Frido: Feature Pyramid Diffusion for Complex Scene Image Synthesis

Wan-Cyuan Fan[1*], Yen-Chun Chen[2†],
DongDong Chen[2], Yu Cheng[2], Lu Yuan[2], Yu-Chiang Frank Wang[1, 3]

[1]National Taiwan University
[2]Microsoft Corporation
[3]NVIDIA
r09942092@ntu.edu.tw

Abstract

Diffusion models (DMs) have shown great potential for high-quality image synthesis. However, when it comes to producing images with complex scenes, how to properly describe both image global structures and object details remains a challenging task. In this paper, we present *Frido*, a **F**eatu**r**e Pyram**id** Diffusi**o**n model performing a multi-scale coarse-to-fine denoising process for image synthesis. Our model decomposes an input image into scale-dependent vector quantized features, followed by a coarse-to-fine modulation for producing image output. During the above multi-scale representation learning stage, additional input conditions like text, scene graph, or image layout can be further exploited. Thus, *Frido* can be also applied for conditional or cross-modality image synthesis. We conduct extensive experiments over various unconditioned and conditional image generation tasks, ranging from text-to-image synthesis, layout-to-image, scene-graph-to-image, to label-to-image. More specifically, we achieved state-of-the-art FID scores on five benchmarks, namely layout-to-image on COCO and OpenImages, scene-graph-to-image on COCO and Visual Genome, and label-to-image on COCO.

Introduction

Generating photo-realistic images is a critical task in computer vision research. In this task, a generative model is designed to learn the underlying data distribution of a given set of images and to be capable of synthesizing new samples from the learned distribution. To this end, series of methods were proposed, including VAEs (Kingma and Welling 2014; Van Den Oord, Vinyals et al. 2017), GANs (Goodfellow et al. 2014; Radford, Metz, and Chintala 2015), flow-based methods (Dinh, Krueger, and Bengio 2014; Kingma and Dhariwal 2018), and the trending diffusion models (DMs) (Sohl-Dickstein et al. 2015; Ho, Jain, and Abbeel 2020). The quality of the generated images has been improved rapidly with the contribution of these lines of works. Moreover, the task itself also evolves from object-centric image synthesis without conditions to complex scene image generation, and sometimes based on multi-modal conditions (e.g., texts, layouts, labels, and scene-graphs).

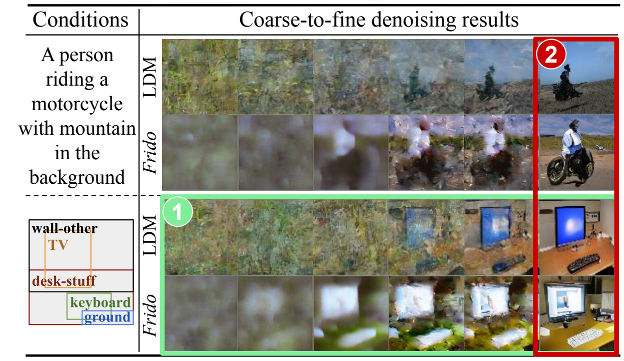

Conditions
Coarse-to-fine denoising results
A person riding a motorcycle with mountain in the background
LDM
Frido
2
1
wall-other
TV
desk-stuff
keyboard
ground
LDM
Frido

Figure 1: Illustration of *Frido*. Given a cross-modal condition, *Frido* generates images in ① a coarse-to-fine manner from structure to object details, producing outputs with ② high semantic correctness and quality. Note that existing models such as the LDMs are not designed to distinguish between high/low-level visual information.

Recently, diffusion models (Ho, Jain, and Abbeel 2020; Nichol and Dhariwal 2021; Ho et al. 2022; Rombach et al. 2022; Ramesh et al. 2022) have demonstrated a remarkable capability of high-quality image synthesis and outperform other classes of generative approaches on multiple tasks, including but not limited to unconditional image generation, text-to-image generation, and image super-resolution. Despite the encouraging progress, diffusion models may fall short when targeted images are more complex and conditioning inputs are highly abstractive. The composition of objects and parts, along with high-level semantic relations are prevailing in those tasks, which are less seen in earlier object-centric benchmarks and may be essential to higher quality generation.

In particular, we point out two major challenges in existing DM works. *First*, most of existing DMs deal with feature maps or image pixels at a single scale/resolution, which might not be able to capture image semantics or compositions in real-world complex scenes. Take the first row of Figure 1 as examples, it can be seen that while LDM (Rombach et al. 2022) generates images containing a "person" given the text condition, semantic structures of "riding a motorcycle" and "mountain in the background" are not sufficiently

[*] work done during research internship at Microsoft
[†] project lead

produced. *Second*, expensive computational resources are typically required for DMs during training and testing due to the iterative denoising processes, especially for producing high-resolution outputs. This not only limits the accessibility but also results in massive carbon emissions. Therefore, a computationally efficient diffusion model that leverages coarse/high-level synthesized outputs for introducing multi-scale visual information would be desirable.

To address these limitations, we propose ***Frido***, a **F**eatu**r**e Pyram**i**d **D**iffusi**o**n model for complex scene image generation.[1] *Frido* is a novel multi-scale coarse-to-fine diffusion and denoising framework, which allows synthesizing images with enhanced global structural fidelity and realistic object details. Specifically, we introduce a novel *feature pyramid U-Net* (PyU-Net) with a *coarse-to-fine modulation* design, enabling our model to denoise visual features from multiple spatial scales in a top-down fashion. These multi-scale features are produced by our MS-VQGAN, a newly designed multi-scale variant of VQGAN (Esser, Rombach, and Ommer 2021) that encodes images into multi-scale visual features (discrete latent codes). As can be seen in Figure 1, as the feature gradually being denoised, the images are reconstructed in a coarse-to-fine manner (decoded by our MS-VQGAN decoder), from global structures to fine-grained details. On the other hand, a recent competitive diffusion model (Rombach et al. 2022) reconstructs images uniformly across spatial scales.

Frido is a generic diffusion framework that can synthesize images from diverse, multi-modal inputs, including texts, box-layouts, scene-graphs, and labels. Moreover, our model introduces minimal extra parameters while allowing us to speed up the notoriously slow inference of conventional DMs. Extensive experiments are done to demonstrate the effectiveness of the new designs. Our contributions are summarized as follows. (*i*) We propose *Frido*, a novel diffusion model to generate photo-realistic images from multi-modal inputs, with a *coarse-to-fine* prior that is under-explored in the DM paradigm. (*ii*) Empirically, we achieve *5 new state-of-the-art* results, including layout-to-image on COCO and OpenImages, scene-graph-to-image on COCO and Visual Genome, and label-to-image on COCO, all are complex scenes with highly abstractive conditions. (*iii*) In practice, *Frido inferences fast*, shown by a head-to-head comparison with an already fast diffusion model, the LDM.

