A Flat Vocabulary or a Rich Hierarchy? Re-introducing Intrinsic Structure Transforms the Autoregressive Image Generation

Lixuan He [1] **Shikang Zheng** [2]

Abstract

Autoregressive (AR) models have shown great promise in image generation, yet they face a fundamental inefficiency stemming from their core component: a vast, unstructured vocabulary of visual tokens. By treating tokens as a flat set, standard models overlook the manifold structure where geometric proximity reflects semantic similarity. This oversight unnecessarily complicates the prediction task, hindering training efficiency and limiting generation quality. To resolve this, we propose **M**anifold-**A**ligned **S**emantic **C**lustering (MASC), a principled framework that constructs a hierarchical semantic tree directly from the codebook's intrinsic geometry. Utilizing a geometry-aware distance metric and density-driven agglomerative construction, MASC faithfully models the token embedding manifold. By transforming the flat, high-dimensional prediction into a structured hierarchical task, MASC introduces a powerful inductive bias that simplifies learning. Designed as a plug-and-play module, MASC accelerates training by up to 71% and significantly boosts generation quality, improving LlamaGen-XL's FID from 2.87 to 2.49. Crucially, MASC further serves as a convergence enabler for complex architectures. These results establish that structuring the prediction space is as vital as architectural innovation, elevating existing AR frameworks to state-of-the-art performance.

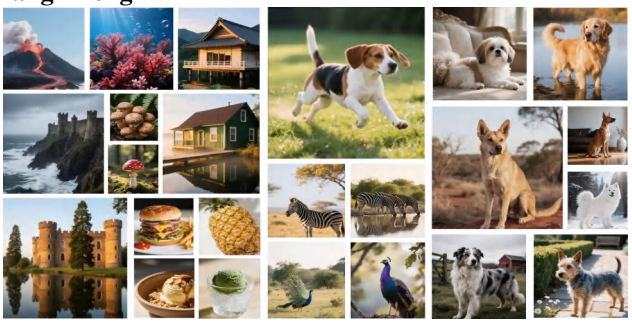

Figure 1. MASC demonstrates strong capabilities in high-fidelity and diverse image generation.

1. Introduction

Autoregressive (AR) image generation (Esser et al., 2021; Sun et al., 2024; Tian et al., 2024; Zhou et al., 2024; Fan et al., 2024; Li et al., 2024) has developed rapidly, demonstrating remarkable scalability similar to that of Large Language Models (LLMs) (Brown et al., 2020; Touvron et al., 2023) and achieving a level of fidelity that rivals leading diffusion-based methods (Peebles & Xie, 2023; Dhariwal & Nichol, 2021; Hatamizadeh et al., 2024). Among various methods, the approach of using discrete tokenizers has attracted widespread attention, not only because they are structurally consistent with the mature LLM paradigm, but also because they offer advantages in training efficiency and ease of optimization. As illustrated in Figure 2, the standard framework for these models consists of two distinct stages: an image tokenizer and an autoregressive causal Transformer. The tokenizer first converts a continuous image into a sequence of discrete integer indices via a learned codebook. Subsequently, the Transformer is trained on these index sequences to causally predict the next token.

The elegance of this paradigm, however, conceals a fundamental drawback stemming from the quantization process: a critical loss of structural information. The AR model is trained on a flat vocabulary of discrete indices, which implicitly treats semantically similar tokens as unrelated entries in a vast categorical space. This process largely disregards the rich geometric relationships embedded within the codebook's high-dimensional vector space. These token embeddings are not randomly scattered; rather, they lie on or near a lower-dimensional semantic manifold (Huh et al., 2023; Van Den Oord et al., 2017), where geometric proximity corresponds to semantic similarity. By ignoring this intrinsic structure, the AR model is forced to tackle a massive N-way (N is the total number of codebook tokens) classification problem at each stepexpending capacity to implicitly re-learn these structures, which leads to sample inefficiency and slow convergence (Ranzato et al., 2015).

Recognizing this challenge, a recent line of work has sought

[1]Tsinghua University [2]Shanghai Jiao Tong University. Correspondence to: Lixuan He <thutsyj23@gmail.com>.

Proceedings of the 43^{rd} International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026.

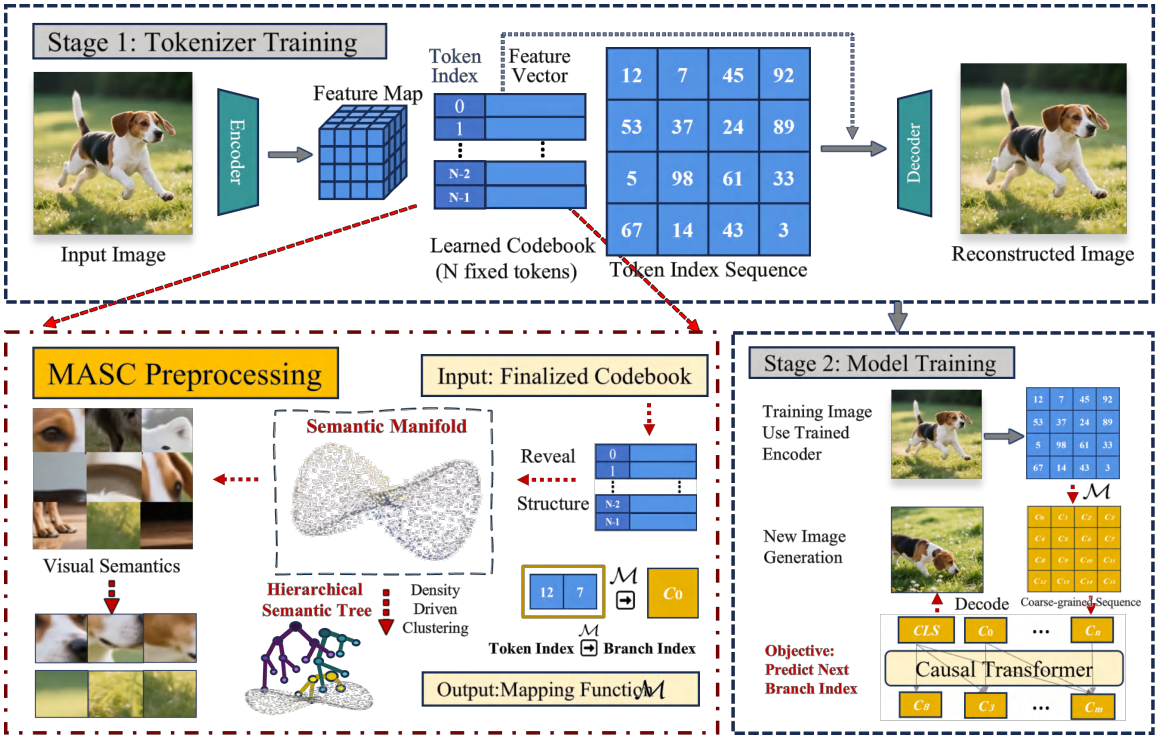

Figure 2. Overview of the MASC-integrated autoregressive generation pipeline, and the bottom row showcases high-fidelity images generated by MASC. **Stage 1:** A standard image tokenizer is trained, yielding a finalized codebook of visual tokens. **MASC Preprocessing:** Our proposed framework takes this codebook and constructs a hierarchical semantic tree, producing a mapping function ($\mathcal{M}$) from fine-grained tokens to coarse-grained semantic branches. **Stage 2:** The transformer is then trained on these simplified branch indices.

to re-introduce this lost structural information by providing the AR model with a codebook prior (Hu et al., 2025; Guo et al., 2025). The common approach is to use k-means clustering to group the token embeddings. However, this strategy is fundamentally limited because k-means is ill-suited to the intrinsic structure of the semantic manifold. Standard Euclidean distance, which k-means relies on, is a poor proxy for true semantic distance on a curved manifold. Cluster centroids, calculated as simple arithmetic means, can fall off the manifold entirely, making them poor representatives of their clusters (Beyer et al., 1999; Huh et al., 2023). Furthermore, the distribution of tokens on this manifold is highly non-uniform, reflecting the statistics of the visual world (Van Den Oord et al., 2017).

In this work, we introduce **M**anifold-**A**ligned **S**emantic **C**lustering (MASC), a framework designed to construct a principled, structure-aware prior that serves as a powerful inductive bias for the AR model. MASC directly confronts the aforementioned challenges with two key innovations: (1) a robust, manifold-aligned similarity metric that is centroid-free, and (2) a density-driven, agglomerative construction process that respects the non-uniform token distribution. The result is a hierarchical semantic tree that transforms the difficult, high-dimensional flat prediction problem into a simpler, structured one, as shown in Figure 3. This unlocks significant gains in both training efficiency and generation quality. Our main contributions are summarized as follows:

[leftmargin=10pt,topsep=0pt]

- We identify structural information loss as an inefficiency in discrete AR models. We show that treating tokens as an unstructured vocabulary ignores the semantic manifold, creating a high-entropy prediction task that hinders scalability and convergence.
- We propose **MASC**, a framework that transforms this flat task into a structured hierarchical prediction. By utilizing a centroid-free, manifold-aligned metric and a density-driven construction, MASC injects a principled inductive bias that models the codebook's intrinsic geometry.
- We provide extensive validation demonstrating that MASC is a universal performance booster. It accelerates training by up to **71%**, elevates generation quality to state-of-the-art levels (e.g., improving LlamaGen-XL FID from 2.87 to **2.49**), and enables convergence for complex architectures that otherwise diverge, proving that structural alignment is essential for generative modeling.

