

Self-conditioned Image Generation via Generating Representations

Tianhong Li¹Dina Katabi¹Kaiming He²¹MIT CSAIL²FAIR, Meta

Abstract

This paper presents **Representation-Conditioned image Generation (RCG)**, a simple yet effective image generation framework which sets a new benchmark in class-unconditional image generation. RCG does not condition on any human annotations. Instead, it conditions on a self-supervised representation distribution which is mapped from the image distribution using a pre-trained encoder. During generation, RCG samples from such representation distribution using a representation diffusion model (RDM), and employs a pixel generator to craft image pixels conditioned on the sampled representation. Such a design provides substantial guidance during the generative process, resulting in high-quality image generation. Tested on ImageNet 256×256, RCG achieves a Frechet Inception Distance (FID) of 3.31 and an Inception Score (IS) of 253.4. These results not only significantly improve the state-of-the-art of class-unconditional image generation but also rival the current leading methods in class-conditional image generation, bridging the long-standing performance gap between these two tasks. Code is available at <https://github.com/LTH14/rcg>.

1. Introduction

Recent advancements in conditional image generation have yielded impressive results, leveraging human annotations such as class labels or text descriptions to guide the generative process [11, 12, 18, 22, 47, 52]. In contrast, unconditional image generation which omits such conditional element has historically been a more challenging task, often yielding less impressive results [3, 18, 19, 39, 43].

This dichotomy mirrors the one seen between supervised and unsupervised learning. Historically, unsupervised learning lagged behind its supervised counterpart in performance. This gap has narrowed with the advent of *self-supervised learning* (SSL), which generates supervisory signals from the data itself, achieving competitive or superior results compared to supervised learning [9, 13, 25–27].

Drawing on this analogy, we consider the problem of *self-conditioned image generation* as a counterpart to self-supervised learning in the realm of image generation. This approach, distinct from traditional unconditional image generation, conditions the pixel generation process on a representation distribution derived from the data distribution

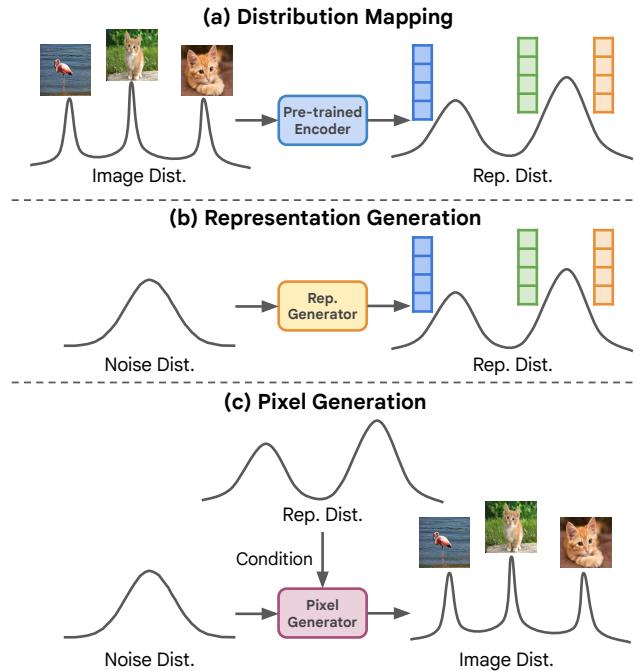


Figure 1. Self-conditioned image generation framework. Unlike traditional unconditional image generation methods which simply map noise distribution to image distribution, self-conditioned image generation consists of three parts: (a) it uses an image encoder (e.g., Moco v3) to map the raw image distribution to a low-dimensional representation distribution; (b) it learns a representation generator to map a noise distribution to the representation distribution; (c) it learns a pixel generator (e.g., LDM [52] or MAGE [39]) to map a noise distribution to the image distribution conditioned on the representation distribution.

itself, as shown in Figure 1c.

Self-conditioned image generation is important for several reasons. Firstly, self-conditioning on representations is a more intuitive approach for unconditional image generation, mirroring an artist’s process of conceptualizing an abstract idea before translating it onto a canvas. Secondly, similar to how self-supervised learning has transcended supervised learning, self-conditioned image generation, utilizing extensive unlabeled datasets, has the potential to exceed the performance of conditional image generation. Thirdly, by omitting reliance on human annotations, self-conditioned generation paves the way for generative applications in domains beyond human annotation capabilities, such as molecule design or drug discovery.

The core of self-conditioned image generation lies in

accurately modeling and sampling from an image representation distribution (Figure 1b). Such image representation should also retain sufficient information to guide the pixel generation process. To realize this, we develop a Representation Diffusion Model (RDM) to generate low-dimensional self-supervised image representations. This distribution is mapped from the image distribution using a self-supervised image encoder (Figure 1a). Our approach offers two significant benefits. First, the RDM can capture the diversity of the representation space’s underlying distribution, enabling it to generate a variety of representations to facilitate image generation. Second, this self-supervised representation space is both structured and of low dimensionality, which simplifies the representation generation task for a straightforward neural network architecture. Consequently, the computational overhead of generating representations is minimal compared to the pixel generation process.

With RDM, we present Representation-Conditioned image Generation (RCG), a simple yet effective framework for self-conditioned image generation. RCG consists of three components: an SSL image encoder (Moco v3 [16]) to transform the image distribution into a compact representation distribution, an RDM to model and sample from this distribution, and a pixel generator to craft image pixels conditioned on the representation. This design enables seamless integration of RCG with common image generative models as its pixel generator, improving their class-unconditional image generation performance by huge margins (Figure 2).

RCG demonstrates exceptional image generation capabilities. Evaluated on ImageNet 256×256, RCG achieves an FID of 3.56 and an Inception Score of 186.9, significantly outperforming all previous class-unconditional generation methods (the closest state-of-the-art result being 7.04 FID and 123.5 Inception Score [39]). Such results can be further improved to 3.31 FID and 253.4 Inception Score with classifier-free guidance. Remarkably, our results are comparable to or even surpass existing class-conditional generation benchmarks. These results underscore the great potential of self-conditioned image generation, potentially heralding a new era in this field.

2. Related Work

Self-supervised Learning. For a considerable period, supervised learning was predominantly superior to unsupervised learning across various computer vision tasks [8, 28, 29, 64]. However, the advent of self-supervised learning has significantly closed this performance gap. Initial efforts in self-supervised learning were centered around creating pretext tasks and training networks to predict associated pseudo-labels [23, 44, 46]. Generative models have also shown the ability to extract representations from images [19, 49]. A relevant work, DiffAE [49], conditions its diffusion model on the representation extracted by a semantic

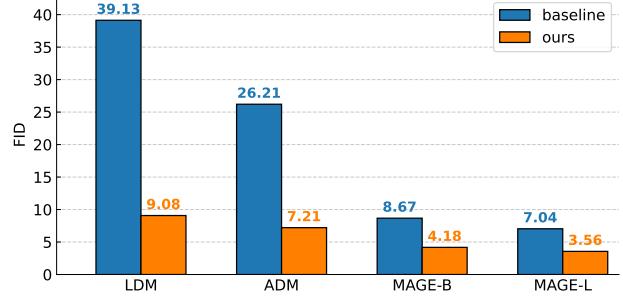


Figure 2. **Class-unconditional image generation performance on ImageNet 256×256 using different pixel generators.** Our method improves **class-unconditional generation quality** by huge margins, regardless of the choice of the pixel generator baseline.

encoder. Such semantic encoder is trained from scratch together with the diffusion model, enabling DiffAE to learn a meaningful and decodable image representation which facilitates image manipulations.

Recently, contrastive learning [14, 15, 38, 45] has shown to be a robust and systematic approach to learning effective representations, achieving results nearly on par with those of supervised learning. Researchers have also discovered that masked image modeling (MIM) is highly effective in self-supervised learning [4, 26, 35, 39, 48]. Such advancements in self-supervised learning have led us to explore the concept of *self-conditioned image generation*. Our proposed framework, RCG, leverages cutting-edge self-supervised learning methods to map the image distribution to a compact representation distribution.