Preliminary

Multiple lines of works to generate photo-realistic images have been proposed, including VAEs, GANs, and Invertible-Flows, and achieved impressive results for object-centric images. However, VAEs suffer from blurry outputs. GANs are notoriously hard to train and lack diversity. Flow-based model suffers shape distortions due to imperfect inverse transform. Our work belongs to the paradigm of diffusion models (DMs), which have been shown to best synthesize high quality images among all deep generative methods. For completeness, we summarize the fundamentals of DMs and a recent improvement, Latent Diffusion Models (LDMs) (Rombach et al. 2022).

Diffusion Models for Image Generation A diffusion model (DM) contains two stages: forward (diffusion) and backward (denoising) processes. In the forward process, the given data $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ is gradually destroyed into an approximately standard normal distribution $\mathbf{x}_T \sim p(\mathbf{x}_T)$ over T steps, where q and p denote the given data manifold and the standard Gaussian distribution, respectively; and $\mathbf{x}$ denotes a data point from q. The diffusion process, formulated by Ho, Jain, and Abbeel (2020), are shown as follows:

$$\begin{aligned} q(\mathbf{x}_{1:T}|\mathbf{x}_0) &= \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}), \text{ and} \\ q(\mathbf{x}_t|\mathbf{x}_{t-1}) &= \mathcal{N}(\sqrt{1-\beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}). \end{aligned} \tag{1}$$

, where β denotes the noise schedule. Can be fixed or learned. By reversing the forward process, Ho, Jain, and Abbeel (2020) obtained the backward process:

$$\begin{aligned} p_\theta(\mathbf{x}_{0:T}) &= p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t), \text{ and} \\ p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) &= \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sigma_\theta(\mathbf{x}_\mathbf{t}, t)). \end{aligned} \tag{2}$$

, where θ denotes the learnable parameters, a U-Net (Ronneberger, Fischer, and Brox 2015) in Ho, Jain, and Abbeel (2020). This is implemented by a neural network predicting each of the denoising steps; and it can be viewed as a Markov chain with a learned Gaussian transition distribution (Dhariwal and Nichol 2021; Pandey et al. 2022).

In practice, we randomly sample a timestep t in $[0, T]$, and then compute $\mathbf{x}_t$ by interpolating $\mathbf{x}_0$ and ϵ with the weight schedule β_t, where ϵ is sampled Gaussian noise. The denoising network ϵ_θ is trained by the following loss:

$$\mathcal{L}_{DM} = \mathbb{E}_{\mathbf{x}_0,\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}_t)\|^2\right]. \tag{3}$$

At a higher level, this loss trains the network to predict the step noise ϵ applied on $\mathbf{x}_{t-1}$ given $\mathbf{x}_t$. To synthesize an image, one can run this denoising network for T steps to gradually denoise a random noise image.

Latent Diffusion Models Most DMs (Nichol et al. 2022; Dhariwal and Nichol 2021) operate on the original image pixels, yielding high dimensional data manifold with input $\mathbf{x}_0 \in \mathcal{R}^{3\times H\times W}$. Such high-dimensional inputs cost huge computation for the diffusion and denoising processes at both training and inference. Very recently, Latent Diffusion Models (LDMs) (Rombach et al. 2022) are proposed to adopt DMs to learn the low-dimensional latent codes, encoded by a VQGAN (Esser, Rombach, and Ommer 2021) or KL-autoencoder (Rombach et al. 2022). Given an image $\mathbf{x}_0$ and the pre-trained autoencoder, containing encoder $\mathcal{E}$ and decoder $\mathcal{D}$, the corresponding latent codes $\mathbf{z_0} = \mathcal{E}(\mathbf{x_0})$ can be produced, where $\mathbf{z}_0 \in \mathcal{R}^{c\times h\times w}$, c is usually set to 4; and h, w are downsampled $8 - 16$ times from H, W. By replacing the image data point $\mathbf{x}$ in Eq. (1) and Eq. (2) with the encoded latent $\mathbf{z}$, the diffusion and denoising processes of a

[1] *Frido* is pronounced as “free-dow”.

LDM can be derived:

$$q(\mathbf{z}_{1:T}|\mathbf{z}_0) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1}), \text{ and}$$
$$p_\theta(\mathbf{z}_{0:T}) = p(\mathbf{z}_T)\prod_{t=1}^{T} p_\theta(\mathbf{z}_{t-1}|\mathbf{z}_t). \tag{4}$$

At inference, the final output image can be reconstructed from the denoised latent $\tilde{\mathbf{x}}_0 = \mathcal{D}(\tilde{\mathbf{z}}_0)$, where $\tilde{\mathbf{z}}_0$ is sampled and denoised using Eq. (4). Since T is typically set to $500 - 1000$ in practice, and the autoencoding is a one-time operation per image, the overall computation is greatly reduced due to the much lower resolution of $\mathbf{z}_0$.

Methodology

Although existing DMs generate high-resolution images for a single object with outstanding quality, most of them only deal with feature maps or image pixels at a single resolution. Since they treat high and low-level visual concepts equally, it is not easy for such DM models to describe the corresponding image semantics or composition. This might limit their uses for synthesizing complex scene images.

To enhance DMs with global structural modeling, we propose to model the latent features in a coarse-to-fine fashion via feature pyramids. We first introduce the Multi-Scale Vector Quantization model (MS-VQGAN), which encodes the image into latent codes at several spatial levels. Next, we propose the feature pyramid diffusion model (*Frido*), extending the diffusion and denoising into a multi-scale, coarse-to-fine fashion. To achieve these, we design a new feature Pyramid U-Net (PyU-Net), equipped with a special modulation mechanism to allow coarse-to-fine learning. In this section, we introduce each component in detail.