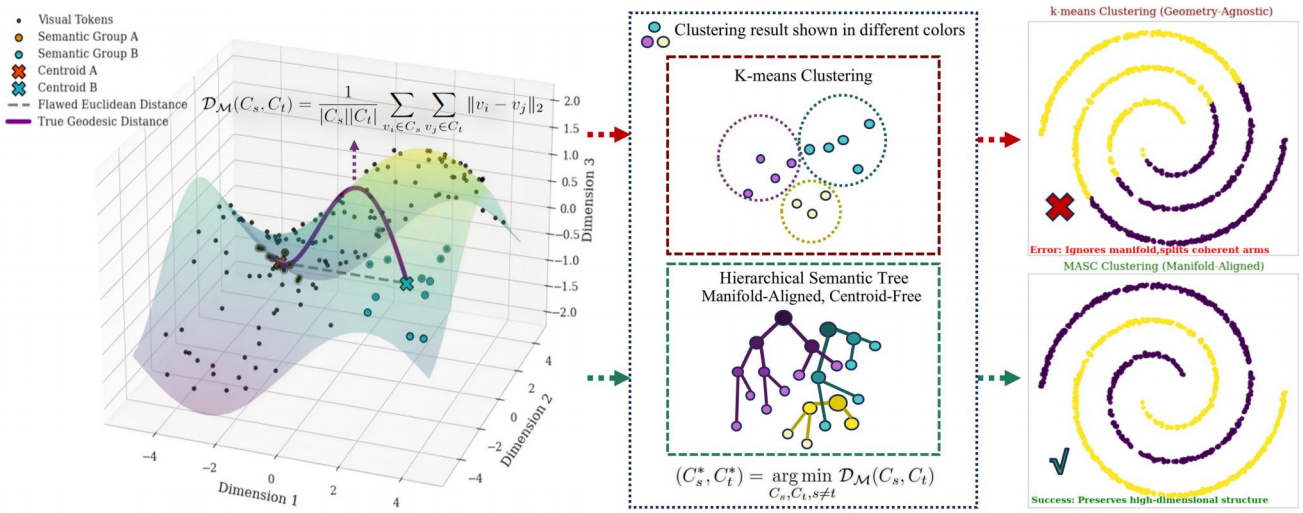

Figure 3. Conceptual illustration of MASC versus k-means. **Left:** Visual tokens reside on a semantic manifold where geodesic distance is a better similarity measure than Euclidean distance (dashed line). **Right:** Geometry-agnostic k-means produces incoherent clusters, whereas MASC's manifold-aligned approach correctly captures the intrinsic data structure and preserves semantic integrity.

2. Related Work

Autoregressive Image Generation. Inspired by Large Language Models (LLMs) (Brown et al., 2020; Touvron et al., 2023), the field of autoregressive (AR) image generation has rapidly matured. Its core paradigm uses a tokenizer (Esser et al., 2021; Yu et al., 2021) to convert images into discrete sequences for a Transformer model, an approach that has proven highly scalable and achieved quality competitive with leading diffusion models (Dhariwal & Nichol, 2021; Peebles & Xie, 2023). A wave of architectural innovations has advanced the field, from adapting LLM designs (Sun et al., 2024) and exploring diverse prediction schemes (Tian et al., 2024; Zhou et al., 2024; Han et al., 2024; Fan et al., 2024), to developing new decoding, refinement, and alignment strategies (Zhang et al., 2025; Cheng et al., 2026; Wu et al., 2025). Despite these significant advancements, a unifying challenge persists: discrete AR models perform prediction over a large, flat, and unstructured vocabulary.

Codebook Priors. Pioneering methods have sought to extract a codebook prior. The predominant approach has been to apply k-means clustering to the token embeddings (Hu et al., 2025; Guo et al., 2025). However, the visual token space is better modeled as a non-uniform semantic manifold (Van Den Oord et al., 2017; Huh et al., 2023), where k-means' core assumptions about Euclidean space and meaningful centroids are violated. This mismatch often results in semantically incoherent clusters and a noisy prior (Beyer et al., 1999). Other works on codebook manipulation focus on improving the tokenizer's representational (Li et al., 2023; Zheng & Vedaldi, 2023) rather than providing a structural prior.

Manifold Learning and Hierarchical Clustering. Our approach is grounded in two established principles. First, Manifold learning posits that token embeddings lie on a low-dimensional manifold (Tenenbaum et al., 2000; Belkin & Niyogi, 2003), where standard Euclidean distance is an unreliable metric, thus necessitating geometry-aware methods (Beyer et al., 1999; Angiulli, 2018). Second, hierarchical clustering (Lukasová, 1979; Ward Jr, 1963) is naturally suited for the non-uniform density of such data. MASC operationalizes a synthesis of these principles, employing a geometry-aware metric within a density-driven algorithm.

3. Methodology

3.1. Preliminaries: The High-Dimensional Prediction Challenge

Discrete token-based autoregressive image generation frameworks (Esser et al., 2021; Sun et al., 2024; Tian et al., 2024) operate via a two-stage process. First, an image tokenizer maps a continuous image $x \in \mathbb{R}^{H \times W \times 3}$ into a sequence of discrete integer indices. An encoder E produces a feature map, and each feature vector $z^{(i,j)}$ is quantized to its nearest token in a learnable codebook $\mathcal{Z} = \{v_1, \ldots, v_N\} \subset \mathbb{R}^d$ via a nearest-neighbor lookup:

$$q^{(i,j)} = _{k \in \{1,\ldots,N\}} \|z^{(i,j)} - v_k\|_2, \tag{1}$$

resulting in a grid of token indices $z^q \in \{1, \ldots, N\}^{h \times w}$, which is flattened into a 1D sequence of length $L = h \times w$.

In the second stage, an autoregressive model $\mathcal{G}$, typically a Transformer, is trained to maximize the likelihood of this token sequence, predicting the next token z_t^q based on the

preceding context $z^q_{<t}$:

$$p(z^q) = \prod_{t=1}^{L} p(z^q_t | z^q_{<t}; \mathcal{G}). \quad (2)$$

This is practically realized by an output layer producing a probability distribution over all N tokens. The core difficulty, which we term the high-dimensional prediction challenge, stems from this process. The model is forced to perform a massive N-way classification at every step, treating the vocabulary as a flat, unstructured set of categories. This quantization step discards the rich geometric structure of the codebook, where semantic relationships are encoded as distances. Without this structural information, the model must expend significant capacity learning the semantic landscape from scratch. A codebook prior, therefore, aims to re-introduce this latent structure, providing an inductive bias to make the prediction task more tractable and efficient.

3.2. MASC: Constructing a Structure-Aware Prior

To resolve the high-dimensional prediction challenge, we introduce **M**anifold-**A**ligned **S**emantic **C**lustering (MASC), a framework designed to construct a structure-aware prior from the codebook. MASC builds a hierarchical semantic tree that models the intrinsic geometric and statistical properties of the token embeddings, providing a powerful and meaningful inductive bias for the AR model. The construction process is composed of two core components: a principled similarity metric and a density-driven clustering algorithm.

3.2.1. A Principled Distance for a Semantic Manifold

A principled prior must be built upon an accurate measure of semantic similarity. As previously discussed, the token embeddings $\{v_i\}$ lie on or near a complex semantic manifold embedded within a high-dimensional space. A naive approach like k-means, which relies on Euclidean distance to cluster centroids, is fundamentally flawed in this context for two primary reasons. First, the arithmetic mean used to compute a centroid is a point in the ambient Euclidean space, which may not lie on the manifold itself. Using such an off-manifold point as a representative for a group of on-manifold tokens is geometrically unsound and can lead to inaccurate cluster assignments (Huh et al., 2023). Second, even if the centroid were valid, standard Euclidean distance is a poor proxy for true semantic similarity in a high-dimensional, curved space, which is more accurately described by the path length along the manifold's surface (Beyer et al., 1999; Tenenbaum et al., 2000).

While computing the true geodesic distance is computationally intractable for an implicitly defined manifold, we can instead devise a practical metric that respects the manifold's properties. Instead of relying on a single, potentially invalid centroid, we employ a robust, **instance-based average distance** to measure the dissimilarity between two clusters, C_s and C_t. This metric aggregates the pairwise Euclidean distances between all tokens across the two clusters:

$$\mathcal{D}(C_s, C_t) = \frac{1}{|C_s||C_t|} \sum_{v_i \in C_s} \sum_{v_j \in C_t} \|v_i - v_j\|_2. \quad (3)$$

This metric is inherently manifold-aligned because it is centroid-free, relying exclusively on the locations of the actual tokens which are guaranteed to lie on the manifold. By averaging over all pairs, it provides a robust measure of inter-cluster separation that implicitly respects the local geometry and is less sensitive to outliers, serving as a sound basis for constructing a meaningful semantic hierarchy.