Image Generation. Recent years have witnessed tremendous progress in deep generative models for image synthesis. One major stream of generative models is built on top of generative adversarial networks (GANs) [7, 24, 36, 62, 63]. Another stream is based on a two-stage scheme [11, 12, 37, 39, 51, 60, 61]: first tokenize the image into a latent space and then apply maximum likelihood estimation and sampling in the latent space. Recently, diffusion models [18, 31, 50, 52, 56] have also achieved superior results on image synthesis. A relevant work, DALLE 2 [50], generates CLIP image embedding conditioned on CLIP text embeddings and image captions and generate images conditioned on the generated image embeddings, demonstrating superior performance in text-to-image generation.

Despite their impressive performance, a notable gap exists between conditional and unconditional generation capabilities [3, 18, 19, 39, 43]. Prior efforts to narrow this gap group images into clusters in the representation space and use these clusters as underlying class labels for self-conditioning or self-guidance [3, 34, 40]. However, this implicitly assumes that the dataset, which is supposed to be unlabeled, is a classification dataset and the optimal number of clusters is close to the number of the classes. Additionally, these methods fall short of generating diverse represen-

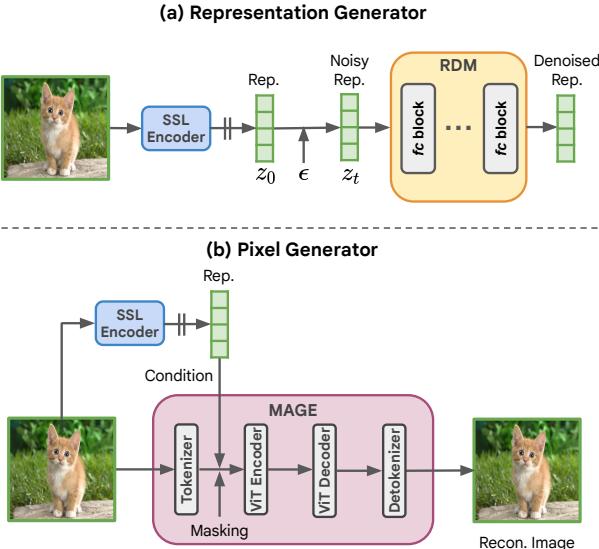


Figure 3. RCG training framework. The pre-trained SSL image encoder extracts representations from images and is fixed during training. To train RDM, we add standard Gaussian noise to the representations and ask the network to denoise them. To train the MAGE pixel generator, we add random masking to the tokenized image and ask the network to reconstruct the missing tokens conditioned on the representation extracted from the same image.

tations – they are unable to produce different representations within the same cluster or the same underlying class.

Other two relevant works are RCDM [5] and IC-GAN [10], where images are generated based on representations extracted from existing images. Nonetheless, these methods rely on ground-truth images to provide representations during generation, a requirement that is impractical in many generative applications.

RCG’s conditioning differs from all prior works. Unlike previous self-conditioned methods which produced a discrete set of pre-computed clusters as conditioning, RCG learns a representation diffusion model to model the underlying distribution of a representation space and generates images conditioned on this representation distribution. The generation of this SSL representation is achieved through a simple yet effective representation diffusion model. To the best of our knowledge, this is the first exploration and solution to generating a low-dimensional SSL representation and using it as conditioning for image generation. This ability to model and sample from such a representation distribution allows the pixel generation process to be guided by a comprehensive understanding of the image without the need for human annotations. Consequently, this leads to a significantly better performance than previous methods in unconditional image generation.

3. Method

RCG comprises three key components: a pre-trained self-supervised image encoder, a representation generator, and

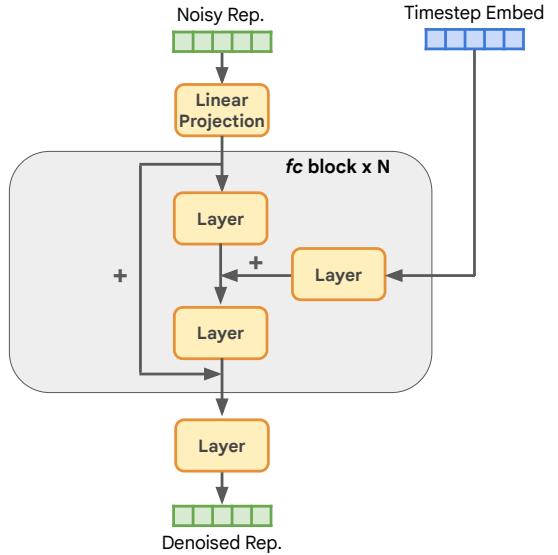


Figure 4. RDM’s backbone architecture. Each “Layer” consists of a LayerNorm layer [1], a SiLU layer [21], and a linear layer. The backbone consists of an input layer that projects the representation to hidden dimension C , followed by N fully connected (fc) blocks, and an output layer that projects the hidden latent back to the original representation dimension.

a pixel generator. Each component’s design is elaborated below:

Image Encoder. RCG employs a pre-trained image encoder to transit the image distribution to a representation distribution. This distribution is characterized by two essential properties: simplicity for modeling by a representation diffusion model, and richness in high-level semantic content for guiding pixel generation. We use image encoders pre-trained with self-supervised contrastive learning methods (Moco v3 [16]), which regularize the representations on a hyper-sphere while achieving state-of-the-art representation learning performance on ImageNet. We take the representations after the projection head (256-dim), and each representation is normalized by its own mean and standard deviation.

Representation Generator. RCG uses a simple yet effective representation diffusion model (RDM) to sample from the representation space. RDM employs a fully connected network with multiple residual blocks as its backbone, shown in Figure 4. Each block consists of an input layer, a timestep embedding projection layer, and an output layer, where each layer consists of a LayerNorm [1], a SiLU [21], and a linear layer. Such an architecture is controlled by two parameters: the number of residual blocks N , and the hidden dimension C .

RDM follows Denoising Diffusion Implicit Models (DDIM) [55] for training and inference. As shown in Figure 3a, during training, image representation z_0 is mixed with standard Gaussian noise variable ϵ : $z_t = \sqrt{\alpha_t}z_0 + \sqrt{1 - \alpha_t}\epsilon$. The RDM backbone is then trained to denoise z_t back to z_0 . During inference, RDM generates representations from Gaussian noise following the DDIM sampling process [55].

Since RDM operates on highly compacted representations, it brings marginal computation overheads for both training and generation (Table 7).

Pixel Generator. The pixel generator in RCG crafts image pixels conditioned on image representations. Conceptually, such a pixel generator can be any modern conditional image generative model by substituting its original conditioning (e.g., class label or text) with SSL representations. In Figure 3b, we take MAGE [39], a parallel decoding generative model as an example. The pixel generator is trained to reconstruct the original image from a masked version of the image, conditioned on the representation of the same image. During inference, the pixel generator generates images from a fully masked image, conditioned on the representation from the representation generator.

We experiment with three representative generative models: ADM [18] and LDM [52], both of which are diffusion-based frameworks, and MAGE [39], a parallel decoding framework. Our experiments show that all three generative models achieve much better performance when conditioned on high-level representations (Figure 2 and Table 6b).

Classifier-free Guidance. One advantage of RCG is that it seamlessly facilitates classifier-free guidance for unconditional generation tasks. Classifier-free guidance, known for enhancing generative model performance, traditionally was not applicable in unconditional generation frameworks [33, 39]. This is because classifier-free guidance is designed to provide guidance for conditional image generation through unconditional generation. Although RCG is also designed for unconditional generation tasks, the pixel generator of RCG is conditioned on self-supervised representations, and thus can seamlessly integrate classifier-free guidance which further boosts its generation performance.

RCG follows Muse [11] to enable classifier-free guidance in its MAGE pixel generator. During training, the MAGE pixel generator is trained without being conditioned on SSL representations with 10% probability. During each inference step, MAGE predicts a logit l_c conditioned on SSL representation, and an unconditional logit l_u , for each masked token. The final logits l_g are formed by l_c moving away from l_u by the guidance scale τ : $l_g = l_c + \tau(l_c - l_u)$. MAGE then samples according to l_g to fill in the remaining masked tokens. Additional implementation details of RCG’s classifier-free guidance are provided in Appendix B.