Learning Multi-Scale Perceptual Latents

Before we model an image in a coarse-to-fine fashion, we first encode it into latent codes with several spatial resolutions. Extending from VQGAN (Esser, Rombach, and Ommer 2021), we train a multi-scale auto-encoder, named MS-VQGAN, with a feature pyramid encoder $\mathcal{E}$ and decoder $\mathcal{D}$. As shown in Figure 2a, given an image $\mathbf{x}_0$, the encoder $\mathcal{E}$ firstly produces a latent feature map set of N scales $\mathcal{Z} = \mathcal{E}(\mathbf{x}_0) = \{\mathbf{z}^{1:N}\}$, where $\mathbf{z}^t \in \mathcal{R}^{c\times \frac{s}{2^{t-1}} \times \frac{s}{2^{t-1}}}$. Note that N and c denote the number of feature maps (stages) and the channel size of the feature, respectively; and s represents the size of the largest feature map. In this design, we are encouraging $\mathbf{z}^1$ to preserve lower-level visual details and $\mathbf{z}^N$ to represent higher-level shape and structures. Secondly, after quantizing and fusing, we upsample these features to the same shape, concat them, and feed them into the decoder $\mathcal{D}$ and reconstruct the image $\mathcal{D}(\mathcal{Z}) = \tilde{\mathbf{x}}_\mathbf{0}$. The objective for this auto-encoder module is the weighted sum of l_2 loss between $\mathbf{x}_0$ and $\tilde{\mathbf{x}}_0$, and other perceptual losses[2] in VQGAN.

We highlight that, with this design, MS-VQGAN can not only encode the input image into multi-scale codes of different semantic levels but also preserve more structure and detail, as later analyzed in Section of model ablation.

Feature Pyramid Latent Diffusion Model

After the MS-VQGAN is trained, we can use it to encode an image into multi-level feature maps $\mathcal{Z}$. Next, we introduce the feature Pyramid Diffusion Model (*Frido*) to model the underlying feature distribution and then generate images from noises. Similar to other DMs, *Frido* contains two parts: the *diffusion process* and the *denoising process*.

Diffusion Process of *Frido* Instead of naively adding noises simultaneously on all N feature scales $\mathcal{Z} = \{\mathbf{z}^1, ..., \mathbf{z}^N\}$ at each of the T steps, we conduct diffusion process sequentially from low-level ($\mathbf{z}^1$) to high-level ($\mathbf{z}^N$), and each level takes T diffusion steps (total of $N \times T$ timesteps). See the top half of Figure 2b for an illustration.

Different from the classical diffusion process that corrupts pixels into noise in an unbiased way, we observe that *Frido*'s diffusion process starts from corrupting the object details, object shape, and finally the structure of the entire image. This allows *Frido* to capture information in different semantic levels. See Fig 1 for qualitative examples.

Denoising Process of *Frido* In the denoising phase, a sequence of neural function estimator $\epsilon_{\theta,t,n}$ is trained, where $t = 1, 2, \ldots, T$ and $n = N, N-1, \ldots, 1$. In order to denoise scale-by-scale, we introduce a novel feature pyramid U-Net (PyU-Net) as the neural approximator. PyU-Net can denoise the multi-scale features from high-level $\mathbf{z}^N$ to low-level $\mathbf{z}^1$ sequentially, achieving a coarse-to-fine generation. We highlight that, different from the LDMs, our PyU-Net is more suitable for coarse-to-fine diffusion with these two novel features: (1) *shared U-Net* with *lightweight level-specific layers* that project features of different levels to a shared space so that the heavier U-Net can be reused across all levels, reducing the trainable parameters, and (2) *coarse-to-fine modulation* to condition the denoising of low-level features on high-level ones that are already generated.

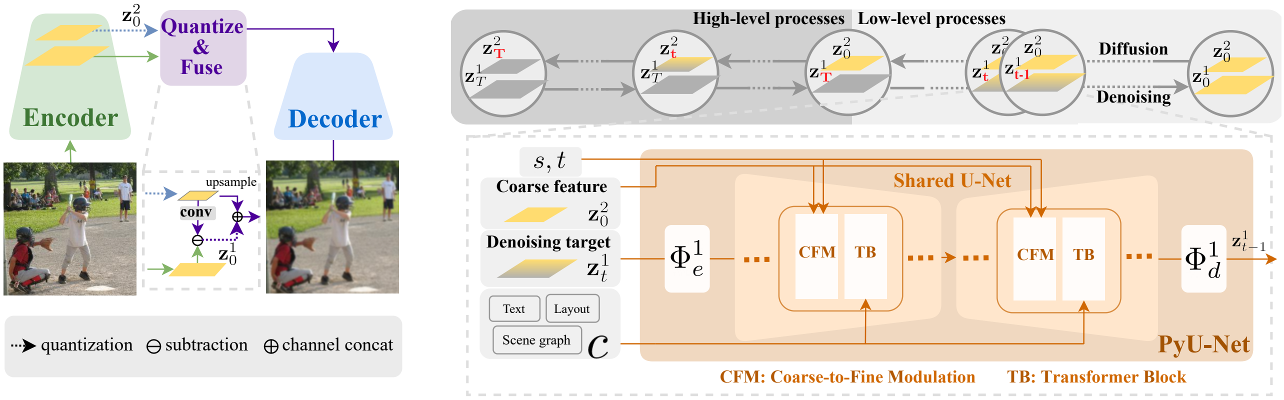
Feature Pyramid U-Net The proposed PyU-Net learns the denoising process in a coarse-to-fine fashion. Take $N = 2$ ($\mathcal{Z} = \{\mathbf{z}_0^1, \mathbf{z}_0^2\}$) as an example (shown in Figure 2(b)), PyU-Net takes 4 inputs: (1) stage s and timestep t embeddings, (2) high-level feature conditions $\mathbf{z}_0^2$, (3) target feature map $\mathbf{z}_t^1$, and (4) other cross-modal conditions $\mathbf{c}$. By jointly observing these inputs, PyU-Net predicts the noise ϵ applied on the target feature $\mathbf{z}_t^1$, as shown in Figure 2b.

Instead of using a separate U-Net for each stage n, we opt for a single shared U-Net to reduce the parameter count. The input denoising target $\mathbf{z}_t^1$ is first projected by level-specific layers Φ_e^1 into a shared space so that a shared U-Net can be applied. Finally, another level-specific projection Φ_d^1 decodes the U-Net output to predict the noise ϵ added on $\mathbf{z}_t^1$, with the following objective similar to Eq. (3):

$$\mathcal{L}_{Frido} = \mathbb{E}_{z_0^n,\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{z}_t^n, \mathbf{z}_0^{n+1:N}, t)\|^2\right]. \tag{5}$$

We note that PyU-Net not only reduces the trainable parameters but also improves the results compared to vanilla per-stage U-Nets. For analysis, please refer to the experiments. Also, for training efficiency, we adopt the teacher forcing trick similar to sequence-to-sequence language models (Brown et al. 2020), where ground truth feature conditions are used while denoising the low-level map.