3.2.2. Density-Driven Hierarchical Construction

Equipped with our similarity metric $\mathcal{D}$, we construct the semantic tree using a bottom-up, agglomerative hierarchical strategy (Lukasová, 1979; Ward Jr, 1963). This choice is crucial as the algorithm is deterministic and inherently respects the non-uniform density distribution of tokens on the manifold. The construction process proceeds as follows:

Initialization: The process begins with N initial clusters, where each cluster $C_j^{(0)}$ contains a single token from the codebook: $C_j^{(0)} = \{v_j\}$ for $j \in \{1, \ldots, N\}$. These form the leaf nodes of our tree.

Iterative Merging: The algorithm then performs $N - 1$ merging iterations. In each iteration i, it identifies the pair of distinct clusters (C_s^*, C_t^*) from the current set of clus-

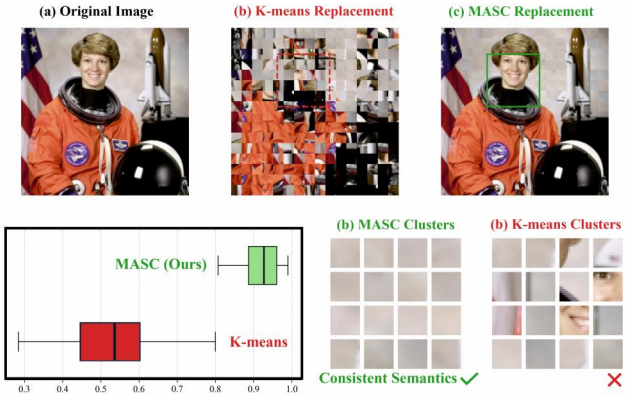

Figure 4. **Analysis of Semantic Coherence. (a-c) Semantic Replacement Test:** Replacing image tokens with random cluster members reveals that MASC (c) preserves semantic structure, whereas K-means (b) introduces artifacts. **(d)** MASC yields higher intra-cluster semantic similarity. **(e-f)** Visualizations confirm MASC groups coherent textures (e), while K-means mixes unrelated semantics based on color (f).

Figure 5. Visual comparison between K-means and MASC. K-means results exhibit high-frequency noise and structural collapse.

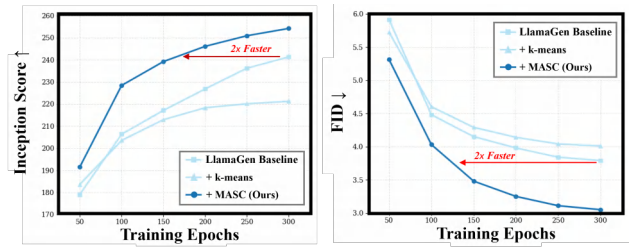

Figure 6. Training dynamics of LlamaGen-L on ImageNet. The plots compare the IS and FID scores over training epochs for the baseline, the + k-means variant, and our + MASC method. The MASC-enhanced model reaches the baseline's final performance in approximately half the training time, and ultimately converging to a better result.

ters $\mathbb{C}^{(i-1)}$ that are most similar to each other, and merges them to form a new parent cluster. The merging criterion is formally expressed as:

$$(C_s^*, C_t^*) = _{C_s, C_t \in \mathbb{C}^{(i-1)}, s \neq t} \mathcal{D}(C_s, C_t). \tag{4}$$

The new set of clusters for the next iteration, $\mathbb{C}^{(i)}$, is then formed by replacing C_s^* and C_t^* with their union, $C_s^* \cup C_t^*$. This merging process is repeated until the root of the MASC tree remains, which encompasses all N original tokens.

The critical insight is that this bottom-up construction is implicitly **density-driven**. High-density regions on the semantic manifold correspond to areas where many tokens with similar visual semantics are tightly packed. Consequently, the pairwise distances $\mathcal{D}$ between tokens or clusters within these regions will be small. Our algorithm, by always merging the closest pair according to Eq. 4, naturally prioritizes forming connections within these dense, semantically coherent regions first. This property avoids the critical flaw of methods like k-means, which can carelessly partition dense regions, and guarantees that the most fundamental semantic similarities are faithfully captured in the hierarchy.

3.3. Integrating the MASC Prior with Autoregressive Models

The MASC-generated tree provides a multi-level structural prior of the codebook. To integrate this prior into the autoregressive framework, we primarily use it to perform a principled form of **vocabulary reduction**. The full semantic tree is cut at a level that yields k distinct branches, where $k < N$ is a hyperparameter. Each of these branches represents a semantically coherent cluster of fine-grained tokens. We then create a mapping $\mathcal{M} : \{1, \ldots, N\} \to \{1, \ldots, k\}$ that assigns each original token index to its corresponding coarse-grained cluster index.

This mapping provides a powerful inductive bias by transforming the model's learning objective. During training, the ground-truth sequences of fine-grained indices z^q are converted into sequences of coarse-grained cluster indices $z^b = \mathcal{M}(z^q)$. The autoregressive model $\mathcal{G}$ is then trained to predict the next cluster index, rather than the next token index:

$$p(z^b) = \prod_{t=1}^{L} p(z_t^b | z_{<t}^b; \mathcal{G}). \tag{5}$$

This fundamentally changes the prediction task from a flat N-way classification to a structured k-way classification. Since each target class now represents a semantically meaningful region of the original embedding space, the learning task is significantly simplified, allowing the model to learn the high-level structure of images more efficiently.

During inference, the AR model generates a sequence of coarse cluster indices z^b, which must be decoded back into a sequence of specific tokens z^q for the decoder to render an image. We explore two strategies for this decoding step:

Random Sampling (Default Strategy). This default strategy leverages the high intra-cluster semantic consistency guaranteed by the MASC construction. For each predicted coarse index z_t^b, we randomly and uniformly sample a fine-grained token from the corresponding cluster $\{v_i | \mathcal{M}(i) = z_t^b\}$. Because all tokens within a MASC-generated cluster are semantically similar, any choice is a reasonable representative, leading to high-quality results with a simple and efficient decoding step.

Hierarchical Decoding (Extended Strategy). For potentially higher fidelity, we explore a two-stage hierarchical decoding approach, inspired by coarse-to-fine pipelines (Guo et al., 2025). In this setup, a small, secondary refinement network, $\mathcal{G}_{\text{refine}}$, is trained. After the main model $\mathcal{G}$ predicts a coarse index z_t^b, $\mathcal{G}_{\text{refine}}$ takes the hidden state from $\mathcal{G}$ (as context) and z_t^b as input. Its task is to then predict the final token by computing a probability distribution only over the

Table 1. Comprehensive performance analysis on ImageNet 256 × 256. We compare **LlamaGen Baseline**, a naive prior with **+ k-means**, and our **+ MASC** method across three model scales. MASC consistently improves efficiency, reduces the prediction task complexity, and yields superior generation quality. Using a vocabulary of k = 8,192

Model	Method	Model Efficiency		Generation Quality		
		Params	Accel.↑	FID ↓	IS ↑	Precision / Recall ↑
LlamaGen-B (111M)	LlamaGen Baseline	111M	-	5.56	185.1	0.85 / 0.45
	+ k-means	94M	-5%	5.72 (+2.9%)	186.6 (+0.8%)	0.83 (-2.4%) / 0.44 (-2.2%)
	+ MASC (Ours)	94M	**+42%**	**4.81** (-13.5%)	**206.4** (+11.5%)	**0.86** (+1.2%) / **0.48** (+6.7%)
LlamaGen-L (343M)	LlamaGen Baseline	343M	-	3.38	246.3	0.83 / 0.51
	+ k-means	311M	-10%	3.81 (+12.7%)	221.2 (-10.2%)	0.77 (-7.2%) / 0.55 (+7.8%)
	+ MASC (Ours)	311M	**+55%**	**2.92** (-13.6%)	**259.2** (+5.2%)	**0.85** (+2.4%) / **0.57** (+11.8%)
LlamaGen-XL (775M)	LlamaGen Baseline	775M	-	2.87	267.6	0.82 / 0.55
	+ k-means	719M	-8%	3.09 (+7.7%)	244.3 (-8.7%)	0.76 (-7.3%) / 0.56 (+1.8%)
	+ MASC (Ours)	719M	**+57%**	**2.58** (-10.1%)	**272.1** (+1.7%)	**0.83** (+1.2%) / **0.58** (+5.5%)

Table 2. **Ablation on vocabulary size (k)** with LlamaGen-XL

Method	Vocab (k)	FID↓	IS↑	Accel.
Baseline	16,384	2.87	267.6	-
MASC	8,192	2.58 (-10.10%)	**272.1** (+1.68%)	+57%
MASC	**4,096**	**2.49** (-13.24%)	269.8 (+0.82%)	**+63%**
MASC	2,048	2.65 (-7.66%)	264.4 (-1.20%)	+71%

tokens within the constrained subspace of the predicted cluster. This allows the model to first identify a semantically correct region and then make a more precise, fine-grained choice within it. The complete MASC construction process, which serves as a one-time offline preprocessing step, is detailed in Algorithm 1.