4. Results

4.1. Setup

We evaluate RCG on ImageNet 256×256 [17] which is a common benchmark dataset for image generation. We generate 50K images and report the Frechet Inception Distance (FID) [30] and Inception Score (IS) [53] as standard metrics to measure the fidelity and diversity of the generated images.

Table 1. **Image generation performance on ImageNet 256×256 without guidance.** RCG outperforms all class-conditional and class-unconditional baselines while requiring similar or less computational costs as later shown in Table 7.

Methods w/o Guidance	FID↓	Inception Score↑
<i>Class-conditional Generation</i>		
ADM [18]	10.94	101.0
LDM-4 [52]	10.56	103.5
DiT-XL/2 [47]	9.62	121.5
BigGAN-deep [6]	6.95	198.2
MDT-XL/2 [22]	6.23	143.0
MaskGIT [12]	6.18	182.1
CDM [32]	4.88	158.7
<i>Class-unconditional Generation</i>		
BigGAN [19]	38.61	24.7
ADM [18]	26.21	39.7
MaskGIT [12]	20.72	42.1
RCDM [5]	19.0	51.9
IC-GAN [10]	15.6	59.0
ADDP [58]	8.9	95.3
MAGE-L [39]	7.04	123.5
RCG-L	3.56	186.9

The FID is measured against the ImageNet validation set. During the training of RCG’s pixel generator, the image is resized so that the smaller side is of length 256, and then randomly flipped and cropped to 256×256. The input to the SSL encoder is further resized to 224×224 to be compatible with its positional embedding size. For our main results, RCG-L uses vision Transformers (ViT-L) [20] pre-trained with Moco v3 [16] as the image encoder, a network with 12 blocks and 1536 hidden dimensions as the backbone of RDM, and MAGE-L [39] as the image generator. The RDM is trained for 200 epochs with a constant learning rate and MAGE-L is trained for 800 epochs with cosine learning rate scheduling. More implementation details and hyper-parameters are provided in Appendix B.

4.2. Class-unconditional Generation

In Table 1, we compare RCG with state-of-the-art generative models on ImageNet 256×256. Since traditional class-unconditional generation does not support either classifier or classifier-free guidance [18, 33], all results in Table 1 are reported without such guidance.

As shown in Figure 5 and Table 1, RCG can generate images with both high fidelity and diversity, achieving 3.56 FID and 186.9 Inception Score, which significantly outperforms previous state-of-the-art class-unconditional image generation methods. Moreover, such a result also outperforms the previous state-of-the-art class-conditional generation method (4.88 FID achieved by CDM [32]), bridging the historical gap between class-conditional and class-unconditional generation. We further show in Appendix A that our representation diffusion model can effortlessly fa-

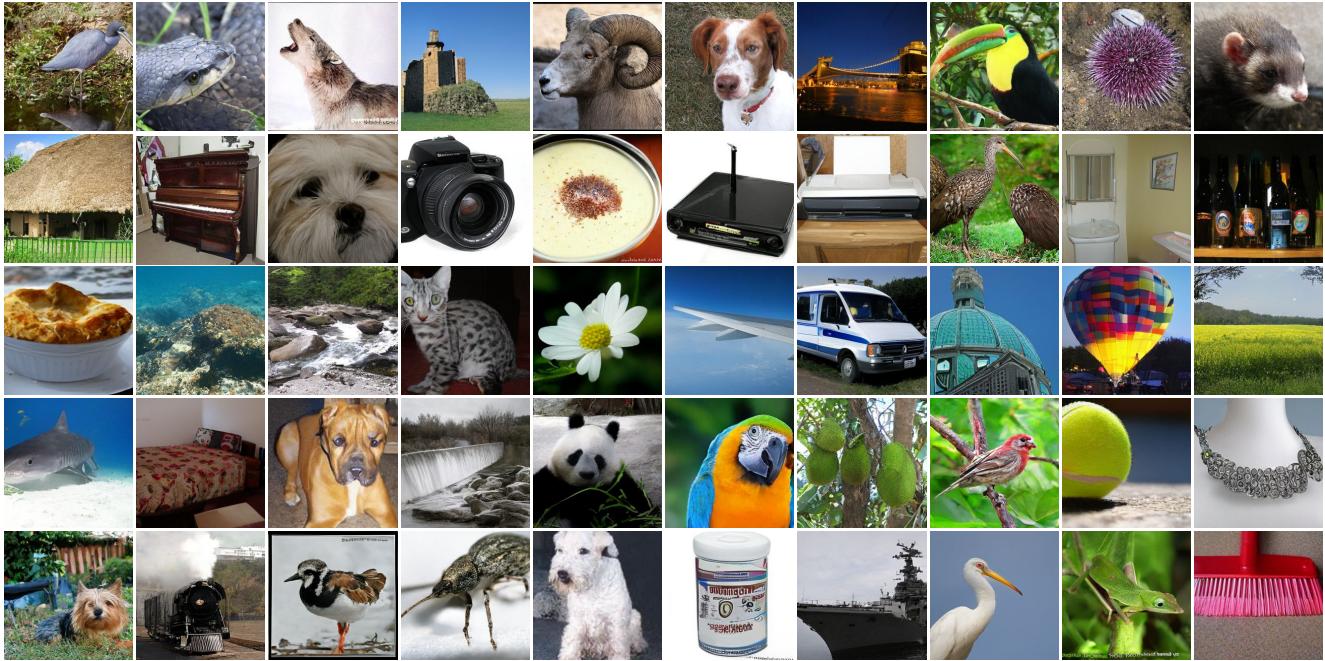


Figure 5. **RCG unconditional image generation results on ImageNet 256×256 without classifier-free guidance.** RCG can generate images with both high fidelity and diversity without conditioning on any human annotations.

Table 2. **Image generation performance on ImageNet 256×256 with guidance.**

RCG seamlessly enables classifier-free guidance for unconditional image generation, achieving results on par with state-of-the-art class-conditional generative models with guidance.

Methods w/ Guidance	FID↓	Inception Score↑
<i>Class-conditional Generation</i>		
ADM-G, U [18]	3.94	215.8
LDM-4-G [52]	3.60	247.7
U-ViT-L-G [2]	3.40	-
DiT-XL-G [47]	2.27	278.2
MDT-XL/2-G [22]	1.79	283.0
<i>Class-unconditional Generation</i>		
RCG-L-G	3.31	253.4

cilitate class-conditional representation generation, thereby enabling RCG to also adeptly perform class-conditional image generation. This result demonstrates the effectiveness of RCG and further highlights the great potential of self-conditioned image generation.

4.3. Classifier-free Guidance

Traditional frameworks for class-unconditional image generation lack the ability to employ classifier guidance [18] in the absence of class labels. Moreover, they are also incompatible with classifier-free guidance as the guidance itself is from unconditional generation. A significant advantage of RCG lies in its ability to integrate classifier-free guidance into its pixel generator. As shown in Table 2, RCG’s per-

Table 3. **FID and Inception Score with different classifier-free guidance scales τ .** The FID is stable for $\tau \geq 1$, while a larger τ keeps improving the Inception Score.

τ	0.0	1.0	2.0	3.0	4.0	5.0	6.0	7.0
FID	3.56	3.29	3.37	3.44	3.31	3.33	3.31	3.39
IS	186.9	228.5	242.4	251.3	250.5	252.7	253.4	252.6

formance is notably improved by classifier-free guidance, reaching levels comparable to leading class-conditional image generation methods that utilize guidance. We also ablate our classifier-free guidance scale τ , as shown in Table 3. $\tau = 1$ can both improve FID and IS, and a larger τ keeps improving the Inception Score.

4.4. Ablations

This section provides a comprehensive ablation study of the three core components of RCG. Our default setup uses Moco v3 ViT-B as the pre-trained image encoder, an RDM with a 12-block, 1536-hidden-dimension backbone trained for 100 epochs, and a MAGE-B pixel generator trained for 200 epochs. The default setting is marked with gray throughout Tables 4 to 6. Unless otherwise stated, all other properties and modules are set to the default settings during each component’s individual ablation.