[2] Patch discriminator loss and perceptual reconstruction loss.



(a) Architecture of MS-VQGAN.

(b) Details of the diffusion and denoising processes.

Figure 2: Overview of *Frido* (best viewed in color). How MS-VQGAN encodes an image into multi-scale feature maps $\mathbf{z}_0^1, \mathbf{z}_0^2$ is illustrated in (a). The quantization enables VQ-VAE learning; and the fusion allows merging all representations from high to low level for the decoder to reconstruct an image. The upper half of (b) demonstrate the coarse-to-fine process, where the denoising is completed for high-level first, and then the lower one. The lower half of (b) details each denoising step. A U-Net is shared not only across timestep t but also the scale level s. Coarse-to-fine gating will be explained in Figure 3.

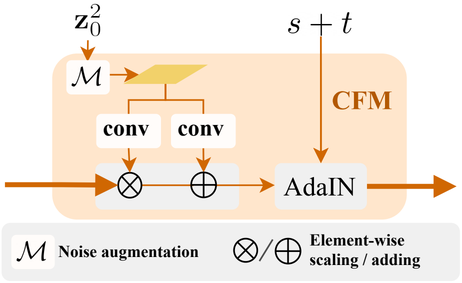

$\mathbf{z}_0^2$
$s + t$
$\mathcal{M}$
CFM
conv
conv
AdaIN
$\mathcal{M}$ Noise augmentation
$\otimes / \oplus$ Element-wise scaling / adding

Figure 3: Framework of coarse-to-fine modulation in PyU-Net. Note that we ignore some intermediate convolution layers and SiLU layers for simplification.

Coarse-to-Fine Modulation *Frido* produces the latent codes sequentially from high-level to low-level feature maps. For example, while generating $\mathbf{z}_t^1$ (low-level), the model is conditioned on $\mathbf{z}_0^2$ (high-level). We, therefore, introduce a coarse-to-fine modulation as shown in Figure 3.

Our coarse-to-fine modulation (CFM) is designed to incorporate (1) 2D high-level features, and (2) 1D stage and time embedding into residual blocks, allowing *Frido* to have the high-level feature as well as stage-temporal awareness. Therefore, in our proposed CFM, there are two types of modulation conducted upon normalized features sequentially, between which an extra convolution (conv) and SiLU layer (Elfwing, Uchibe, and Doya 2018) are inserted.

Specifically, given the high-level ground truth $\mathbf{z}_0^2$, we apply noise augmentation by $\mathcal{M}(\mathbf{z}_0^2) = f_z$, where $\mathcal{M}(\mathbf{z}_0^n) = \alpha \cdot \mathbf{z}_0^n + (1-\alpha) \cdot \epsilon$. We note that $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, and the scaler α is a hyper-parameter. After that, assume the input of CFM to be f_i, in the first modulation, we modulate the normalized feature norm(f_i) with 2D scaling and shifting parameters from high-level feature f_z with two convs respectively, producing intermediate representation h as follows:

$$h = \text{conv} \circ \text{SiLU}(\text{conv}(f_z) \times \text{norm}(f_i) + \text{conv}(f_z)). \quad (6)$$

In the second modulation, to equip U-Net with stage-temporal awareness, we further modulate h with 1D stage+time embedding and produce output f_o, similar to Eq. 6. Note that we use linear layers to transform $s + t$ and also add a conv and SiLU after the AdaIN (Huang and Belongie 2017).

To summarize, we highlight that our PyU-Net framework equips DM with the ability to learn in a coarse-to-fine fashion with a moderate increase of parameters compared to classical hierarchical learning strategy (Razavi, Van den Oord, and Vinyals 2019). *Frido* inherits three generative paradigms, VAE, GAN, and DM, and is further embedded with a coarse-to-fine prior. Moreover, the diffusion operates on lower-resolution maps first, resulting a speedup at inference. Later, we show that SOTA results can be achieved under a similar compute budget to a strong, fast DM.

Experiments

In this section, we empirically demonstrate that *Frido* generates high-quality complex scene images that are also consistent to the multi-modal conditions, through the lens of text-to-image, scene-graph-to-image, and label-to-image generation tasks. Moreover, to emphasize the capability of capturing multiple objects in the images globally, we conduct experiments on layout-to-image generation. Lastly, extensive analyses are performed to validate design choices. We show that *Frido* achieved state-of-the-art FID scores on multiple tasks under 5 settings with improved inference speed.

Notations *Frido* can be trained with different feature resolutions and levels. For simplicity and readability, a latent feature map where each feature corresponding to $n \times n$ original image pixels is denoted fn. For example, a *Frido* to generate

Methods	FID↓	IS↑	CLIP↑
Methods under standard T2I setting			
AttnGAN (Xu et al. 2018)	33.10	23.61	-
Obj-GAN (Li et al. 2019)	36.52	24.09	-
DM-GAN (Zhu et al. 2019)	27.34	**32.32**	-
DF-GAN (Tao et al. 2022)	21.42	-	-
LDM-8† (Rombach et al. 2022)	17.61	19.34	0.6500
VQ-diffusion‡ (Gu et al. 2022)	14.06	21.85	0.6770
LDM-8-G†	12.27	27.86	0.6927
Frido-f16f8	15.38	19.32	0.6607
Frido-f16f8-G	**11.24**	26.82	**0.7046**
Methods with external pre-trained CLIP			
LAFITE-CLIP‡ (Zhou et al. 2022)	**8.12**	**32.24**	0.7915
Frido-f16f8-G-CLIPr	8.97	27.43	**0.7991**

Table 1: Text-to-image generation on COCO. For LDM scores, $T = 250$; for *Frido*, $T = 200$. †: reproduced with official code and configs. ‡: obtained from official model checkpoints. G: classifier-free guidance with scale = 2.0. Note that LAFITE used CLIP at training, while *Frido* uses it at inference only (CLIPr).

256×256 images using 32×32 high-level and 64×64 low-level latent code is denoted *Frido*-f8f4. For LDM baselines, LDM-n encodes $n \times n$ pixels per feature.