3.4. Empirical Verification of Semantic Coherence

To empirically validate that our manifold-aligned clusters capture high-level semantics rather than low-level color statistics, we present a Semantic Replacement Test in Figure 4. By randomly replacing image tokens with other candidates from the same cluster, we observe that MASC successfully preserves global structure and object identity, whereas geometry-agnostic k-means leads to severe semantic collapse and artifacts. This qualitative superiority is further corroborated by the high textural coherence of MASC cluster assignments (Figure 4e) and a significantly higher intra-cluster semantic similarity score measured by DINOv2. We refer readers to Appendix B for the detailed experimental setup and extended analysis.

Accelerated Convergence. The simplified learning task naturally leads to a more efficient training process. As shown in Table 1 and visualized in Figure 8, MASC-enhanced models converge significantly faster by learning from a more structured prediction target, achieving training accelerations of up to **+57%** on the LlamaGen-XL model. Conversely, the poorly structured prior from k-means impedes learning, resulting in negative acceleration.

4. Experiments

4.1. Experimental Setup

Dataset. All experiments are conducted on the **ImageNet-1K** benchmark (Deng et al., 2009) for class-conditional generation, with images resized to 256 × 256 pixels following standard practice (Sun et al., 2024; Tian et al., 2024).

Evaluation Metrics. For **Generation Quality**, we report FID (Heusel et al., 2018), IS (Salimans et al., 2016), Precision, and Recall (Kynkäänniemi et al., 2019). And we evaluate **Training Efficiency** via Training Acceleration, the percentage of epochs saved to reach the baseline's final FID.

4.2. Main Validation: MASC on LlamaGen

Backbone Models and Baselines. We use the **LlamaGen** (B, L, XL) family (Sun et al., 2024) for a controlled analysis. We compare three variants: the original **LlamaGen Baseline** with a flat vocabulary of $N = 16,384$ tokens; a **+ k-means** baseline using a vocabulary of $k = 8,192$ clusters from standard k-means; and our proposed **+ MASC** method, trained on a $k = 8,192$ (default) vocabulary derived from the MASC tree.

The core results of our study, presented in Table 1, demonstrate that MASC provides a more effective codebook prior than k-means, leading to measurable gains in task simplification, training efficiency, and final generation quality across all model scales.

Analysis of Prediction Uncertainty. The entropy metrics in Table 1 provide direct evidence that MASC simplifies the learning problem. Unlike the k-means prior, which fails to reduce relative uncertainty, MASC enables significantly more confident predictions, substantially reducing the Normalized Prediction Entropy across all scales.

Enhanced Generation Performance. The structured prediction space also translates to improved final generation

Table 3. Performance of MASC-enhanced frameworks against the SOTA AR model on ImageNet. We evaluate MASC as a plug-and-play module alongside **k-means** and **Random Clustering** controls to isolate the benefits of manifold-aligned structure.

Method	Params	Accel.	Wall-clock (h)	FID↓	IS↑
ViT-VQGAN (Yu et al., 2021)	1.7B	-	-	4.17	175.1
Transfusion (Zhou et al., 2024)	700M	-	-	2.45	288.5
Infinity (Han et al., 2024)	2B	-	-	2.39	291.0
VAR-d24 (Tian et al., 2024) (Baseline)	1.0B	-	520	2.09	312.9
+ Random (2k)	1.0B	+7%	483	3.84 (+83%)	288.5 (-7.8%)
+ k-means (2k)	1.0B	-12%	582	2.56 (+22%)	292.4 (-6.5%)
+ MASC (2k)	1.0B	**+28%**	**374**	**1.98 (-5.26%)**	311.4 (-0.48%)
+ MASC (1k)	1.0B	+39%	317	2.24 (+7.18%)	303.8 (-2.91%)
RandAR-XL (Pang et al., 2025) (Baseline)	775M	-	480	2.25	317.8
+ Random (8k)	775M	+4%	461	4.12 (+83%)	282.4 (-11.1%)
+ k-means (8k)	775M	-8%	518	2.62 (+16%)	298.1 (-6.2%)
+ MASC (8k)	775M	**+32%**	**326**	**2.02 (-10.22%)**	**320.5 (+0.85%)**
+ MASC (4k)	775M	+41%	283	2.13 (-5.33%)	318.2 (+0.13%)
+ MASC (2k)	775M	+49%	245	2.42 (+7.56%)	305.1 (-4.00%)
CTF-LlamaGen-L (Guo et al., 2025) (Baseline)	653M	-	360	2.97	291.5
+ Random (8k)	653M	+6%	340	4.65 (+56.6%)	265.4 (-9.0%)
+ k-means (8k)	653M	-10%	396	3.42 (+15.2%)	278.2 (-4.6%)
+ MASC (8k)	653M	**+29%**	**255**	**2.51 (-15.49%)**	**296.9 (+1.85%)**
IAR-B (Hu et al., 2025) (Baseline)	111M	-	152	5.14	202.0
+ Random (8k)	111M	+5%	144	6.82 (+32.7%)	175.4 (-13.2%)
+ MASC (8k)	111M	**+21%**	**118**	**4.96 (-3.50%)**	**214.3 (+6.09%)**
IAR-L (Hu et al., 2025) (Baseline)	343M	-	296	3.18	234.8
+ Random (8k)	343M	+5%	282	4.02 (+26.4%)	215.1 (-8.4%)
+ MASC (8k)	343M	**+25%**	**225**	**2.97 (-6.60%)**	**267.3 (+13.84%)**
IAR-XL (Hu et al., 2025) (Baseline)	775M	-	467	2.52	248.1
+ Random (8k)	775M	+5%	445	3.25 (+29.0%)	235.4 (-5.1%)
+ MASC (8k)	775M	**+32%**	**340**	**2.35 (-6.75%)**	**284.3 (+14.59%)**
Generalization on GigaTok-L (Xiong et al., 2025)					
GigaTok-L (Baseline)	16k Voc.	-	420	3.39	263.7
+ Random (8k)	8k Voc.	+3%	407	5.14 (+51%)	232.1 (-12%)
+ k-means (8k)	8k Voc.	-5%	441	3.67 (+8%)	259.8 (-1.5%)
+ MASC (8k)	8k Voc.	**+19%**	**340**	**3.35 (-1.18%)**	263.5 (-0.08%)
+ MASC (4k)	4k Voc.	+42%	243	3.86 (+13.86%)	256.4 (-2.77%)
+ MASC (2k)	2k Voc.	+51%	215	4.02 (+18.58%)	256.4 (-5.88%)

quality. By easing the learning burden, MASC allows the model to better capture the data distribution. Consequently, MASC-enhanced models consistently and significantly outperform both the vanilla and k-means baselines across all key metrics. For instance, on the flagship LlamaGen-XL model, MASC improves the FID from 2.87 to **2.58**, while also boosting both IS and recall. Furthermore, an analysis of the precision-recall metrics reveals that MASC tends to improve Recall more substantially than Precision. We attribute this to the simplified task of predicting a coherent semantic branch rather than a single token; this reduces learning pressure and allows the model to better capture the full diversity of the training data.

Impact of Vocabulary Size (k). We further investigate the sensitivity of MASC to the cluster count k, which dictates the granularity of the structured prior. While our main experiments utilize $k = 8,192$ for parity with baselines, results in Table 2 reveal that MASC demonstrates remarkable robustness at lower resolutions. MASC robustly achieves its peak FID of **2.49** at $k = 4,096$, simultaneously boosting

Table 4. Performance of MASC-enhanced frameworks against the SOTA AR model on ImageNet. We evaluate MASC as a plug-and-play module alongside **k-means** and **Random Clustering** controls to isolate the benefits of manifold-aligned structure.

Method	Params	Accel.	Wall-clock (h)	FID↓	IS↑
IAR-L (Hu et al., 2025) (Baseline)	343M	-	296	3.18	234.8
+ Random (8k)	343M	+5%	282	4.02 (+26.4%)	215.1 (-8.4%)
+ MASC (8k)	343M	**+25%**	**225**	**2.97 (-6.60%)**	**267.3 (+13.84%)**
Generalization on GigaTok-L (Xiong et al., 2025)					
GigaTok-L (Baseline)	16k Voc.	-	420	3.39	263.7
+ Random (8k)	8k Voc.	+3%	407	5.14 (+51%)	232.1 (-12%)
+ k-means (8k)	8k Voc.	-5%	441	3.67 (+8%)	259.8 (-1.5%)
+ MASC (8k)	8k Voc.	**+19%**	**340**	**3.35 (-1.18%)**	263.5 (-0.08%)
+ MASC (4k)	4k Voc.	+42%	243	3.86 (+13.86%)	256.4 (-2.77%)
+ MASC (2k)	2k Voc.	+51%	215	4.02 (+18.58%)	256.4 (-5.88%)

Table 5. MASC acts as a convergence enabler for Advanced AR Architectures. We evaluate MASC on **RAR** framework (Yu et al., 2024). While RAR **fails to converge** when trained directly with large-vocabulary tokenizers (LlamaGen/GigaTok, 16k tokens) due to the immense search space, MASC enables stable training.