Pre-trained Encoder. We explore different pre-trained image encoder setup in Table 4. Table 4a compares image encoders trained via various SSL methods (Moco v3, DINO,

Method	FID	IS	Model	params	lin.	FID	IS	Projection Dim	FID	IS
No condition	14.23	57.7	ViT-S	22M	73.2	5.77	120.8	32	9.14	81.0
Moco v3 [16]	5.07	142.5	ViT-B	86M	76.7	5.07	142.5	64	6.09	119.2
DINO [9]	7.53	160.8	ViT-L	304M	77.6	5.06	148.2	128	5.19	143.3
iBOT [65]	8.05	148.7						256	5.07	142.5
								768	6.10	112.7

(a) **Pre-training method.** RCG achieves good performance with encoders pre-trained with different contrastive learning methods.

(b) **Model size.** RCG scales up with larger pre-trained encoders with better linear probing accuracy.

(c) **Projection dimension.** The dimensionality of the image representation is important in RCG’s performance.

Table 4. **Pre-trained encoder ablation experiments on ImageNet 256×256.** If not specified, the default pre-trained encoder is Moco v3 ViT-B with 256 projection dimension. Default settings are marked in gray .

#Blocks	FID	IS	Hidden Dim	FID	IS	Epochs	FID	IS	#Steps	FID	IS
3	7.53	113.5	256	12.99	67.3	10	5.94	124.4	20	5.80	120.3
6	5.40	132.9	512	9.07	99.8	50	5.21	138.3	50	5.28	133.0
12	5.07	142.5	1024	5.35	132.0	100	5.07	142.5	100	5.15	138.1
18	5.20	141.9	1536	5.07	142.5	200	5.07	145.1	250	5.07	142.5
24	5.13	141.5	2048	5.09	142.8	300	5.05	144.3	500	5.07	142.9

(a) **Model depth.** A deeper RDM can improve generation performance.

(b) **Model width.** A wider RDM can improve generation performance.

(c) **Training epochs.** Training RDM longer can improve generation performance.

(d) **Diffusion steps.** More sampling steps can improve generation performance.

Table 5. **RDM ablation experiments on ImageNet 256×256.** If not specified, the default RDM backbone is of 12 blocks and 1536 hidden dimensions, trained for 100 epochs, and takes 250 sampling steps during generation. Default settings are marked in gray .

Conditioning	FID	IS	Method	FID	IS	Epochs	FID	IS
No condition	14.23	57.7	LDM [39]	39.13	22.8	100	6.03	127.7
Class label	5.83	147.3	LDM+RDM	9.08 (-30.05)	101.9 (+79.1)	200	5.07	142.5
Generated rep.	5.07	142.5	ADM [18]	26.21	39.7	400	4.48	158.8
Oracle rep.	4.37	149.0	ADM+RDM	7.21 (-19.00)	108.9 (+69.2)	800	4.15	172.0

(a) **Conditioning.** Conditioning on generated representations improves over both unconditional and class-conditional baselines.

(b) **Pixel generator framework.** RCG consistently improves unconditional image generation performance with different generative models as its pixel generator.

(c) **Training epochs.** Training the MAGE pixel generator longer can improve generation performance.

Table 6. **Pixel generator ablation experiments on ImageNet 256×256.** If not specified, the default pixel generator is MAGE-B trained for 200 epochs. In Table 6b, ADM+RDM is trained for 100 epochs, LDM+RDM is trained for 40 epochs, and MAGE+RDM is trained for 800 epochs. The LDM paper does not include class-unconditional generation results on ImageNet, so we report its re-implementation result in [39]. Default settings are marked in gray .

and iBOT), highlighting their substantial improvements over the unconditional baseline. Additionally, an encoder trained with DeiT [59] in a supervised manner also exhibits impressive performance (5.51 FID and 211.7 IS), indicating RCG’s adaptability to both supervised and self-supervised pre-training approaches.

Table 4b assesses the impact of model size on the pre-trained encoder. Larger models with better linear probing accuracy consistently enhance generation performance, although a smaller ViT-S model (22M parameters) still achieves decent results (5.77 FID and 120.8 IS).

We further analyze the effect of image representation dimensionality, using Moco v3 ViT-B models trained with different output dimensions from their projection head. Table 4c shows that neither excessively low nor high-dimensional representations are ideal – too low dimensions

lose vital image information, while too high dimensions pose challenges for the representation generator.

Representation Generator. Table 5 ablates the representation diffusion model. The RDM’s architecture consists of fully connected blocks, with the network’s depth and width determined by the number of blocks and hidden dimensions. Table 5a and Table 5b ablate these parameters, indicating an optimal balance at 12 blocks and 1536 hidden dimensions. Further, Table 5c and Table 5d suggest that RDM’s performance saturates at around 200 training epochs and 250 diffusion steps. Despite incurring only marginal computational costs, the RDM proves highly effective in generating SSL representations as evidenced in Table 6a.

Pixel Generator. Table 6 ablates RCG’s pixel generator. Table 6a experiments with class-unconditional, class-conditional, and self-conditioned MAGE-B, assessing dif-

ferent conditioning during generation. Without any conditioning, the class-unconditional MAGE-B trained for 200 epochs yields only 14.23 FID and 57.7 IS. On the other hand, when conditioned on generated representations, MAGE-B achieves 5.07 FID and 142.5 IS, which significantly surpasses the class-unconditional baseline and further outperforms the class-conditional baseline in FID. This shows that representations could provide even more guidance than class labels. It is also quite close to the “upper bound” which is conditioned on oracle representations from ImageNet *real* images during pixel generation, demonstrating the effectiveness of RDM in generating realistic SSL representations.

Prior works in self-conditioned image generation have primarily focused on categorizing images into clusters within the representation space, using these clusters as pseudo class-conditioning [3, 34, 40]. We also evaluate the performance of this clustering-based conditioning in RCG, employing k -means within the Moco v3 ViT-B representation space to form 1000 clusters. Such conditioning achieves 6.60 FID and 121.9 IS, which falls short of the results achieved by conditioning on generated representations. This is because of the limited information contained within such discrete clusters, which is insufficient for providing detailed guidance for pixel generation. It is also important to note that this clustering approach relies on prior knowledge about the total number of classes, a piece of information that is often not available in general unlabeled datasets.

Conceptually, RCG’s pixel generator can integrate with various generative models. We validate this by testing ADM, LDM, and MAGE as pixel generators. As shown in Table 6b, conditioning on representations significantly improves the class-unconditional generation performance of all three generators. Additionally, Table 6c indicates that extending training epochs further improves performance, aligning with existing research [18, 39, 52]. These results show that RCG is a general self-conditioned image generation framework, seamlessly improving class-unconditional generation performance when combined with different modern generative models.

4.5. Computational Cost

In Table 7, we present a detailed evaluation of RCG’s computational costs, including the number of parameters, training costs, and generation throughput. The training cost is measured using a cluster of 64 V100 GPUs. The generation throughput is measured on a single V100 GPU. As LDM and ADM measure their generation throughput on a single NVIDIA A100 [52], we convert it to V100 throughput by assuming a $\times 2.2$ speedup of A100 vs V100 [54].

RCG-L uses a pre-trained Moco v3 ViT-L encoder, an RDM with 12 blocks and 1536 hidden dimensions, and a MAGE-L pixel generator. The training phase involves 200 epochs for the RDM and 800 epochs for the MAGE-L. Dur-

Table 7. **Computational cost on ImageNet 256×256.** RCG achieves a much smaller FID with similar or less computational cost as baseline methods.

Method	#Params (M)	Training Cost (days)	Epochs	Throughput (samples/s)	FID
LDM-8 [52]	395	1.2	150	0.9	39.13
ADM [18]	554	14.3	400	0.05	26.21
MAGE-B [39]	176	5.5	1600	3.9	8.67
MAGE-L [39]	439	10.7	1600	2.4	7.04
RCG-B	63+176	0.3+0.8	100+200	3.6	5.07
RCG-B	63+176	0.6+3.3	200+800	3.6	4.18
RCG-L	63+439	0.3+1.5	100+200	2.2	4.23
RCG-L	63+439	0.6+6.0	200+800	2.2	3.56

ing the generation process, the RDM undergoes 250 diffusion steps, while MAGE-L performs 20 parallel decoding steps. We also report RCG-B’s computational costs and FID with less training costs and smaller number of parameters (Moco v3 ViT-B as image encoder, MAGE-B as pixel generator). Given that the Moco v3 ViT encoder is pre-trained and not needed for generation, its parameters and training costs are excluded. As indicated in the table, the RDM module adds only minor costs in comparison to the pixel generator. This demonstrates RCG’s compatibility with modern generative models, highlighting its ability to enhance generation performance with minimal computational burdens.

