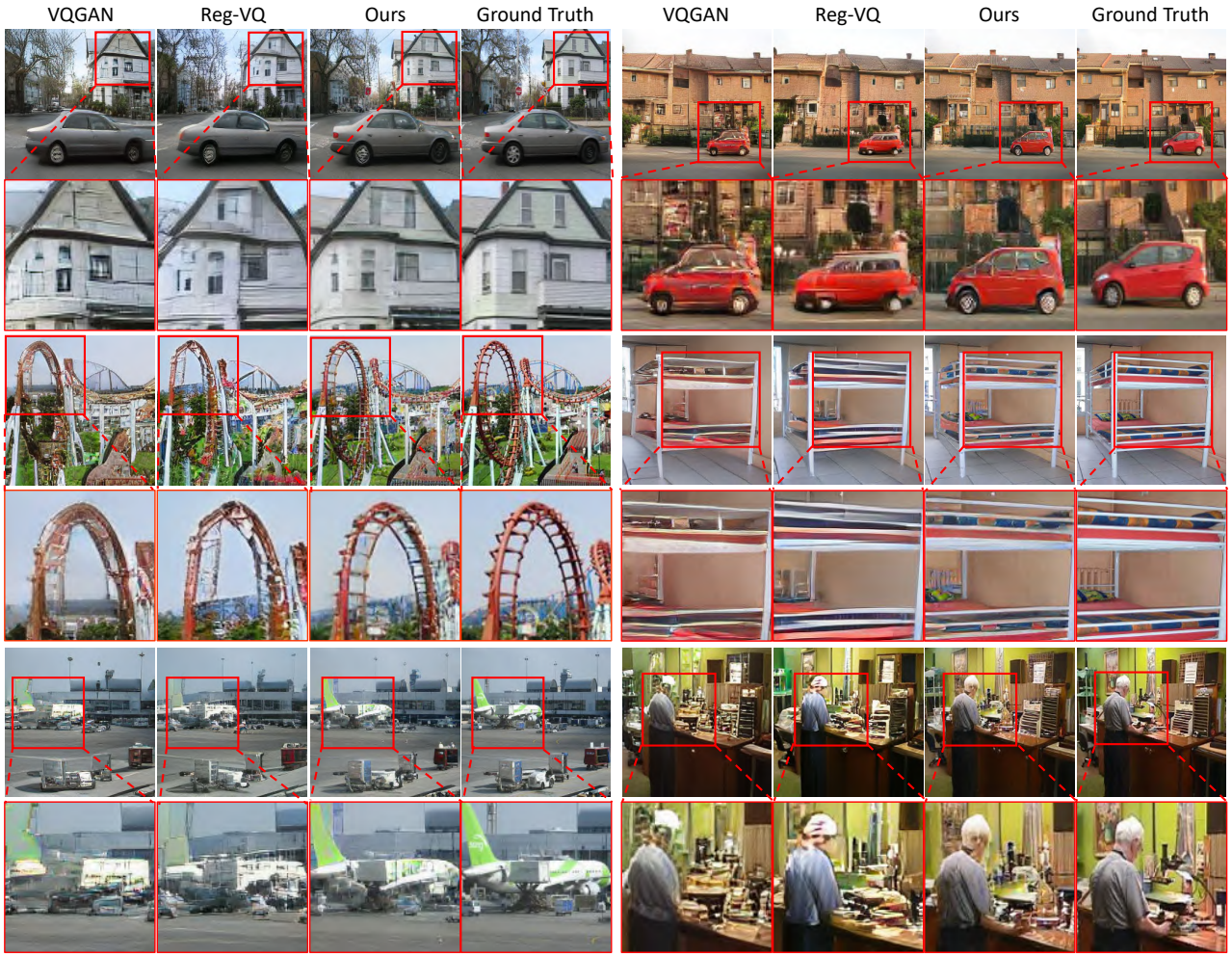

Figure 13. Additional reconstruction results on ADE20K dataset.

Table 8. Multi-resolution cross-domain results on DIV2K validation dataset. We train the three models on ADE20K with 256×256 resolution and test them on five different resolutions from low to high.

Resolution	Method	PSNR↑	SSIM↑	LPIPS↓	rFID↓
127×80	VQGAN	14.14	0.227	0.211	222.94
	Reg-VQ	13.88	0.241	0.211	223.91
	Ours	**14.99**	**0.326**	**0.147**	**195.77**
254×160	VQGAN	15.63	0.304	0.207	197.58
	Reg-VQ	15.51	0.326	0.203	185.31
	Ours	**16.88**	**0.417**	**0.138**	**130.06**
508×320	VQGAN	17.33	0.389	0.182	132.09
	Reg-VQ	17.24	0.406	0.178	129.39
	Ours	**19.05**	**0.514**	**0.117**	**73.73**
1016×640	VQGAN	19.42	0.485	0.148	79.33
	Reg-VQ	19.24	0.498	0.147	77.45
	Ours	**21.73**	**0.614**	**0.094**	**41.28**
2032×1280	VQGAN	21.18	0.552	0.119	50.66
	Reg-VQ	20.84	0.561	0.121	49.16
	Ours	**23.76**	**0.676**	**0.072**	**23.62**

Figure 14. Additional reconstruction results on CelebA-HQ dataset.