Setup	Config (Vocab)	FID↓	IS↑
Reference: RAR-L + MaskGIT		1.70	299.5
RAR-L + LlamaGen	None (Original)	N/A (Diverged)	N/A (Diverged)
	MASC (2k)	1.67	300.5
	MASC (4k)	**1.61**	**304.1**
RAR-L + GigaTok	None (Original)	N/A (Diverged)	N/A (Diverged)
	MASC (2k)	1.64	302.8
	MASC (4k)	**1.57**	**307.4**

training acceleration to **+63%**.

4.3. Generality and Application Extensions

We validate MASC's versatility and robustness beyond the primary LlamaGen setup. The results in Table 4 and Table 5 confirm its effectiveness as a general-purpose, plug-and-play module across diverse autoregressive paradigms.

Boosting Diverse AR Architectures. By re-introducing intrinsic structure to the codebook, MASC significantly boosts VAR (Tian et al., 2024), RandAR (Pang et al., 2025), IAR (Hu et al., 2025), and CTF (Guo et al., 2025).

Enabling Convergence for Complex Objectives. Crucially, MASC serves as a convergence enabler for advanced architectures with immense search spaces. As detailed in Table 5, the state-of-the-art RAR framework (Yu et al., 2024), which employs random permutation objectives, fails to converge when trained directly on large-vocabulary tokenizers (16k) due to the unstructured prediction task.

Mechanism Behind RAR Stabilization. To further understand this effect, we compare different 4k target spaces under the same RAR setup in Table 6. Merely reducing the output space is insufficient: random clustering and k-means still lead to high prediction uncertainty and unstable gradients. In contrast, MASC produces semantically coherent coarse targets, substantially reducing both normalized prediction entropy and gradient-norm variation. This suggests that the convergence benefit comes from the combination of a smaller search space and a more structured semantic target space, rather than vocabulary reduction alone.

Table 6. **Mechanism analysis for RAR convergence.** Under the same 4k target size, MASC yields lower prediction entropy, lower gradient variation, and better FID, showing that semantic structure matters beyond vocabulary reduction.

Target Space	Norm. Entropy↓	Grad. CV↓	FID↓
Random clustering (4k)	0.84	1.33	4.91
k-means clustering (4k)	0.79	1.07	4.24
MASC (4k)	**0.63**	**0.61**	**1.61**

Synergy with Strong Tokenizers. We further confirm MASC's robustness when applied to high-capacity tokenizers like GigaTok-L (Xiong et al., 2025). Results in Table 4 show that MASC is not merely a fix for suboptimal codebooks; even with GigaTok, MASC ($k = 8,192$) improves FID to **3.35**. This indicates that MASC's geometric clustering captures high-level semantic structures that are complementary to the tokenizer's representational quality.

Efficiency Gains. A unifying advantage of MASC is training acceleration across all evaluated frameworks. By transforming the flat prediction task into a simplified structured one, MASC reduces the learning burden, yielding accelerations of **+28%** for VAR and **+32%** for RandAR.

4.4. Ablation Study about Clustering Strategy

To isolate the contributions of our two core innovations, we conduct an ablation study presented in Table 7. The results

Table 7. Expanded ablation study of MASC's core components. This table ablates the Clustering Strategy to dissect the contributions of our key innovations: the Manifold Distance and Bottom-Up Construction.

Model	Clustering Strategy	FID↓	IS↑
LlamaGen-B (111M)	k-means++	5.41	187.9
	MASC w/ Centroid	5.17 (-4.44%)	196.2 (+4.42%)
	Full MASC	**4.81** (-11.09%)	**206.4** (+9.85%)
LlamaGen-L (343M)	k-means++	3.85	223.7
	MASC w/ Centroid	3.68 (-4.42%)	246.5 (+10.19%)
	Full MASC	**2.92** (-24.16%)	**259.2** (+15.87%)
LlamaGen-XL (775M)	k-means++	3.17	246.5
	MASC w/ Centroid	3.02 (-4.73%)	268.1 (+8.76%)
	Full MASC	**2.58** (-18.61%)	**272.1** (+10.39%)

reveal a clear performance hierarchy across all model scales: the density-driven construction alone (MASC w/ Centroid) consistently outperforms the stronger k-means++ (Arthur & Vassilvitskii, 2007) baseline. This confirms that both the density-driven (bottom-up) construction and the manifold-aligned similarity metric are critical, synergistic components, essential to addressing the geometric and density properties of the codebook.

5. Conclusion and Discussion

In this work, we address the inefficiency of the flat vocabulary in autoregressive image generation. We introduce MASC, a plug-and-play framework that constructs a density-driven, hierarchical prior directly from the codebook's geometry, thereby transforming the high-entropy task into a structured prediction problem. Our evaluation demonstrates that MASC is a universal booster which can adapt to various tokenizers, establishing a new method for efficient and scalable generative modeling.

Beyond immediate performance gains, our findings reveal a critical insight: structuring the output space is not merely an optimization trick but a prerequisite for scaling advanced autoregressive paradigms. Notably, we discover that MASC serves as a convergence enabler for complex architectures like RAR (Yu et al., 2024); by organizing the immense search space of large vocabularies, MASC renders the training of random-permutation models tractable where they otherwise diverge. This implies that as the field moves towards larger codebooks and non-raster generation orders, the structural alignment between the tokenizer and the autoregressive model becomes as crucial as architectural innovation. Furthermore, MASC points out an important avenue for geometry-centric generative modeling, where the efficiency of learning is governed by how faithfully the discrete prediction hierarchy reflects the latent manifold density. Future research may explore context-aware decoders to further exploit this hierarchy or develop dynamic priors that adaptively reshape the prediction space during inference.

Acknowledgements

We are grateful to Linfeng Zhang for his helpful advice and support. We also thank the reviewers for their constructive feedback, which helped improve the quality of this work.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Angiulli, F. On the behavior of intrinsically high-dimensional spaces: distances, direct and reverse nearest neighbors, and hubness. *Journal of Machine Learning Research*, 18(170):1–60, 2018.

Arthur, D. and Vassilvitskii, S. k-means++: The advantages of careful seeding. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 1027–1035. Society for Industrial and Applied Mathematics, 2007.

Belkin, M. and Niyogi, P. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Beyer, K., Goldstein, J., Ramakrishnan, R., and Shaft, U. When is "nearest neighbor" meaningful? In *Database Theory—ICDT'99: 7th International Conference Jerusalem, Israel, January 10–12, 1999 Proceedings 7*, pp. 217–235. Springer, 1999.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Cheng, C., Song, L., An, D., Xiao, Y., Zhang, X., Sun, H., and Shan, Y. From prediction to perfection: Introducing refinement to autoregressive image generation, 2026. URL https://arxiv.org/abs/2505.16324.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Dhariwal, P. and Nichol, A. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

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.

Fan, L., Li, T., Qin, S., Li, Y., Sun, C., Rubinstein, M., Sun, D., He, K., and Tian, Y. Fluid: Scaling autoregressive text-to-image generative models with continuous tokens. *arXiv preprint arXiv:2410.13863*, 2024.

Guo, Z., Zhang, K., and Shieh, M. Q. Improving autoregressive image generation through coarse-to-fine token prediction. *arXiv preprint arXiv:2503.16194*, 2025.

Han, J., Liu, J., Jiang, Y., Yan, B., Zhang, Y., Yuan, Z., Peng, B., and Liu, X. Infinity: Scaling bitwise autoregressive modeling for high-resolution image synthesis. *arXiv preprint arXiv:2412.04431*, 2024.

Hatamizadeh, A., Song, J., Liu, G., Kautz, J., and Vahdat, A. Diffit: Diffusion vision transformers for image generation, 2024. URL https://arxiv.org/abs/2312.02139.

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, 2018. URL https://arxiv.org/abs/1706.08500.

Hu, T., Zhang, J., Yi, R., Weng, J., Wang, Y., Zeng, X., Xue, Z., and Ma, L. Improving autoregressive visual generation with cluster-oriented token prediction. *arXiv preprint arXiv:2501.00880*, 2025.

Huh, M., Cheung, B., Agrawal, P., and Isola, P. Straightening out the straight-through estimator: Overcoming optimization challenges in vector quantized networks. In *International Conference on Machine Learning*, pp. 14096–14113. PMLR, 2023.

Kynkäänniemi, T., Karras, T., Laine, S., Lehtinen, J., and Aila, T. Improved precision and recall metric for assessing generative models, 2019. URL https://arxiv.org/abs/1904.06991.