Datasets and Evaluation

The main tasks we considered are text-to-image generation (T2I) on COCO (Lin et al. 2014), scene-graph-to-image generation (SG2I) on COCO-stuff and Visual Genome (VG) (Krishna et al. 2017), label-to-image generation (Label2I) (Jyothi et al. 2019) on COCO-stuff (Lin et al. 2014), and layout-to-image generation (Layout2I) on COCO-stuff and OpenImages (Kuznetsova et al. 2020). The standard metrics used to evaluate image synthesis tasks are Fréchet inception distance (FID) (Heusel et al. 2017) and Inception score (IS) (Salimans et al. 2016). In addition, we considered other task-specific metrics such as CLIP score (Hessel et al. 2021), Precision and Recall (Sajjadi et al. 2018), SceneFID (Sylvain et al. 2021), YOLO score (Li et al. 2021), PSNR (Hore and Ziou 2010), and SSIM (Wang et al. 2004) when applicable. Please see the supplementary[3] for detailed settings. For completeness, we also conducted user preference studies and experimented on unconditional image generation (UIG), including LSUN-bed (Yu et al. 2015), CelebA-HQ (Lee et al. 2020), and Lanscape (Skorokhodov, Sotnikov, and Elhoseiny 2021). Due to the page limit, please refer to the supplementary for more results.

Conditional Complex Scene Generation

Text Conditional Image Generation We first experiment on the standard text-to-image (T2I) generation for COCO, and the results are shown in Table 1. We consider standard setting of training on COCO train2014 split. Orthogonal to recent T2I models pre-trained on huge image-text pairs, our goal is to synthesize images from diverse conditions. In this setting, FID measures the image quality and CLIP-Score assesses the image-text consistency. For completeness, IS is also reported, though FID is known for a stronger correlation with human judgment than IS (Zhang et al. 2021; Sylvain et al. 2021). Besides standard diffusion inference, we also report the variant with classifier-free guidance (Nichol et al. 2022). As shown in Table 1, for both inference types, *Frido* significantly outperforms the previous best model LDM by $\approx$ 2 points for FID and $\approx$ 1 point for CLIP-Score, achieving state-of-the-art scores on FID (15.38 vs. 11.24) and CLIP-Score (0.6607 vs. 0.7046). In a different setting, LAFITE (Zhou et al. 2022) incorporated pre-trained CLIP (Radford et al. 2021), which contained abundant text-image knowledge from web-scale data pairs. As an initial step for incorporating CLIP knowledge with *Frido*, we report the results with a test-time only CLIP ranking trick (Ding et al. 2021) (10 inferences). We can see that CLIPr further improves all metrics significantly, achieving comparable FID and CLIP-Score to LAFITE. An orthogonal direction to utilize CLIP at training similar to LAFITE is left to future works.

Methods	COCO			Visual Genome		
	FID↓	IS↑	CLIP↑	FID↓	IS↑	CLIP↑
GT	-	-	0.766	-	-	0.662
Sg2Im	127.0	6.179	-	-	-	-
WSGC	119.1	7.235	-	45.7	10.69	-
LDMs-8†	49.14	13.33	0.627	36.88	14.60	0.611
Frido-f16f8	**46.11**	**13.41**	**0.642**	**31.61**	**15.07**	**0.613**

Table 2: Scene-graph-to-image generation on COCO and Visual Genome. †: reproduced with official code and configs. Note that both LDM and *Frido* are inferenced with classifier-free guidance.

Image Generation from Scene Graph To further verify the claimed semantic relation capturing, we run SG2I on COCO-stuff and VG datasets, and the results are shown in Table 2. Clearly, *Frido* outperforms all previous methods, including sg2im (Johnson, Gupta, and Li 2018), WSGC (Herzig et al. 2020), and LDMs, in terms of FID and IS, achieving new state-of-the-art. Moreover, to quantitatively measure the semantic correctness of the image w.r.t. its SG condition, we transform the SG to captions by concatenating the relation triplets (i.e., subject-predicate-object) and report the CLIP-score of the resulting image-caption pairs. Our model surpasses previous work by $\approx 2\%$ on COCO and $\approx 0.2\%$ on VG. This empirically verifies that, with the feature pyramid and coarse-to-fine generation strategy, *Frido* improves modeling of complex relations.

Label-to-Image Generation Label-to-image produces scene image conditioned on image-level labels. Unlike T2I or SG2I, where scene structure is specified by the text conditions, this task requires a model to combine objects more freely and synthesize a coherent image. In addition to FID and IS, precision and recall are reported for object-level quality and diversity measurement, respectively. We conduct experiments on Label2I with COCO-stuff. As the shown in Table 3, our model outperforms all previous approaches, in-

[3] https://arxiv.org/abs/2208.13753

Name	FID	IS	Precision	Recall
3-8 labels in the image				
LayoutVAE	60.7	-	-	-
+LostGAN	74.06	11.66	0.231	0.473
LDMs-8†	51.45	**15.05**	0.434	0.576
Frido-f16f8	**47.39**	14.73	**0.437**	**0.595**
2-30 labels in the image				
LDMs-8†	29.17	**18.00**	0.563	**0.554**
Frido-f16f8	**27.65**	17.70	**0.573**	0.542

Table 3: Label-to-image generation on COCO. †: reproduced with official code and configs.

Methods	COCO 256			OpenImage 256	
	FID↓	YOLO↑	sFID↓	FID↓	sFID↓
LostGAN-V2	42.55	-	-	-	-
OC-GAN	41.65	-	-	-	-
SPADE	41.11	-	-	-	-
VQGAN+T	56.58	-	24.07	45.33	15.85
LDM-8 (100 steps)	42.06	-	-	-	-
LDM-8† (100 steps)	41.02	14.67	21.63	-	-
LDM-4 (200 steps)	40.91	-	-	32.02	-
Frido-f16f8 (100 steps)	38.95	16.71	17.69	-	-
Frido-f8f4 (200 steps)	**37.14**	**17.22**	**14.91**	**29.04**	**12.77**

Table 4: Layout-to-image generation on COCO (segmentation challenge split) and OpenImages. †: reproduced with official code and configs. sFID denotes scene FID.

cluding LayoutVAE (Jyothi et al. 2019)[4] and LDMs, on not only FID but also precision and recall under the more common 3-8 labels setting. This indicates that, *Frido* achieves better image quality and data manifold modeling of multi-object images. We further challenge *Frido* with a harder 2-30 labels setting and still establish SOTA FID.

