KOALA: Self-Attention Matters in Knowledge Distillation of Latent Diffusion Models for Memory-Efficient and Fast Image Synthesis

Youngwan Lee[1,2] Kwanyong Park[1] Yoorhim Cho[3] Yong-Ju Lee[1] Sung Ju Hwang[2]

[1]Electronics and Telecommunications Research Institute (ETRI), South Korea
[2]Korea Advanced Institute of Science and Technology (KAIST), South Korea
[3]Sookmyung Women's University, South Korea

project page: https://youngwanlee.github.io/KOALA/

Abstract

Stable diffusion is the mainstay of the text-to-image (T2I) synthesis in the community due to its generation performance and open-source nature. Recently, Stable Diffusion XL (SDXL), the successor of stable diffusion, has received a lot of attention due to its significant performance improvements with a higher resolution of 1024×1024 *and a larger model. However, its increased computation cost and model size require higher-end hardware (*e.g.*, bigger VRAM GPU) for end-users, incurring higher costs of operation. To address this problem, in this work, we propose an efficient latent diffusion model for text-to-image synthesis obtained by distilling the knowledge of SDXL. To this end, we first perform an in-depth analysis of the denoising U-Net in SDXL, which is the main bottleneck of the model, and then design a more efficient U-Net based on the analysis. Secondly, we explore how to effectively distill the generation capability of SDXL into an efficient U-Net and eventually identify four essential factors, the core of which is that self-attention is the most important part. With our efficient U-Net and self-attention-based knowledge distillation strategy, we build our efficient T2I models, called KOALA-1B &-700M, while reducing the model size up to 54% and 69% of the original SDXL model. In particular, the KOALA-700M is more than twice as fast as SDXL while still retaining a decent generation quality. We hope that due to its balanced speed-performance tradeoff, our KOALA models can serve as a cost-effective alternative to SDXL in resource-constrained environments.*

1. Introduction

Since the emergence of the Stable diffusion models (SDMs) [41, 47, 48] which are based on the latent diffusion model [46], not only has text-to-image synthesis greatly advanced but also applications utilizing it have been actively developed, such as image editing [8, 63], controllable image synthesis [38, 71] personalized image synthesis [12, 30, 52], text-to-video generation [3, 25] and 3D asset synthesis [32, 42, 70]. While these downstream tasks benefit from SDM's superior image generation quality as a backbone, its massive computation costs and large model size require expensive hardware equipment and thus incur huge costs. Furthermore, a more recent version of the stable diffusion model, SDXL [41], demonstrates significantly improved image generation quality with a higher resolution of 1024×1024, but at the cost of more computations and memory requirement.

To alleviate this computation burden, several works have been proposed, which introduce quantization [59], hardware-aware optimization [7, 9], denoising step reduction [31, 37, 54], and architectural model optimization [26, 31]. In particular, the denoising step reduction [31, 37, 54] and architectural model compression [26] methods adopt the knowledge distillation (KD) scheme [15, 18] by allowing the model to mimic the output of the SDM as a teacher model. The step-distillation methods [31, 37, 54] allow the denoised latent of the diffusion model in the early denoising steps to mimic the output in the later denoising steps of the teacher model. As an orthogonal work for the architectural model compression, BK-SDM [26] exploits KD when compressing the most heavy-weight part, U-Net, in SDM-v1.4 [47]. BK-SDM builds a compressed U-Net by simply removing some blocks and allows the compressed U-Net to mimic the last features at each stage and the predicted noise of the teacher model during the pre-training phase. However, the compression method proposed by BK-SDM achieves a limited compression rate (33%) when applied to the larger SDXL than SDM-v1.4 and the strategy for feature

Figure 1. **Generated samples by our KOALA-700M trained by the proposed knowledge-distillation approach with SDXL [41]**. With the following settings: FP-16 precision, 1024 $\times$ 1024 resolution, and 25 denoising steps with Euler discrete scheduler [24] same as the huggingface's SDXL-Base-1.0 model [66], the inference time is 1.4 seconds on an NVIDIA 4090 (24GB) GPU, which is over 2$\times$ faster than SDXL-Base-1.0 (3.3s) while reducing the U-Net model size by 69%.

distillation in U-Net has *not yet been fully explored*.