Li, L., Liu, T., Wang, C., Qiu, M., Chen, C., Gao, M., and Zhou, A. Resizing codebook of vector quantization without retraining. *Multimedia Systems*, 29(3):1499–1512, 2023.

Li, T., Tian, Y., Li, H., Deng, M., and He, K. Autoregressive image generation without vector quantization. *Advances in Neural Information Processing Systems*, 37:56424–56445, 2024.

Lukasová, A. Hierarchical agglomerative clustering procedure. *Pattern Recognition*, 11(5-6):365–381, 1979.

Pang, Z., Zhang, T., Luan, F., Man, Y., Tan, H., Zhang, K., Freeman, W. T., and Wang, Y.-X. Randar: Decoder-only autoregressive visual generation in random orders, 2025. URL https://arxiv.org/abs/2412.01827.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Ranzato, M., Chopra, S., Auli, M., and Zaremba, W. Sequence level training with recurrent neural networks. *arXiv preprint arXiv:1511.06732*, 2015.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training gans, 2016. URL https://arxiv.org/abs/1606.03498.

Sun, P., Jiang, Y., Chen, S., Zhang, S., Peng, B., Luo, P., and Yuan, Z. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

Tenenbaum, J. B., De Silva, V., and Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

Tian, K., Jiang, Y., Yuan, Z., Peng, B., and Wang, L. Visual autoregressive modeling: Scalable image generation via next-scale prediction. *Advances in neural information processing systems*, 37:84839–84865, 2024.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Van Den Oord, A., Vinyals, O., and Kavukcuoglu, K. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Ward Jr, J. H. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association*, 58(301):236–244, 1963.

Wu, P., Zhu, K., Liu, Y., Tang, L., Yang, J., Peng, Y., Zhai, W., Cao, Y., and Zha, Z.-J. Alitok: Towards sequence modeling alignment between tokenizer and autoregressive model. *arXiv preprint arXiv:2506.05289*, 2025.

Xiong, T., Liew, J. H., Huang, Z., Feng, J., and Liu, X. Gigatok: Scaling visual tokenizers to 3 billion parameters for autoregressive image generation. *arXiv preprint arXiv:2504.08736*, 2025.

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.

Yu, Q., He, J., Deng, X., Shen, X., and Chen, L.-C. Randomized autoregressive visual generation. *arXiv preprint arXiv:2411.00776*, 2024.

Zhang, Y., Chen, F., He, S., and Zhuang, B. Zipar: Parallel autoregressive image generation through spatial locality, 2025.

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

Zhou, C., Yu, L., Babu, A., Tirumala, K., Yasunaga, M., Shamis, L., Kahn, J., Ma, X., Zettlemoyer, L., and Levy, O. Transfusion: Predict the next token and diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*, 2024.

A. Experimental Setup and Implementation Details

This section provides the detailed experimental configurations required to reproduce the results presented in this paper, covering the hardware and software environment, training hyperparameters for the backbone models, parameter settings for our proposed MASC algorithm and other baselines, and specifics of the evaluation metrics.

A.1. Hardware and Software Environment

All models were trained and evaluated on a server cluster equipped with 8 NVIDIA H100 80GB GPUs. We use PyTorch `v2.1.0` as our primary deep learning framework, accelerated with CUDA `v12.1` and cuDNN `v8.9`. To ensure a consistent experimental environment, we used Python `v3.10`.

A.2. Backbone Model Training Hyperparameters

For a fair comparison, all variants of LlamaGen (Sun et al., 2024) (Baseline, +k-means, and +MASC) followed the original paper's training configuration, adapted for different model scales. All models were trained for 300 epochs on the ImageNet-1K (Deng et al., 2009) dataset. Detailed hyperparameters are provided in Table 8.

Table 8. Detailed training hyperparameters for the LlamaGen (B, L, XL) backbone models. These settings were uniformly applied to the Baseline, +k-means, and +MASC methods to ensure a fair comparison.

Hyperparameter	LlamaGen-B	LlamaGen-L	LlamaGen-XL
Model Architecture			
Parameter Count (Baseline)	111M	343M	775M
Parameter Count (+k-means / +MASC)	94M	311M	719M
Optimizer			
Optimizer	AdamW	AdamW	AdamW
Betas (β_1, β_2)	(0.9, 0.95)	(0.9, 0.95)	(0.9, 0.95)
Epsilon (ϵ)	1×10^{-8}	1×10^{-8}	1×10^{-8}
Weight Decay	0.05	0.05	0.05
Learning Rate Schedule			
Peak Learning Rate	1×10^{-4}	1×10^{-4}	2×10^{-4}
LR Scheduler	Cosine Annealing	Cosine Annealing	Cosine Annealing
Final Learning Rate	1×10^{-5}	1×10^{-5}	2×10^{-5}
Training Configuration			
Training Epochs	300	300	300
Global Batch Size	256	256	512
Gradient Clipping	1.0	1.0	1.0

A.3. MASC and Baselines Hyperparameters

This section details the parameter settings for the clustering algorithms used to construct the codebook prior. All methods operate on the codebook from the pre-trained VQ-VAE tokenizer of LlamaGen, which has a vocabulary size of $N = 16,384$.

[leftmargin=15pt]

- **MASC (Ours)**: Our method is primarily controlled by the target number of clusters, k. In all comparative experiments, we set $k = 8,192$ to maintain an identical number of model parameters as the k-means baseline. The MASC algorithm is deterministic as it involves no random initialization. The distance metric used is detailed in Equation (3) of the main paper.

- **k-means**: We used the standard implementation from the `scikit-learn` library. The target number of clusters was set to $k = 8,192$. For stability, we used a random initialization strategy (`init='random'`) and performed 10 independent runs, selecting the result with the lowest inertia. The maximum number of iterations (`max_iter`) was set to 300, with a convergence tolerance (`tol`) of 1×10^{-4}.

- **k-means++**: As a stronger baseline to k-means, we also set $k = 8,192$. The only difference from the standard k-means setup is the use of the k-means++ seeding strategy (Arthur & Vassilvitskii, 2007) (`init='k-means++'`). All other parameters (max iterations, tolerance) remained the same.

A.4. Evaluation Details

To ensure a rigorous and comparable evaluation, we followed the standard procedures below:

[leftmargin=15pt]

- **Generation Quality Metrics**: We generated 50,000 samples in total for evaluation: 50 images for each of the 1,000 classes in the ImageNet 1K validation set. Fréchet Inception Distance (FID), Inception Score (IS), Precision, and Recall were all calculated using the `torch-fidelity` library. The FID was computed against the widely-used pre-calculated Inception-V3 feature statistics from the full ImageNet training set.

- **Prediction Uncertainty Metrics**: Prediction uncertainty is measured by the Shannon entropy of the model's output probability distribution P_t at each generation step. For a vocabulary of size V (where $V = 16,384$ for the Baseline and $V = 8,192$ for +k-means/+MASC), the entropy is calculated as:

$$H(P_t) = -\sum_{i=1}^{V} P_t(i) \log_2 P_t(i) \tag{6}$$

  The values we report are the average entropy across all generation steps ($t = 1, \ldots, 256$) for all images in the ImageNet validation set. The Normalized Entropy is calculated as $H_{\text{norm}} = H_{\text{actual}} / \log_2(V)$ to provide a fair comparison of task complexity, independent of vocabulary size.

B. Algorithmic and Theoretical Analysis of MASC

This section provides a deeper analysis of the MASC algorithm, focusing on its computational complexity and the principled motivation behind its core design choices, particularly the manifold-aligned distance metric.

B.1. Computational Complexity Analysis

The MASC framework is designed as a one-time, offline preprocessing step. While its asymptotic complexity is higher than that of k-means, its deterministic nature and efficient implementation make it highly practical for typical codebook sizes. We analyze its complexity below.

Let N be the number of tokens in the codebook, d be the embedding dimension, and k be the target number of clusters.

[leftmargin=15pt]

- **Initialization (Distance Matrix Pre-computation)**: The first step of the algorithm is to compute the initial $N \times N$ pairwise Euclidean distance matrix $\mathbf{D}$. Calculating the distance between two d-dimensional vectors takes $O(d)$ time. Since there are $\binom{N}{2} = O(N^2)$ pairs, the total time complexity for this step is $\mathcal{O}(N^2 d)$. This step also requires $\mathcal{O}(N^2)$ space to store the distance matrix.

- **Iterative Merging Loop**: The algorithm performs $N - k$ merging iterations. In each iteration, the following operations are performed:

  [leftmargin=15pt, nosep]

  1. *Find Minimum Distance*: The algorithm must find the minimum value in the current distance matrix to identify the next pair of clusters to merge. A naive scan of the (upper-triangular part of the) matrix takes $\mathcal{O}(N^2)$ time.
  2. *Update Distance Matrix*: After merging clusters C_{s^*} and C_{t^*} into a new cluster represented by index s^*, we must update the distances from this new cluster to all other active clusters C_u. As shown in Algorithm 1, this update is performed efficiently in $\mathcal{O}(N)$ time by iterating through the remaining active clusters and applying the weighted average formula, rather than recomputing all pairwise distances from scratch.