4.6. Qualitative Results

Representation Reconstruction. Figure 6 illustrates RCG’s ability to generate images that align semantically with given representations. We extract SSL representations using examples from ImageNet 256×256. For each representation, we generate a variety of images by varying the random seed for the generation process. The images generated by RCG, while differing in specific details, consistently capture the semantic essence of the original images. This result highlights RCG’s capability to leverage semantic information in image representations to guide the generation process, without compromising the diversity that is important in unconditional image generation.

Representation Interpolation. Leveraging RCG’s dependency on representations, we can semantically transit between two images by linearly interpolating their respective representations. Figure 7 showcases such interpolation between pairs of ImageNet images. The interpolated images remain realistic across varying interpolation rates, and their semantic contents smoothly transition from one image to the other. This shows that the representation space of RCG is both smooth and semantically rich. This also demonstrates RCG’s potential in manipulating image semantics within a low-dimensional representation space, offering new possibilities to control image generation.

GT Image

Generated Images



Figure 6. RCG image generation results conditioned on representations extracted from images in ImageNet. The generated images follow the same semantics as the original image but with diverse appearances.

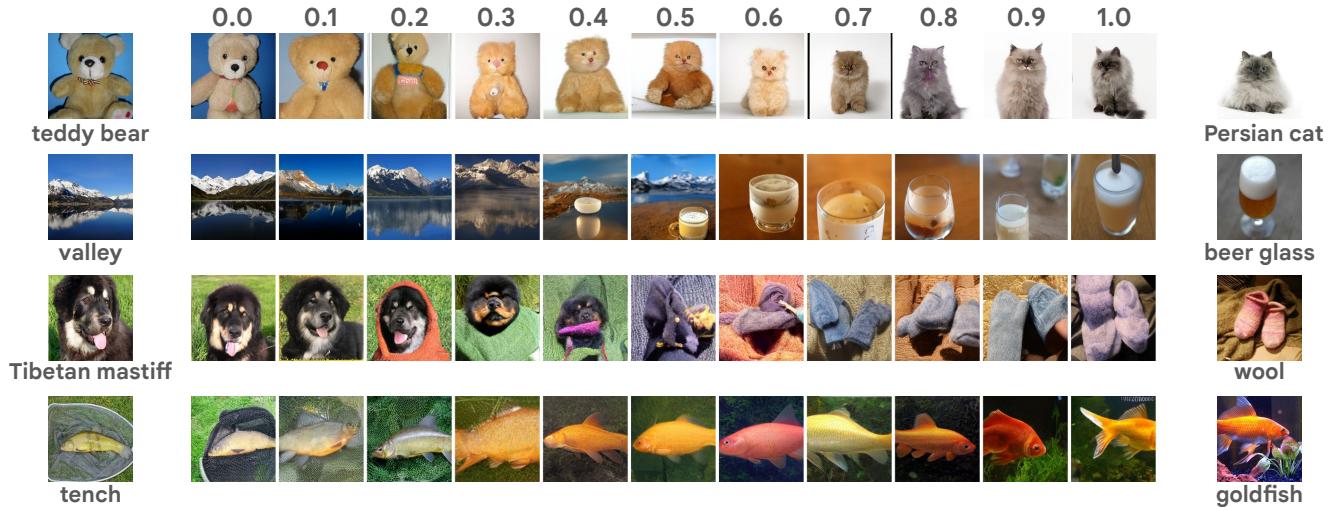


Figure 7. RCG image generation results conditioned on interpolated representations from two images. The semantics of the interpolated images gradually transfer between the two images.

5. Discussion

Computer vision has entered a new era where learning from extensive, unlabeled datasets is becoming increasingly common. Despite this trend, the training of image generation models still mostly relies on labeled datasets, which could be attributed to the large performance gap between conditional and unconditional image generation. Our paper addresses this issue by exploring self-conditioned image generation, which we propose as a nexus between conditional and unconditional image generation. We demonstrate that the long-standing performance gap can be effectively bridged by generating images conditioned on SSL representations

and leveraging a representation diffusion model to model and sample from this representation space. We believe this approach has the potential to liberate image generation from the constraints of human annotations, enabling it to fully harness the vast amounts of unlabeled data and even generalize to modalities that are beyond the scope of human annotation capabilities.

Acknowledgements. We thank Huiwen Chang, Saining Xie, Zhuang Liu, Xinlei Chen, and Mike Rabbat for their discussion and feedback. We also thank Xinlei Chen for his support on MoCo v3.

References

- [1] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.
- [2] Fan Bao, Chongxuan Li, Yue Cao, and Jun Zhu. All are worth words: a vit backbone for score-based diffusion models. In *NeurIPS 2022 Workshop on Score-Based Methods*, 2022.
- [3] Fan Bao, Chongxuan Li, Jiacheng Sun, and Jun Zhu. Why are conditional generative models better than unconditional ones? *arXiv preprint arXiv:2212.00362*, 2022.
- [4] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *arXiv preprint arXiv:2106.08254*, 2021.
- [5] Florian Bordes, Randall Balestrieri, and Pascal Vincent. High fidelity visualization of what your self-supervised representation knows about. *arXiv preprint arXiv:2112.09164*, 2021.
- [6] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.
- [7] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale GAN training for high fidelity natural image synthesis. In *Int. Conf. on Learning Representations (ICLR)*, 2019.
- [8] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European conference on computer vision (ECCV)*, pages 132–149, 2018.
- [9] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Int. Conference on Computer Vision (ICCV)*, pages 9650–9660, 2021.
- [10] Arantxa Casanova, Marlene Careil, Jakob Verbeek, Michal Drozdzal, and Adriana Romero Soriano. Instance-conditioned gan. *Advances in Neural Information Processing Systems*, 34:27517–27529, 2021.
- [11] Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.
- [12] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. Maskgit: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11315–11325, 2022.
- [13] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *icml*, pages 1597–1607. PMLR, 2020.
- [14] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.
- [15] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E Hinton. Big self-supervised models are strong semi-supervised learners. *Advances in Neural Information Processing Systems*, 33, 2020.
- [16] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. In *Int. Conference on Computer Vision (ICCV)*, pages 9640–9649, 2021.
- [17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [18] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021.
- [19] Jeff Donahue and Karen Simonyan. Large scale adversarial representation learning. *Advances in neural information processing systems*, 32, 2019.
- [20] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *Int. Conf. on Learning Representations (ICLR)*, 2021.
- [21] Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural networks*, 107:3–11, 2018.
- [22] Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Masked diffusion transformer is a strong image synthesizer. *arXiv preprint arXiv:2303.14389*, 2023.
- [23] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.
- [24] Ian J Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. 2014.
- [25] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent: A new approach to self-supervised learning. *arXiv preprint arXiv:2006.07733*, 2020.
- [26] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16000–16009, June 2022.
- [27] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738, 2020.
- [28] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.
- [29] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- [30] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.
- [31] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffu-