Layout-to-Image Generation Our Layout2I results show-cases that multiple objects' shapes and details can be synthesized. Specifically, we compare our *Frido* with previous methods under two different settings. Firstly, we follow LDM and perform experiments on COCO stuff segmentation challenge split and OpenImage datasets. The results are shown in Table 4. One can find that *Frido* outperforms previous methods, including LostGAN-v2 (Sun and Wu 2019), OC-GAN (Sylvain et al. 2021), SPADE (Park et al. 2019), VQGAN+T (combining Esser, Rombach, and Ommer 2021 and Brown et al. 2020), and LDM, on FID by at least 2 points, achieving new state-of-the-art for both COCO and OpenImages. Moreover, we achieve the best YOLO scores and sceneFID, indicating the most visually realistic instance-level objects. Secondly, we follow TwFA (Yang et al. 2022) and conduct experiments on standard COCO stuff and Visual Genome datasets. Please refer to the supplementary material for more detail.

[4] LayoutVAE implements Label2I as Label2Layout + Layout2I and reported 128 resolution result. We follow Yang et al. (2021) to adopt LostGAN (Sun and Wu 2019) to achieve 256 resolution.

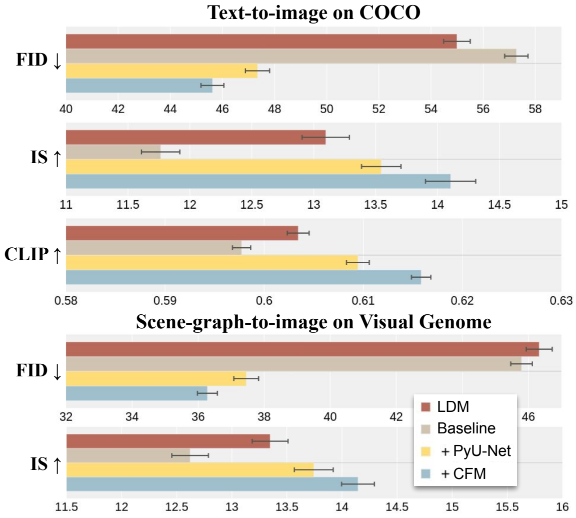


Figure 4: Model ablation on COCO T2I and VG SG2I. CFM denotes our coarse-to-fine modulation.

Model Analysis

Model Ablation To verify the key novel designs of *Frido*, we perform ablation studies on two tasks: text-to-image (T2I) on COCO and scene-graph-to-image (SG2I) on Visual Genome. Figure 4 showcases the contribution of each deployed component in *Frido*. For ablations and hyper-parameter tunings, we train for 250K iterations to allow more experiments. Models with the best dev scores are further trained to obtain the final test scores in Sec. . We report mean and the corresponding 95% confidence interval by conducting the bootstrap test (Koehn 2004) and the sample size equals to test set size; resampled for 100 times. For the baseline, we use two LDMs and perform a simple sequential learning strategy. More specifically, the first LDM learns the distribution of the high-level feature map (LDM-16); and the second LDM is deployed to model the low-level feature of f8 (LDM-8). In this baseline model, we concat LDM-8's output feature map and the denoising target feature feed into the LDM-8 for denoising. To justify the shared U-Net design of PyU-Net, we first apply this module without CFM to the baseline. Sharing U-Net reduces the model parameters from 1.18B (baseline) to 590M (baseline + PyU-Net). Finally, the coarse-to-fine modulation is added, with only a minor increase in parameter count (total of 697M), and performance is further boosted for all metrics. We can see that each component significantly improves the generation results; models with PyU-Net and CFM are significantly better than the LDMs on all metrics.

Computation Cost Analysis Here we analyze the inference cost of our model. In Fig. 5, we compare *Frido* with LDM on the speed-quality trade-off. In this figure, we inference each model with different inference timesteps T and then plot the FID scores and against the per-image inference cost. Note that the experiments are done on validation splits with batch size of 32 using 1 V100. It is demonstrated that under similar inference budget, *Frido* achieve decent per-

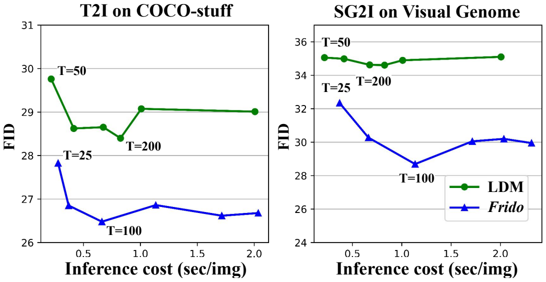


Figure 5: Speed-quality trade-off analysis. Lower FID indicates better image quality.

formance gain comparing to LDM, confirming the claimed efficiency of our model. For other comparisons on FLOPs, parameter counts, and inference time, please see the supplementary. Note that by operating in the latent space, LDM is among the faster ones within the DM model class. *Frido* further reduce the cost by putting part of the denoising load at lower-resolution.

Takeaways The empirical studies have shown that *Frido* significantly outperforms the baseline LDM for complex scene image synthesis, and even achieves SOTA in 5 settings. Our modeling novelty, including PyU-Net and the coarse-to-fine modulation, are statistically effective. Last but not least, *Frido* is more efficient, as can be seen in a head-to-head comparison to LDM, which mitigates the notorious heavy inference cost for diffusion models.

Related Work

More Generative Models for Image Synthesis The community has witnessed great progress of image synthesis in the past decade. Other than the previously discussed works, the families of GANs (Liao et al. 2022; Xu et al. 2018; Brock, Donahue, and Simonyan 2019; Gafni et al. 2022; Zhang et al. 2021; Hinz, Heinrich, and Wermter 2020; Karras et al. 2021), VAEs (Sohn, Lee, and Yan 2015), autoregressive models (Razavi, Van den Oord, and Vinyals 2019; Chang et al. 2022; Yu et al. 2022), flow-based methods (Dinh, Sohl-Dickstein, and Bengio 2017), and diffusion-based models (Saharia et al. 2022; Gu et al. 2022) have all made great contribution to shape this field. *Frido* is a hybrid of the VAE and DM family, combining the best of both worlds for outstanding image quality on complex scenes, and significantly improved DM inference. Very recently, large-scale pre-training for text-to-image generation (Ramesh et al. 2022) has gained vast attention and achieved superior results. *Frido* is orthogonal to these models, as we investigate coarse-to-fine synthesis and multimodality inputs beyond text.