Since finding the minimum distance is the computational bottleneck in each of the $N - k$ iterations, the total time complexity for the merging loop is $\mathcal{O}((N - k)N^2)$, which simplifies to $\mathcal{O}(N^3)$.

- **Total Complexity**: The overall time complexity of the optimized MASC algorithm is the sum of the initialization and merging steps: $\mathcal{O}(N^2 d + N^3)$. Given that in practice N is often significantly larger than d (e.g., $N = 16,384$, $d = 32$), the $\mathcal{O}(N^3)$ term dominates. The space complexity is dominated by the storage of the distance matrix, resulting in $\mathcal{O}(N^2)$.

Comparison with K-means: The standard k-means algorithm has a time complexity of $\mathcal{O}(I \cdot k \cdot N \cdot d)$, where I is the number of iterations. While this appears more favorable asymptotically, several practical considerations are important. First, k-means is a heuristic algorithm sensitive to initialization, often requiring multiple runs to find a reasonable solution. In contrast, MASC is deterministic and always yields the same optimal hierarchy for a given distance metric. Second, the cost of MASC is a one-time, upfront computational investment. For a typical codebook size of $N = 16,384$, our optimized implementation runs in under a minute on a single modern GPU, a negligible cost when compared to the hundreds of GPU-hours required for training the main autoregressive model. This makes MASC a highly practical and efficient choice for constructing a high-quality, stable, and reproducible codebook prior. This finding is consistent with prior work on similar agglomerative clustering methods, which also report feasible runtimes for large codebooks.

B.2. Justification of the Manifold-Aligned Distance Metric

The choice of the distance metric in Equation (3) is a cornerstone of MASC's design, intended to be a robust and principled proxy for semantic similarity on the codebook manifold. This instance-based average distance, also known in hierarchical clustering literature as the Unweighted Pair Group Method with Arithmetic Mean (UPGMA) or average-linkage, offers a crucial advantage over centroid-based methods and other common linkage criteria.

As established in the main paper, centroid-based distances are ill-suited because centroids are often off-manifold and Euclidean distance is a poor approximation of the true geodesic distance on the manifold (Beyer et al., 1999; Huh et al., 2023). Our metric is centroid-free by design. To further justify its selection, we compare it against two other common centroid-free linkage criteria used in hierarchical clustering:

[leftmargin=15pt]

- **Single-linkage (MIN)**: This metric defines the distance between two clusters as the minimum distance between any two points in the respective clusters, i.e., $\mathcal{D}(C_s, C_t) = \min_{v_i \in C_s, v_j \in C_t} \|v_i - v_j\|_2$. While computationally efficient, single-linkage is highly susceptible to the chaining effect, where a few intermediate points can cause two otherwise distant clusters to be merged (see Figure 7). This behavior is undesirable for our task, as it can lead to large, elongated, and semantically diverse clusters, failing to capture the compact semantic groups we aim to model.

- **Complete-linkage (MAX)**: This metric uses the maximum distance between any two points in the clusters, i.e., $\mathcal{D}(C_s, C_t) = \max_{v_i \in C_s, v_j \in C_t} \|v_i - v_j\|_2$. It is the opposite of single-linkage and tends to produce very compact, roughly spherical clusters. However, it is highly sensitive to outliers and may fail to correctly group elongated or non-convex shapes that are nevertheless semantically coherent on the manifold.

- **Average-linkage (MASC's choice)**: Our chosen metric calculates the average distance between all pairs of points across two clusters. It serves as a robust compromise between the extremes of single- and complete-linkage. By considering the entire distribution of points within both clusters, it is less sensitive to outliers than complete-linkage and less prone to the chaining effect than single-linkage. This formulation effectively measures the overall closeness of two clusters, making it well-suited to identifying semantically coherent groups, even if they form non-spherical shapes on the manifold. Its effectiveness in capturing discriminative priors for codebooks has been validated in related works.

In summary, the choice of average-linkage is a deliberate design decision to create a clustering hierarchy that is robust, deterministic, and best reflects the underlying semantic structure of the codebook manifold by balancing compactness and shape-invariance.

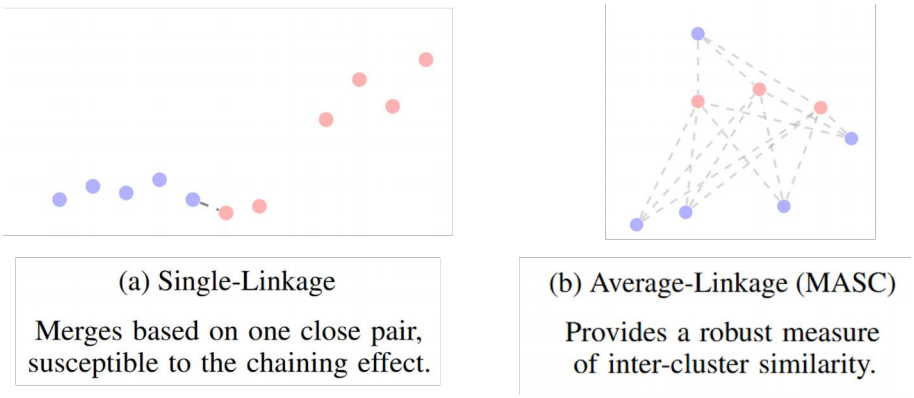

Figure 7. A conceptual illustration of linkage criteria. (a) Single-linkage may incorrectly merge two distinct semantic groups if they are connected by a bridge of a few close points. (b) Average-linkage, as used in MASC, considers the overall distribution of points and is more robust, correctly identifying distinct clusters.

Algorithm 1 Manifold-Aligned Semantic Clustering (MASC) Construction

Require: Codebook embeddings $\mathcal{Z} = \{v_1, \ldots, v_N\} \in \mathbb{R}^{N \times d}$, target number of clusters k.
Ensure: A mapping $\mathcal{M} : \{1, \ldots, N\} \to \{1, \ldots, k\}$ from token indices to cluster indices.
0: $\{$*Initialization*$\}$
0: Initialize N active clusters, $C_j \leftarrow \{v_j\}$, and their sizes, $|C_j| \leftarrow 1$, for $j = 1, \ldots, N$.
0: Pre-compute the full $N \times N$ pairwise Euclidean distance matrix $\mathbf{D}$, where $\mathbf{D}_{st} = \|v_s - v_t\|_2$.
0: Set diagonal elements $\mathbf{D}_{ss} \leftarrow \infty$ to prevent self-merging.
0: $\{$*Bottom-Up Hierarchical Construction*$\}$
0: **for** $i \leftarrow 1$ to $N - k$ **do**
0: Find the pair of active clusters with the minimum distance: $(s^*, t^*) \leftarrow_{s,t} \mathbf{D}_{st}$.
0: $\{$Update distance matrix efficiently using a weighted average$\}$
0: **for** each remaining active cluster C_u where $u \neq s^*, t^*$ **do**
0: $\mathbf{D}_{s^*,u} \leftarrow \frac{|C_{s^*}|\mathbf{D}_{s^*,u} + |C_{t^*}|\mathbf{D}_{t^*,u}}{|C_{s^*}| + |C_{t^*}|}$; $\mathbf{D}_{u,s^*} \leftarrow \mathbf{D}_{s^*,u}$
0: **end for**
0: $\{$Update cluster size and deactivate the merged cluster $C_{t^*}\}$
0: $|C_{s^*}| \leftarrow |C_{s^*}| + |C_{t^*}|$.
0: Set row and column t^* of $\mathbf{D}$ to ∞.
0: Keep track that all original tokens from cluster C_{t^*} now belong to cluster C_{s^*}.
0: **end for**
0: $\{$*Final Mapping Construction*$\}$
0: The remaining k active clusters form the final coarse vocabulary.
0: Assign a unique index from $\{1, \ldots, k\}$ to each of the final k active clusters.
0: Construct the mapping $\mathcal{M}$ by assigning each original token v_i to its final cluster index.
0: **return** The mapping $\mathcal{M}$. =0

C. Extended Experimental Results and Analyses

This section expands upon the experimental results presented in the main paper. We provide detailed analyses, including qualitative visualizations of cluster coherence, ablation studies on key hyperparameters, and a deeper look into the effects of different decoding strategies. These results collectively offer a comprehensive validation of the MASC framework.

C.1. Ablation Study on the Number of Coarse Clusters (k)

The number of coarse clusters, k, is a critical hyperparameter that balances the trade-off between simplifying the prediction space and preserving sufficient visual information. A small k results in large, semantically broad clusters, leading to a loss of detail (high intra-cluster variance). A large k approaches the original flat vocabulary, diminishing the benefits of clustering. We performed an ablation study on k using the LlamaGen-L backbone. The results in Table 9 show that performance peaks

around $k = 8,192$. This configuration provides a significant reduction in vocabulary size while retaining enough granularity for high-fidelity image generation, justifying its use as our default setting.