- sion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pages 6840–6851, 2020.
- [32] Jonathan Ho, Chitwan Saharia, William Chan, David J Fleet, Mohammad Norouzi, and Tim Salimans. Cascaded diffusion models for high fidelity image generation. *The Journal of Machine Learning Research*, 23(1):2249–2281, 2022.
- [33] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.
- [34] Vincent Tao Hu, David W. Zhang, Yuki M. Asano, Gertjan J. Burghouts, and Cees G. M. Snoek. Self-guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18413–18422, June 2023.
- [35] Zhicheng Huang, Xiaojie Jin, Chengze Lu, Qibin Hou, Ming-Ming Cheng, Dongmei Fu, Xiaohui Shen, and Jiashi Feng. Contrastive masked autoencoders are stronger vision learners. *arXiv:2207.13532v1*, 2022.
- [36] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019.
- [37] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [38] Tianhong Li, Peng Cao, Yuan Yuan, Lijie Fan, Yuzhe Yang, Rogerio S Feris, Piotr Indyk, and Dina Katabi. Targeted supervised contrastive learning for long-tailed recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6918–6928, 2022.
- [39] Tianhong Li, Huiwen Chang, Shlok Mishra, Han Zhang, Dina Katabi, and Dilip Krishnan. Mage: Masked generative encoder to unify representation learning and image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2142–2152, 2023.
- [40] Steven Liu, Tongzhou Wang, David Bau, Jun-Yan Zhu, and Antonio Torralba. Diverse image generation via self-conditioned gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14286–14295, 2020.
- [41] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.
- [42] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.
- [43] Mario Lučić, Michael Tschannen, Marvin Ritter, Xiaohua Zhai, Olivier Bachem, and Sylvain Gelly. High-fidelity image generation with fewer labels. In *International conference on machine learning*, pages 4183–4192. PMLR, 2019.
- [44] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *European Conference on Computer Vision*, pages 69–84. Springer, 2016.
- [45] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [46] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2536–2544, 2016.
- [47] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023.
- [48] Zhiliang Peng, Li Dong, Hangbo Bao, Qixiang Ye, and Furu Wei. Beit v2: Masked image modeling with vector-quantized visual tokenizers. *arXiv preprint arXiv:2208.06366*, 2022.
- [49] Konpat Preechakul, Nattanat Chatthee, Suttisak Wizadwongsu, and Supasorn Suwajanakorn. Diffusion autoencoders: Toward a meaningful and decodable representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10619–10629, 2022.
- [50] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.
- [51] Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.
- [52] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695, 2022.
- [53] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.
- [54] Dave Salvator. Nvidia developer blog. <https://developer.nvidia.com/blog/getting-immediate-speedups-with-a100-tf32>, 2020.
- [55] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.
- [56] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *Int. Conf. on Learning Representations (ICLR)*, 2021.
- [57] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2818–2826, 2016.
- [58] Changyao Tian, Chenxin Tao, Jifeng Dai, Hao Li, Ziheng Li, Lewei Lu, Xiaogang Wang, Hongsheng Li, Gao Huang, and Xizhou Zhu. Addp: Learning general representations for image recognition and generation with alternating denoising diffusion process. *arXiv preprint arXiv:2306.05423*, 2023.
- [59] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International conference on machine learning*, pages 10347–10357. PMLR, 2021.
- [60] Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.
- [61] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang,

- James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *arXiv preprint arXiv:2110.04627*, 2021.
- [62] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *Int. Conference on Machine Learning (ICML)*, pages 7354–7363, 2019.
- [63] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. StackGAN: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017.
- [64] Mingmin Zhao, Tianhong Li, Mohammad Abu Alsheikh, Yonglong Tian, Hang Zhao, Antonio Torralba, and Dina Katabi. Through-wall human pose estimation using radio signals. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7356–7365, 2018.
- [65] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. ibot: Image bert pre-training with online tokenizer. *arXiv preprint arXiv:2111.07832*, 2021.

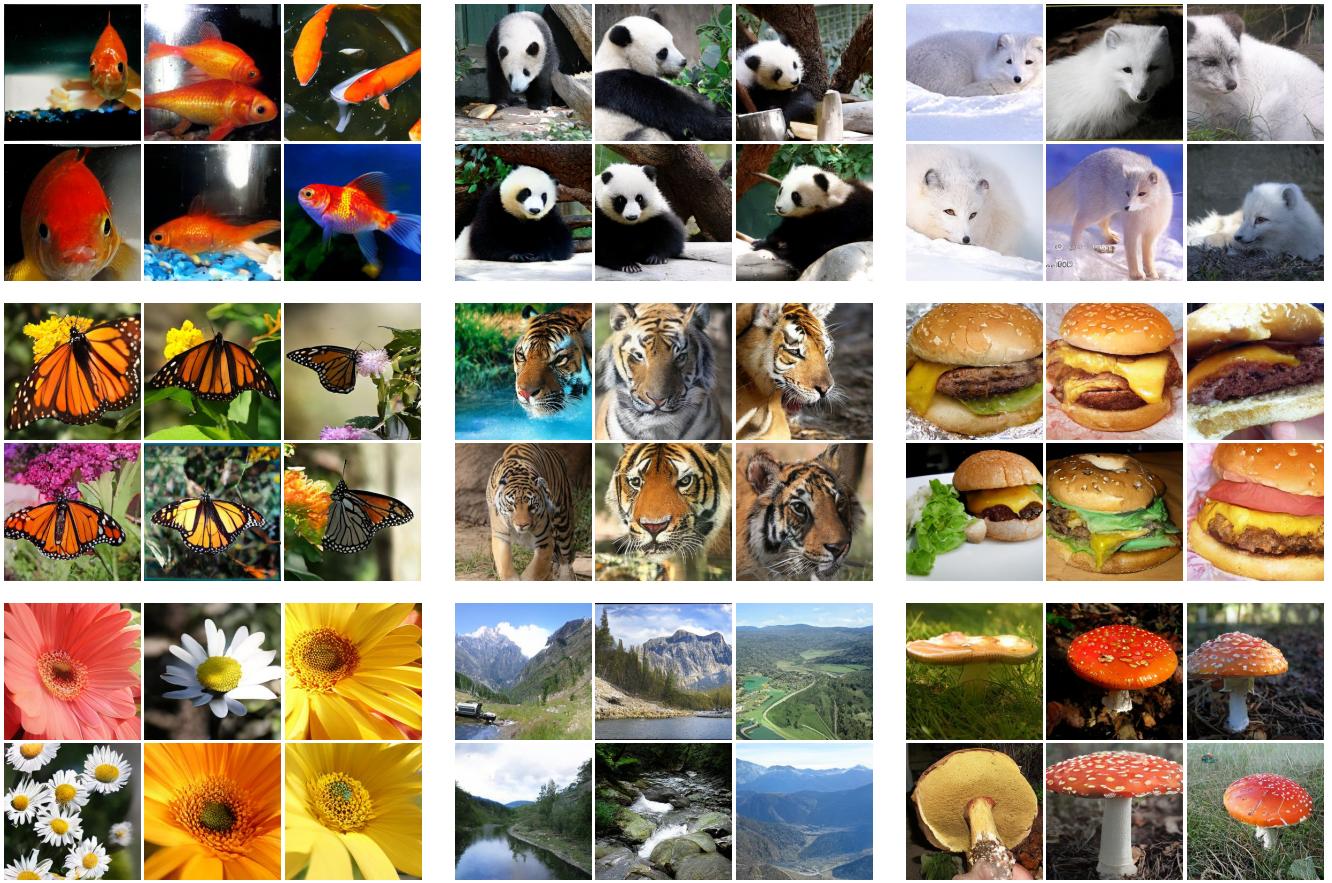


Figure 8. **RCG class-conditional image generation results on ImageNet 256×256.** Classes are 1: goldfish, 388: panda, 279: Arctic fox; 323: monarch butterfly, 292: tiger, 933: cheeseburger; 985: daisy, 979: valley, 992: agaric

Table 8. **Image generation performance on ImageNet 256×256.**
RCG seamlessly enables class-conditional image generation with a class-conditional RDM.

Methods	FID↓	Inception Score↑
Class-uncond. RCG-L	3.56	186.9
Class-uncond. RCG-L-G	3.31	253.4
Class-cond. RCG-L	3.49	215.5
Class-cond. RCG-L-G	3.90	300.7

A. Class-conditional Generation

While unconditional generation is particularly suited for large unlabeled datasets and less interpretable modalities, conditional generation offers the distinct advantage of allowing users to direct the model toward generating specific visual contents. This capability is important in numerous downstream generative applications. Unlike previous methods which typically need to re-train or fine-tune the entire network for different conditioning, RCG seamlessly facilitates conditional image generation by training a task-specific conditional RDM while using the same pixel generator. To

demonstrate this, we experiment with the common class-conditional image generation task on ImageNet. Specifically, in addition to the timestep embedding, a class embedding is added to each fully connected block of RDM, enabling the generation of representations for a specific class. The pixel generator then crafts the image pixels conditioned on the generated class-specific representation. As shown in Table 8 and Figure 8, this simple modification allows the user to control the class of the generated image while keeping RCG’s superior generation performance.