Two-Stage Generative Models Recently, many two-stage generative models (Van Oord, Kalchbrenner, and Kavukcuoglu 2016; Jahn, Rombach, and Ommer 2021; Pandey et al. 2022; Zhou et al. 2022) are proposed to tackle the drawbacks of the one-stage models. The representative VQ-VAE (Van Den Oord, Vinyals et al. 2017) first encodes an image into a discrete latent space with a lower spatial resolution and then uses an auto-regressive network to model such space. The first step is called Vector Quantization (VQ), which reduces the input information to allow auto-encoder learning. In addition, VQ bridges images to other modalities, such as language (Ding et al. 2021; Chen et al. 2020) and audio (Yan et al. 2021), seamlessly by converting to discrete tokens. In the second stage, an auto-regressive(e.g. PixelCNN (Van den Oord et al. 2016), VQGAN) or diffusion model (LDM, VQ-Diffusion (Tang et al. 2022)) is adopted to model the encoded latent space. *Frido* contributes to both stages by proposing MS-VQGAN and PyU-Net for DM.

Coarse-to-Fine Image Generation Approaches Instead of generating a full-resolution image in one step, coarse-to-fine generation synthesizes an image with multiple steps, from low to high resolution in pixel space (Gregor et al. 2015; Mansimov et al. 2016; Ho et al. 2022) or from high-level to low-level information in latent space (Razavi, Van den Oord, and Vinyals 2019; Child et al. 2019). These allow model to better capture the information in different levels and have shown to achieve higher quality. For instance, AttnGAN (Xu et al. 2018) and StackGAN (Zhang et al. 2017, 2018) first produce an image in low resolution (e.g., 1/8 of the full-size) and then iteratively scale up the generated image until achieving the final resolution. Different from the above works, we share the core network for each scale. Therefore, the overhead compared to single-scale models is minimized.

Conclusion

We propose *Frido*, a new image generative model, empowering an under-explored coarse-to-fine prior in the diffusion model family. Key designs such as multi-scale codebooks, a single shared U-Net, and the special modulation mechanism are shown to be effective via extensive experiments. Empirically, we apply this model to a diverse set of cross-modal image synthesis tasks and achieve 5 new state-of-the-art results. From a practical aspect, *Frido* also mitigates the well-known slow inference pain-point of diffusion methods.

References

Brock, A.; Donahue, J.; and Simonyan, K. 2019. Large scale GAN training for high fidelity natural image synthesis. In *ICLR*.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. In *NeurIPS*.

Chang, H.; Zhang, H.; Jiang, L.; Liu, C.; and Freeman, W. T. 2022. Maskgit: Masked generative image transformer. In *CVPR*.

Chen, M.; Radford, A.; Child, R.; Wu, J.; Jun, H.; Luan, D.; and Sutskever, I. 2020. Generative pretraining from pixels. In *ICML*.

Child, R.; Gray, S.; Radford, A.; and Sutskever, I. 2019. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*.

Dhariwal, P.; and Nichol, A. 2021. Diffusion models beat gans on image synthesis. In *NeurIPS*.

Ding, M.; Yang, Z.; Hong, W.; Zheng, W.; Zhou, C.; Yin, D.; Lin, J.; Zou, X.; Shao, Z.; Yang, H.; et al. 2021. Cogview: Mastering text-to-image generation via transformers. In *NeurIPS*.

Dinh, L.; Krueger, D.; and Bengio, Y. 2014. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516*.

Dinh, L.; Sohl-Dickstein, J.; and Bengio, S. 2017. Density estimation using real nvp. In *ICLR*.

Elfwing, S.; Uchibe, E.; and Doya, K. 2018. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. *Neural Networks*.

Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *CVPR*.

Gafni, O.; Polyak, A.; Ashual, O.; Sheynin, S.; Parikh, D.; and Taigman, Y. 2022. Make-a-scene: Scene-based text-to-image generation with human priors. In *ECCV*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial nets. In *NeurIPS*.

Gregor, K.; Danihelka, I.; Graves, A.; Rezende, D.; and Wierstra, D. 2015. Draw: A recurrent neural network for image generation. In *ICML*.

Gu, S.; Chen, D.; Bao, J.; Wen, F.; Zhang, B.; Chen, D.; Yuan, L.; and Guo, B. 2022. Vector quantized diffusion model for text-to-image synthesis. In *CVPR*.

Herzig, R.; Bar, A.; Xu, H.; Chechik, G.; Darrell, T.; and Globerson, A. 2020. Learning canonical representations for scene graph to image generation. In *ECCV*.

Hessel, J.; Holtzman, A.; Forbes, M.; Bras, R. L.; and Choi, Y. 2021. Clipscore: A reference-free evaluation metric for image captioning. In *EMNLP*.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*.

Hinz, T.; Heinrich, S.; and Wermter, S. 2020. Semantic object accuracy for generative text-to-image synthesis. *TPAMI*.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. In *NeurIPS*.

Ho, J.; Saharia, C.; Chan, W.; Fleet, D. J.; Norouzi, M.; and Salimans, T. 2022. Cascaded Diffusion Models for High Fidelity Image Generation. *JMLR*.

Hore, A.; and Ziou, D. 2010. Image quality metrics: PSNR vs. SSIM. In *ICPR*.

Huang, X.; and Belongie, S. 2017. Arbitrary style transfer in real-time with adaptive instance normalization. In *ICCV*.

Jahn, M.; Rombach, R.; and Ommer, B. 2021. High-Resolution Complex Scene Synthesis with Transformers. *arXiv preprint arXiv:2105.06458*.

Johnson, J.; Gupta, A.; and Li, F.-F. 2018. Image generation from scene graphs. In *CVPR*.

Jyothi, A. A.; Durand, T.; He, J.; Sigal, L.; and Mori, G. 2019. Layoutvae: Stochastic scene layout generation from a label set. In *ICCV*.

Karras, T.; Aittala, M.; Laine, S.; Härkönen, E.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2021. Alias-free generative adversarial networks. In *NeurIPS*.

Kingma, D. P.; and Dhariwal, P. 2018. Glow: Generative flow with invertible 1x1 convolutions. In *NeurIPS*.

Kingma, D. P.; and Welling, M. 2014. Auto-encoding variational bayes. In *ICLR*.

Koehn, P. 2004. Statistical significance tests for machine translation evaluation. In *EMNLP*.

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*.

Kuznetsova, A.; Rom, H.; Alldrin, N.; Uijlings, J.; Krasin, I.; Pont-Tuset, J.; Kamali, S.; Popov, S.; Malloci, M.; Kolesnikov, A.; et al. 2020. The open images dataset v4. *IJCV*.