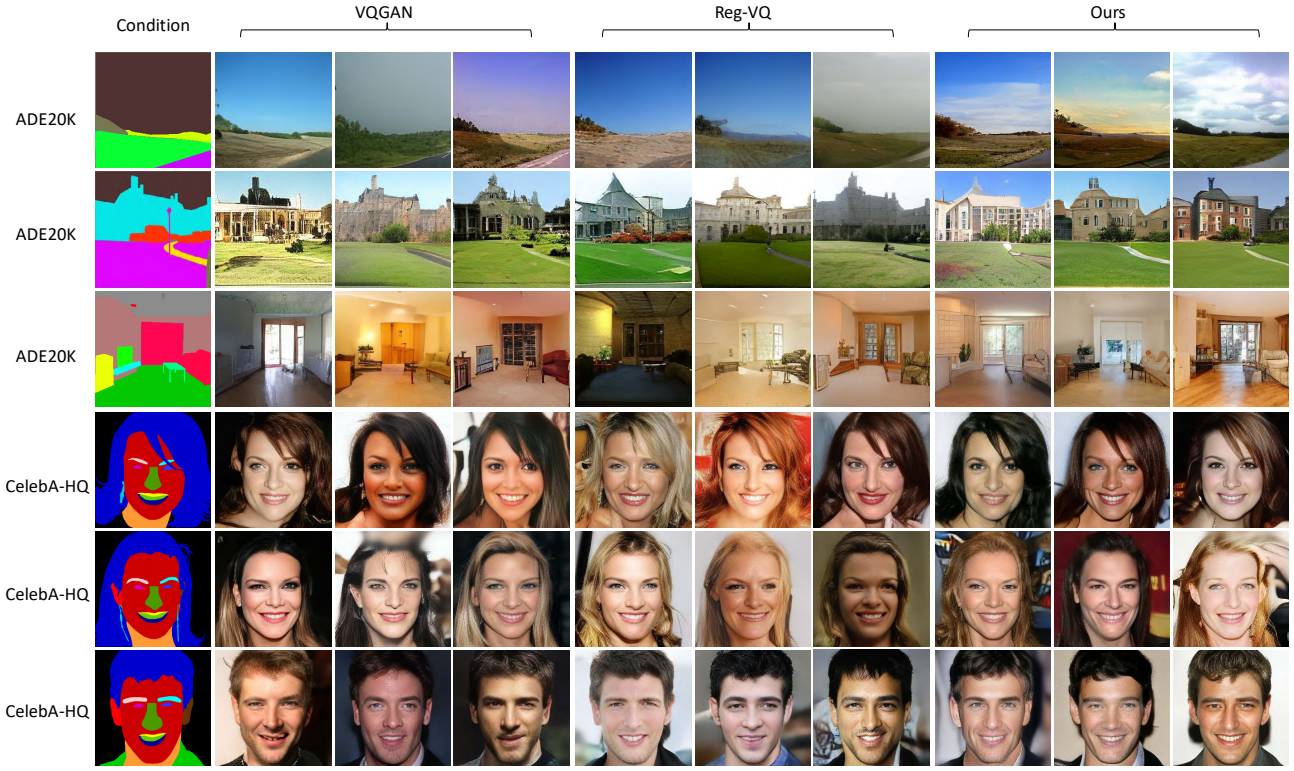

Figure 15. Generation results on ADE20K and CelebA-HQ datasets. The first column is the semantic segmentation map and the subsequent columns show the generated results conditioned on it.