Importantly, conditional RCG does not need to re-train the representation-conditioned pixel generator, largely reducing the training overheads. For any new conditioning, only the lightweight RDM needs to be re-trained. This enables pre-training of the SSL encoder and pixel generator on large-scale unlabeled datasets, and task-specific training of conditional RDM on a small-scale labeled dataset. We believe that this property, similar to self-supervised learning, allows RCG to both benefit from large unlabeled datasets and adapt to various downstream generative tasks with minimal overheads.

Table 9. RDM implementation details.

config	value
#blocks	12
hidden dimension	1536
#params	63M
optimizer	AdamW [42]
learning rate	5.12e-4
weight decay	0.01
optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.999$
batch size	512
learning rate schedule	constant
training epochs	200
augmentation	Resize(256) RandCrop(256) RandomFlip (0.5)
diffusion steps	1000
noise schedule	linear
DDIM steps	250
η	1.0

B. Implementation Details

In this section, we describe the implementation details of RCG, including hyper-parameters, model architecture, and training paradigm. Our code and pre-trained model weights are also released at <https://github.com/LTH14/rcg>.

Image Encoder. We use Vision Transformers (ViTs) [20] pre-trained with Moco v3 [16] as the default image encoder. We evaluate three ViT variants (ViT-S, ViT-B, and ViT-L) in the main paper, each trained on ImageNet for 300 epochs. We utilize the image representation after the MLP projection head, favoring its adjustable dimensionality. An output dimension of 256 has proven the most effective. The representation of each image is normalized by its own mean and variance. Detailed training recipes of our pre-trained image encoder can be found in [16].

Representation Diffusion Model (RDM). Our RDM architecture employs a backbone of multiple fully connected blocks. We use 12 blocks and maintain a consistent hidden dimension of 1536 across the network. The timestep t is discretized into 1000 values, each embedded into a 256-dimensional vector. For class-conditional RDM, we embed each class label into a 512-dimensional vector. Both timestep and class label embeddings are projected to 1536 dimensions using different linear projection layers in each block. Detailed hyper-parameters for RDM’s training and generation can be found in Table 9.

Pixel Generator. We experiment with ADM [18], LDM [52] and MAGE [39] as the pixel generator. For representation-conditioned ADM and LDM, we substitute the original class embedding conditioning with the image representation. We follow ADM’s original training recipe to train representation-conditioned ADM for 100 epochs. We follow LDM-8’s original training recipe, with modifications including a batch size of 256, a learning rate

Table 10. Rep.-conditioned MAGE implementation details.

config	value
optimizer	AdamW [42]
base learning rate	1.5e-4
weight decay	0.05
optimizer momentum	$\beta_1, \beta_2 = 0.9, 0.95$
batch size	4096
learning rate schedule	cosine decay [41]
warmup epochs	10
training epochs	800
gradient clip	3.0
label smoothing [57]	0.1
dropout	0.1
augmentation	Resize(256) RandCrop(256) RandomFlip (0.5)
masking ratio min	0.5
masking ratio max	1.0
masking ratio mode	0.75
masking ratio std	0.25
rep. drop rate	0.1
parallel-decoding temperature	6.0 (B) 11.0 (L)
parallel-decoding steps	20
guidance scale (τ)	1.0 (B) 6.0 (L)
guidance scale schedule	linear [11]

of 6.4e-5, and a training duration of 40 epochs. For representation-conditioned MAGE, we replace the default “fake” class token embedding [C_0] with the image representation for conditioning. Our implementation of classifier-free guidance follows Muse [11], incorporating a linear guidance scale scheduling. Detailed hyper-parameters for our representation-conditioned MAGE are provided in Table 10.

C. Additional Qualitative Results

We include more qualitative results, including class-unconditional image generation (Figure 9), class-conditional image generation (Figure 10), and the comparison between generation results with or without classifier-free guidance (Figure 11). All results demonstrate RCG’s superior performance in image generation.

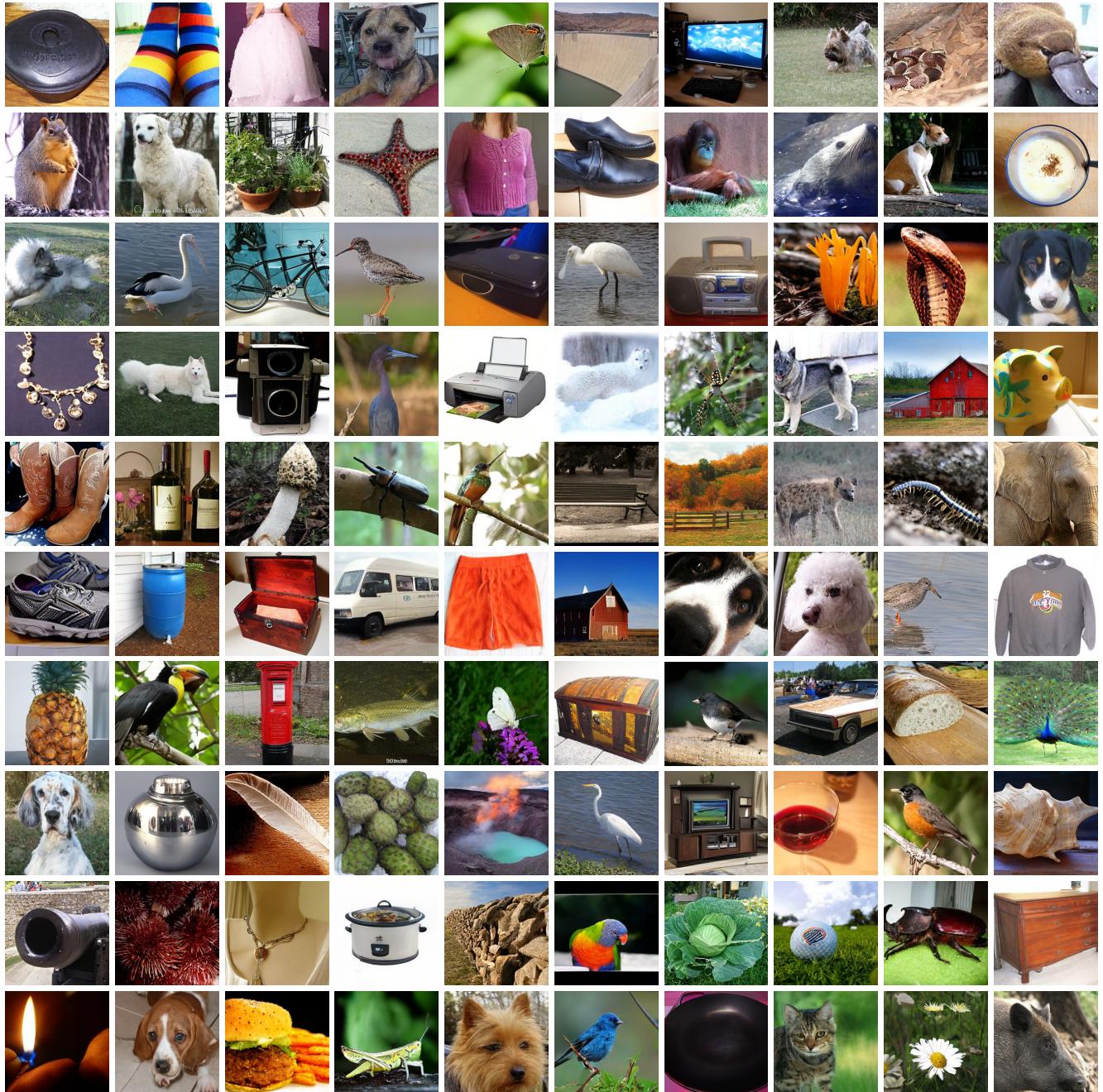


Figure 9. More RCG class-unconditional image generation results on ImageNet 256×256.