Lee, C.-H.; Liu, Z.; Wu, L.; and Luo, P. 2020. MaskGAN: Towards Diverse and Interactive Facial Image Manipulation. In *CVPR*.

Li, W.; Zhang, P.; Zhang, L.; Huang, Q.; He, X.; Lyu, S.; and Gao, J. 2019. Object-driven text-to-image synthesis via adversarial training. In *CVPR*.

Li, Z.; Wu, J.; Koh, I.; Tang, Y.; and Sun, L. 2021. Image Synthesis from Layout with Locality-Aware Mask Adaption. In *ICCV*.

Liao, W.; Hu, K.; Yang, M. Y.; and Rosenhahn, B. 2022. Text to image generation with semantic-spatial aware GAN. In *CVPR*.

Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; and Zitnick, C. L. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Mansimov, E.; Parisotto, E.; Ba, J. L.; and Salakhutdinov, R. 2016. Generating images from captions with attention. In *ICLR*.

Nichol, A.; Dhariwal, P.; Ramesh, A.; Shyam, P.; Mishkin, P.; McGrew, B.; Sutskever, I.; and Chen, M. 2022. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*.

Nichol, A. Q.; and Dhariwal, P. 2021. Improved denoising diffusion probabilistic models. In *ICML*.

Pandey, K.; Mukherjee, A.; Rai, P.; and Kumar, A. 2022. DiffuseVAE: Efficient, Controllable and High-Fidelity Generation from Low-Dimensional Latents. *TMLR*.

Park, T.; Liu, M.-Y.; Wang, T.-C.; and Zhu, J.-Y. 2019. Semantic image synthesis with spatially-adaptive normalization. In *CVPR*.

Radford, A.; Kim, J. W.; Hallacy, C.; Ramesh, A.; Goh, G.; Agarwal, S.; Sastry, G.; Askell, A.; Mishkin, P.; Clark, J.; et al. 2021. Learning transferable visual models from natural language supervision. In *ICML*.

Radford, A.; Metz, L.; and Chintala, S. 2015. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

Ramesh, A.; Dhariwal, P.; Nichol, A.; Chu, C.; and Chen, M. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.

Razavi, A.; Van den Oord, A.; and Vinyals, O. 2019. Generating diverse high-fidelity images with vq-vae-2. In *NeurIPS*.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *CVPR*.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*.

Saharia, C.; Chan, W.; Saxena, S.; Li, L.; Whang, J.; Denton, E.; Ghasemipour, S. K. S.; Ayan, B. K.; Mahdavi, S. S.; Lopes, R. G.; et al. 2022. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. In *NeurIPS*.

Sajjadi, M. S.; Bachem, O.; Lucic, M.; Bousquet, O.; and Gelly, S. 2018. Assessing generative models via precision and recall. In *NeurIPS*.

Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; and Chen, X. 2016. Improved techniques for training gans. In *NeurIPS*.

Skorokhodov, I.; Sotnikov, G.; and Elhoseiny, M. 2021. Aligning latent and image spaces to connect the unconnectable. In *ICCV*.

Sohl-Dickstein, J.; Weiss, E.; Maheswaranathan, N.; and Ganguli, S. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *ICML*.

Sohn, K.; Lee, H.; and Yan, X. 2015. Learning structured output representation using deep conditional generative models. In *NeurIPS*.

Sun, W.; and Wu, T. 2019. Image synthesis from reconfigurable layout and style. In *ICCV*.

Sylvain, T.; Zhang, P.; Bengio, Y.; Hjelm, R. D.; and Sharma, S. 2021. Object-centric image generation from layouts. In *AAAI*.

Tang, Z.; Gu, S.; Bao, J.; Chen, D.; and Wen, F. 2022. Improved Vector Quantized Diffusion Models. *arXiv preprint arXiv:2205.16007*.

Tao, M.; Tang, H.; Wu, F.; Jing, X.-Y.; Bao, B.-K.; and Xu, C. 2022. DF-GAN: A Simple and Effective Baseline for Text-to-Image Synthesis. In *CVPR*.

Van den Oord, A.; Kalchbrenner, N.; Espeholt, L.; Vinyals, O.; Graves, A.; et al. 2016. Conditional image generation with pixelcnn decoders. In *NeurIPS*.

Van Den Oord, A.; Vinyals, O.; et al. 2017. Neural discrete representation learning. In *NeurIPS*.

Van Oord, A.; Kalchbrenner, N.; and Kavukcuoglu, K. 2016. Pixel recurrent neural networks. In *ICML*.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *TIP*.

Xu, T.; Zhang, P.; Huang, Q.; Zhang, H.; Gan, Z.; Huang, X.; and He, X. 2018. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*.

Yan, W.; Zhang, Y.; Abbeel, P.; and Srinivas, A. 2021. Videogpt: Video generation using vq-vae and transformers. *arXiv preprint arXiv:2104.10157*.

Yang, C.-F.; Fan, W.-C.; Yang, F.-E.; and Wang, Y.-C. F. 2021. Layouttransformer: Scene layout generation with conceptual and spatial diversity. In *CVPR*.

Yang, Z.; Liu, D.; Wang, C.; Yang, J.; and Tao, D. 2022. Modeling Image Composition for Complex Scene Generation. In *CVPR*.

Yu, F.; Seff, A.; Zhang, Y.; Song, S.; Funkhouser, T.; and Xiao, J. 2015. Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*.

Yu, J.; Li, X.; Koh, J. Y.; Zhang, H.; Pang, R.; Qin, J.; Ku, A.; Xu, Y.; Baldridge, J.; and Wu, Y. 2022. Vector-quantized image modeling with improved vqgan. In *ICML*.

Zhang, H.; Koh, J. Y.; Baldridge, J.; Lee, H.; and Yang, Y. 2021. Cross-modal contrastive learning for text-to-image generation. In *CVPR*.

Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; and Metaxas, D. N. 2017. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*.

Zhang, H.; Xu, T.; Li, H.; Zhang, S.; Wang, X.; Huang, X.; and Metaxas, D. N. 2018. Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *TPAMI*.

Zhou, Y.; Zhang, R.; Chen, C.; Li, C.; Tensmeyer, C.; Yu, T.; Gu, J.; Xu, J.; and Sun, T. 2022. LAFITE: Towards Language-Free Training for Text-to-Image Generation. In *CVPR*.

Zhu, M.; Pan, P.; Chen, W.; and Yang, Y. 2019. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis. In *CVPR*.