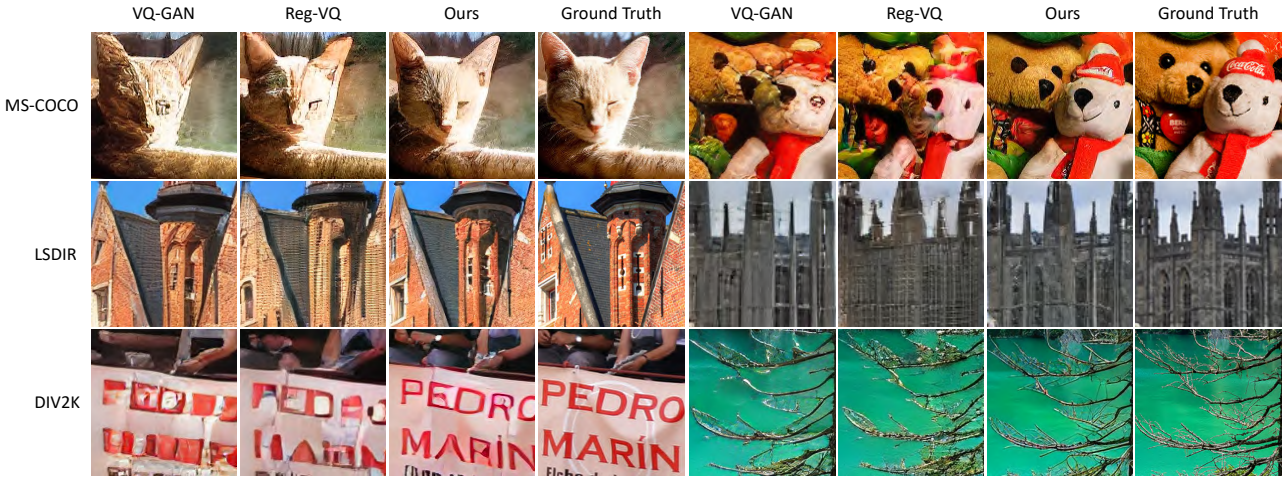

Figure 16. Cross-domain reconstruction results on MS-COCO, LSDIR, and DIV2K datasets.

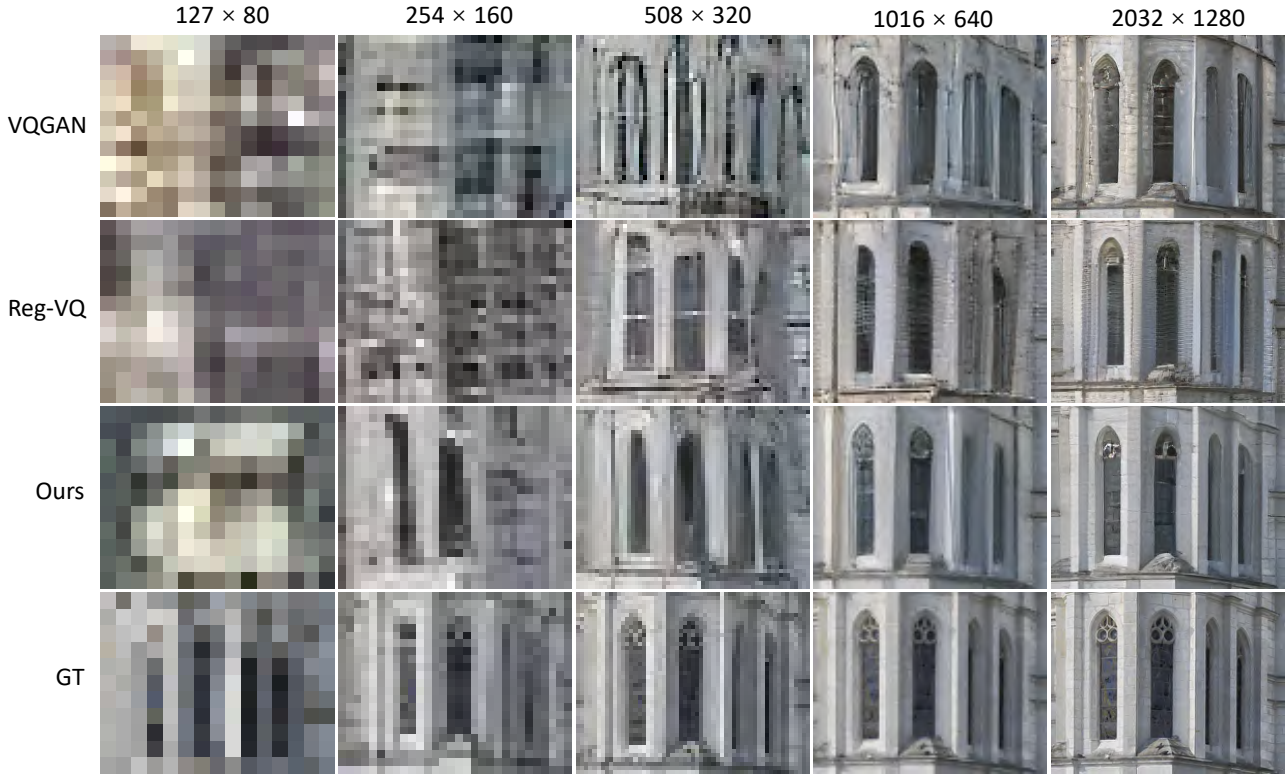

Figure 17. Multi-resolution cross-domain visualization on DIV2K validation set (0865) with five resolutions from high to low. Please zoom in for a better view.

Figure 18. Super-resolution visualization on DRealSR validation set.