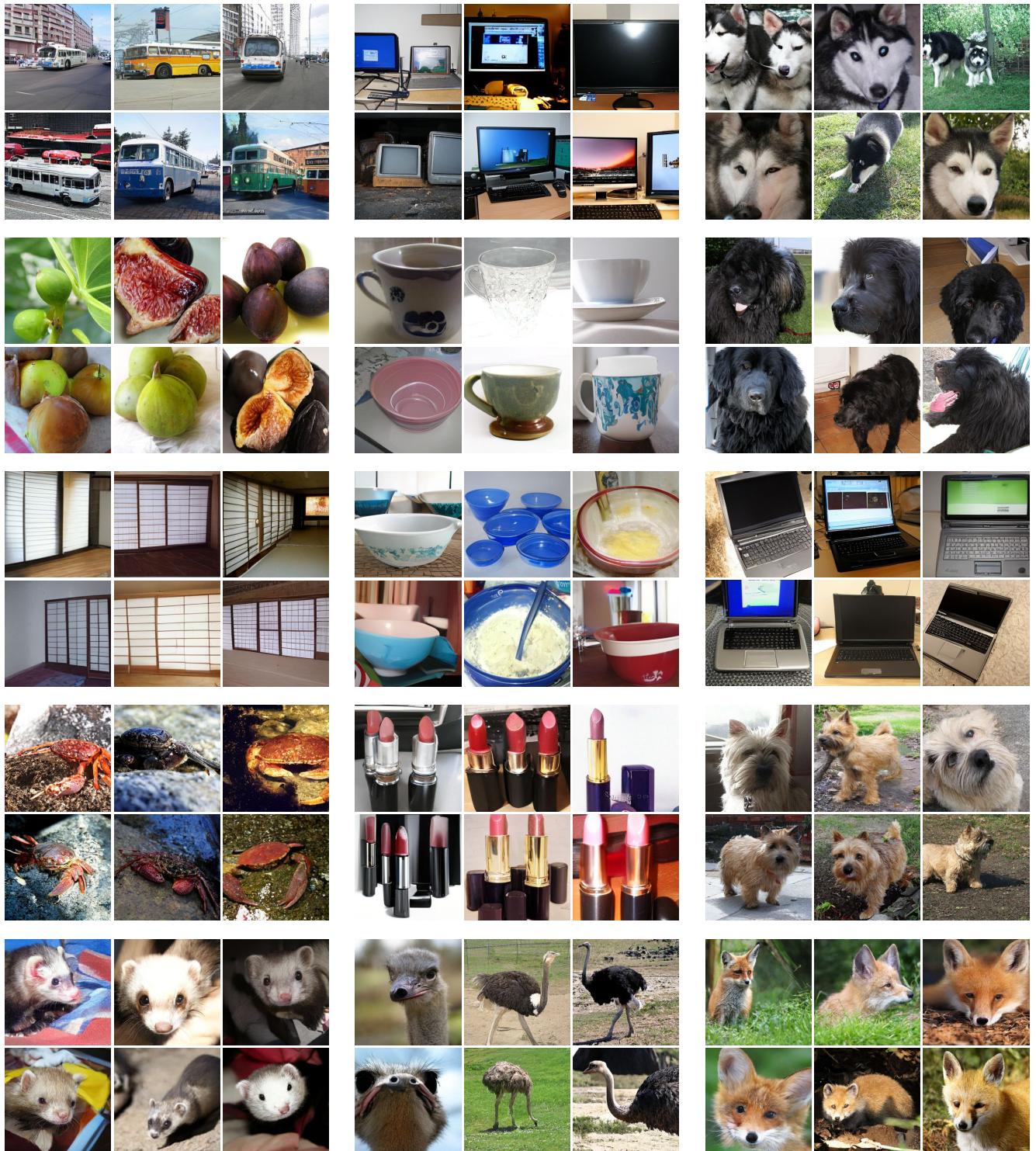


Figure 10. **More RCG class-conditional image generation results on ImageNet 256×256.** Classes are 874: trolleybus, 664: monitor, 249: malamute; 952: fig, 968: cup, 256: Newfoundland; 789: shoji, 659: mixing bowl, 681: notebook; 119: rock crab, 629: lipstick, 192: cairn; 359: ferret, 9: ostrich, 277: red fox.

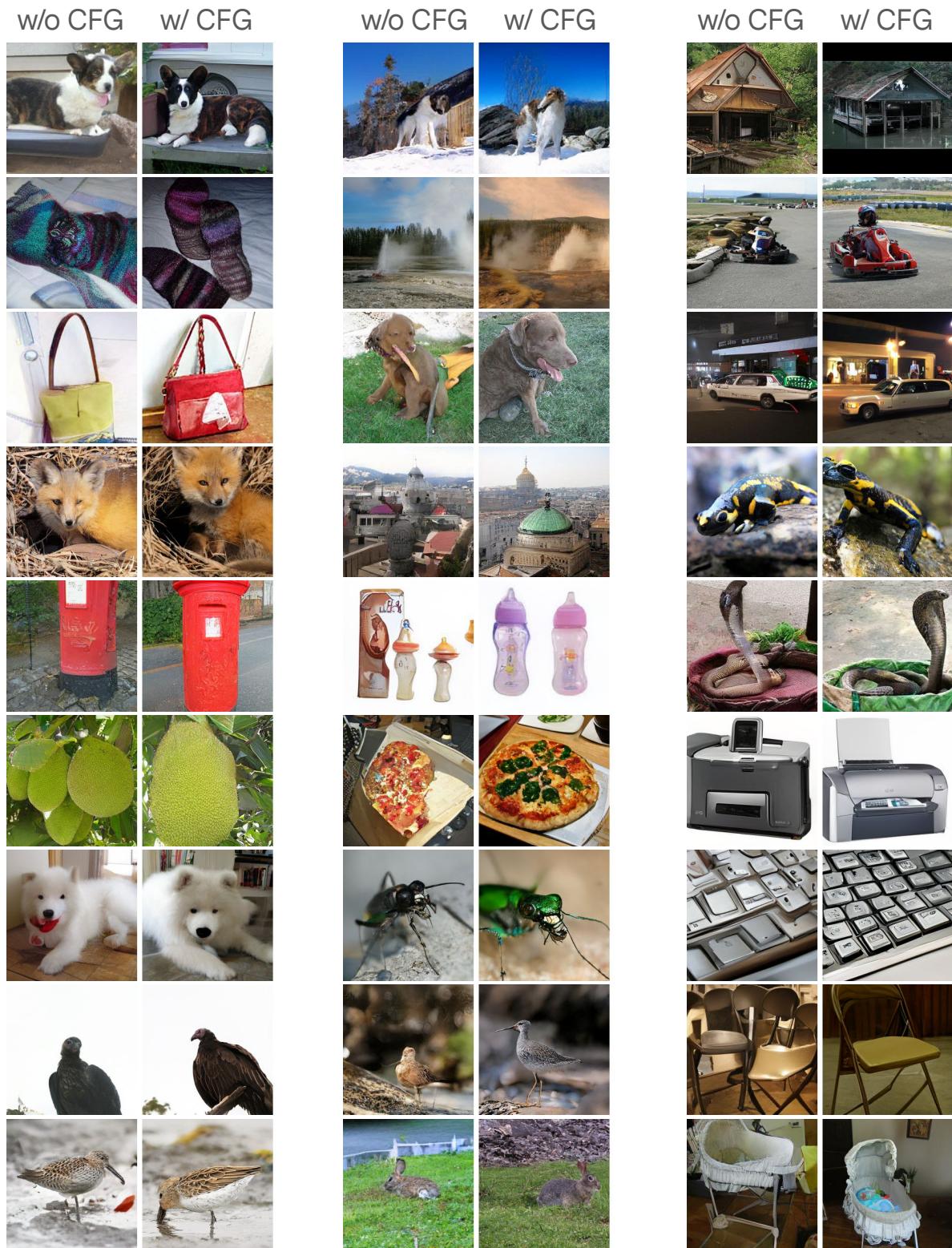


Figure 11. **Class-unconditional image generation results on ImageNet 256×256, with or without classifier-free guidance.** RCG achieves strong generation performance even without CFG. Incorporating CFG further improves the generation quality.