Table 9. Ablation study on the number of clusters (k) for MASC, evaluated on the LlamaGen-L model. Performance generally improves as k increases, but the gains diminish while the model size and prediction complexity grow. We select $k = 8,192$ as it offers the best balance of performance, efficiency, and task simplification.

k	# Params	Norm. Entropy ↓	FID ↓	IS ↑	Recall ↑
2,048	298M	0.11	4.15	231.4	0.52
4,096	305M	0.12	3.37	250.1	0.55
8,192	**311M**	**0.14**	**2.92**	**259.2**	**0.57**
12,288	327M	0.16	2.90	261.5	0.57

C.2. Ablation Study on Inference Decoding Strategy

As described in Section 3.3, we primarily use a simple and efficient Random Sampling strategy for decoding the generated coarse cluster indices into fine-grained tokens. We also proposed an extended Hierarchical Decoding strategy that employs a small refinement network. Here, we analyze the trade-offs between these two approaches. The refinement network is a single-layer Transformer decoder with 4 attention heads and an embedding dimension of 512, adding approximately 25M parameters.

Table 10 shows that while the learned Hierarchical Decoding strategy provides a marginal improvement in generation quality (a 0.05 reduction in FID), it comes at the cost of additional parameters and a slight decrease in inference speed. The strong performance of the default Random Sampling strategy underscores the high semantic consistency of the clusters produced by MASC. Any token within a cluster serves as a good representative. This makes Random Sampling an excellent default choice, offering a compelling balance of simplicity, efficiency, and high performance.

Table 10. Comparison of decoding strategies for LlamaGen-L + MASC. Hierarchical Decoding offers a slight fidelity gain at the cost of increased model size and reduced inference speed.

Decoding Strategy	Add. Params	Speed (img/s)	FID ↓	IS ↑
Random Sampling (Default)	**0M**	**8.71**	**2.92**	**259.2**
Hierarchical Decoding	+25M	8.25	2.87	260.8

D. Empirical Analysis

To rigorously verify the core hypothesis of MASC, we conducted a series of comprehensive empirical analyses. These experiments go beyond standard generation metrics to directly probe the internal coherence of the constructed clusters.

D.1. Semantic Replacement Test

We propose the Semantic Replacement Test to evaluate the semantic consistency within clusters. The intuition is straightforward: if a cluster truly groups semantically similar tokens , then replacing a token in an image with another random token from the same cluster should preserve the overall semantic structure and texture of the image, despite potential loss of high-frequency details. Conversely, if a cluster is incoherent (grouping visually unrelated tokens), such replacement would introduce significant noise and semantic artifacts.

Experimental Protocol: For each image token z_i, we replaced it with a token z_i' randomly sampled from the same cluster C_k assigned by either k-means or MASC. We then decoded the perturbed token sequences back into images and measured the reconstruction quality using rFID, PSNR, and SSIM against the original images.

Results: As shown in Table 11, MASC significantly outperforms k-means and k-means++ across all metrics. Notably, MASC achieves a much lower rFID and higher SSIM, indicating that the substituted tokens are perceptually and structurally much closer to the originals. This confirms that MASC clusters maintain a high degree of semantic interchangeability, whereas k-means clusters often group unrelated tokens, leading to destructive artifacts upon replacement.

Table 11. Image reconstruction performance with replaced tokens. We replace image tokens extracted by the tokenizer with random tokens from the same cluster. MASC demonstrates significantly less performance degradation, indicating higher intra-cluster semantic consistency.

Method	Vocab. (k)	rFID ↓	PSNR ↑	SSIM ↑
No Replacement	16,384	2.26	21.03	0.542
w/ k-means	8,192	4.45	19.02	0.452
w/ k-means++	8,192	4.17	19.29	0.469
w/ MASC (Ours)	8,192	**2.92**	**20.49**	**0.513**
w/ k-means	4,096	8.41	18.37	0.431
w/ k-means++	4,096	8.17	18.52	0.439
w/ MASC (Ours)	4,096	**6.73**	**19.62**	**0.473**

Qualitative Visual Analysis: We visually examine the impact of token replacement in Figure 4 (a-c).

- **K-means Replacement (Fig. 4b):** The image suffers from severe semantic noise and artifacts. For instance, the astronaut's face is corrupted with unrelated textures. This indicates that k-means clusters group visually disparate tokens simply because they share similar average color values.
- **MASC Replacement (Fig. 4c):** In stark contrast, MASC replacement preserves the global structure, object identity, and local textures. Although high-frequency details are naturally smoothed due to random sampling, the semantic content remains intact—the face looks like a face, and the suit retains its material appearance.

D.2. Visualizing Cluster Assignments

To further demystify the clusters, we directly visualize the image patches corresponding to tokens within a single cluster.

- **MASC Clusters (Fig. 4e):** MASC demonstrates remarkable **semantic purity**. The visualized cluster consistently groups patches with specific textures, proving that our density-driven metric successfully captures the underlying manifold structure of visual features.
- **K-means Clusters (Fig. 4f):** K-means clusters exhibit **semantic confusion**. As k-means relies on Euclidean distance to centroids, it groups patches with similar average colors into the same cluster, ignoring their distinct structural semantics.

D.3. Quantitative Semantic Consistency

Finally, we quantify the semantic homogeneity using DINOv2, a vision foundation model known for capturing high-level visual semantics. We computed the pairwise cosine similarity of DINOv2 features for tokens within the same cluster. Figure 4(d) shows that MASC clusters achieve a significantly higher median intra-cluster cosine similarity compared to k-means. The tighter distribution and higher mean score quantitatively validate that MASC groups tokens that are aligned in a deep semantic space, not just in pixel space.

E. Additional Evaluation Metrics.

To quantify **Prediction Uncertainty**, we employ Shannon entropy, defined as $H(P_t) = -\sum_{i=1}^{V} P_t(i) \log_2 P_t(i)$, on the model's output distribution P_t. We report both the average entropy and a normalized version, $H_{\text{norm}} = H_{\text{actual}} / \log_2(V)$, for a vocabulary-size-agnostic comparison.

Table 12. Prediction Uncertainty analysis (Entropy) on ImageNet 256 × 256. Comparison of Baseline, k-means, and our MASC method.

Model	Method	Pred. / Norm. Entropy
LlamaGen-B (111M)	LlamaGen Baseline	2.81 / 0.20
	+ k-means	2.72 / 0.21
	+ MASC (Ours)	**1.90 / 0.15**
LlamaGen-L (343M)	LlamaGen Baseline	2.65 / 0.19
	+ k-means	2.58 / 0.20
	+ MASC (Ours)	**1.82 / 0.14**
LlamaGen-XL (775M)	LlamaGen Baseline	2.58 / 0.18
	+ k-means	2.51 / 0.19
	+ MASC (Ours)	**1.75 / 0.13**

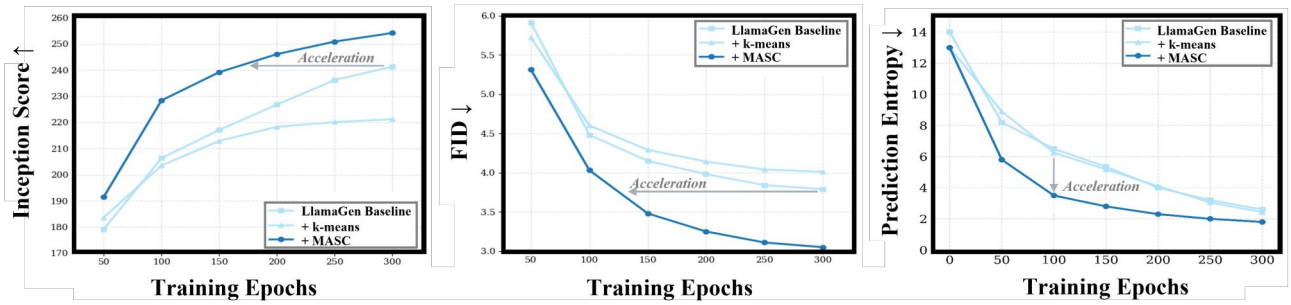

Figure 8. Training dynamics of LlamaGen-L on ImageNet. The plots compare the IS (left, higher is better), FID (middle, lower is better) and Prediction Entropy (right, lower is better) scores over training epochs for the Vanilla baseline, the + k-means variant, and our + MASC method. This visualizes the training acceleration benefit of operating in a low-entropy prediction space.

F. Additional Generated Samples

To further showcase the generation capabilities of our MASC-enhanced model, Figure 9 presents a broader gallery of generated images. These samples were generated using the LlamaGen-XL + MASC model, with a Classifier-Free Guidance (CFG) scale of 2.5, which we found to produce visually pleasing results. The images span a wide range of ImageNet classes, including animals, objects, food, and scenes, demonstrating the model's ability to generate diverse, high-fidelity, and compositionally sound images.

Figure 9. Additional samples generated by LlamaGen-XL + MASC.