In this work, our goal is to build a more efficient text-to-image synthesis model by distilling the generation capability of SDXL [41]. To this end, we first perform an in-depth analysis of SDXL's denoising U-Net, which requires the most number of parameters and computational cost, and find that most of the parameters are concentrated at the lowest feature level due to the large number of transformer blocks. Based on the analysis, we design an efficient U-Net by reducing the origin SDXL's U-Net by up to 69% (vs. BK's method: 33%). Furthermore, we investigate how to effectively distill SDXL as a teacher model and find four essential factors for feature-level knowledge distillation. The core of these findings is that self-attention features are the most crucial for distillation due to the fact that self-attention-based KD allows models to learn more discriminative representations between objects or attributes.

With our knowledge distillation (KD) strategies, we train an efficient text-to-image synthesis model on top of SDXL [41], called KOALA, by only replacing SDXL's U-Net with our efficient U-Net. KOALA is trained on a smaller *publicly available* LAION-Aesthestics-V2-6+ [57], which has only 8M text-image pairs. Recent studies [2, 28, 67, 68] have shown that FID [17] is not well correlated with the fidelity of the generated image, and thus we use two alternative evaluation metrics: Human Preference Score (HPSv2) [67] for visual aesthetics and T2I-Compbench [21] for image-text alignment. Our efficient KOALA models consistently outperform BK-SDM [26]'s KD method in both metrics. Furthermore, our smaller

model, KOALA-700M, shows better performance than SDM-v2.0 [48], which is one of the most widely used in the community, while having a similar model size and inference speed. Lastly, to validate its practical impact, we perform inference analysis on a variety of *consumer-grade* GPUs with different memory sizes (*e.g.*, 8GB, 11GB, and 24GB), and the results show that whereas SDXL cannot be mounted on an 8GB GPU, our KOALA-700M can run on it while still retaining decent image generation quality as shown in Fig. 1. Our main contributions are as follows:

1. We design two efficient denoising U-Net architectures with model sizes (1.13B/782M) more than twice as small as SDXL's U-Net (2.56B).
2. We perform a comprehensive analysis of the knowledge distillation strategies for SDXL, finding four essential factors for feature distillation.
3. We build two efficient T2I models pre-trained by the proposed KD, called KOALA-1B/700M, which is more than 2× smaller and faster compared to SDXL-Base.
4. We perform a systemical analysis of inference on a variety of GPUs, showing that our KOALA-700M can operate on an economical GPU with 8 GB of memory.

2. Related Works

Knowledge distillation for efficient T2I diffusion models. Denoising diffusion models [20, 61] dominate the recent state-of-the-art text-to-image (T2I) diffusion models [10, 44, 46, 53] due to their unprecedented high quality and diversity. However, a significant shortcoming of these models is their intensive computational demands during sampling time, which constrains their utility in practical scenarios. To alleviate this problem, early efforts [37, 54] have focused on improving the sampling speed by reducing the number of required sampling steps. In particular, Salimans et al. [54] proposes a concept of *step distillation*, which trains a student model with fewer steps, distilled from a pre-trained diffusion model as a teacher model. Meng et al. [37] expand this concept to classifier-free guided diffusion models, facilitating the step distillation for modern text-to-image diffusion models. Although these methods significantly speed up the inference of models, the hardware prerequisites still pose challenges to practitioners.

As another line of research, BK-SDM [26] attempts an architectural compression of diffusion models. They first eliminate redundant network components to construct a shallow model. Then, a simple knowledge distillation method [15, 50] is employed to transfer the knowledge from the original pre-trained diffusion model (*e.g.*, SDM-v1.4 [47]). Remarkably, this compressed model has achieved substantial reductions in sampling time, GPU memory demands, and storage requirements, with only a modest degradation in performance. However, BK-SDM's simple block removal method has limitations in compressing more complex and larger U-Net models such as SDXL [41]. Beyond the BK-SDM method, which only distills the last feature at each stage, there is still room for further exploration in distilling knowledge from more complex U-Net in SDXL.

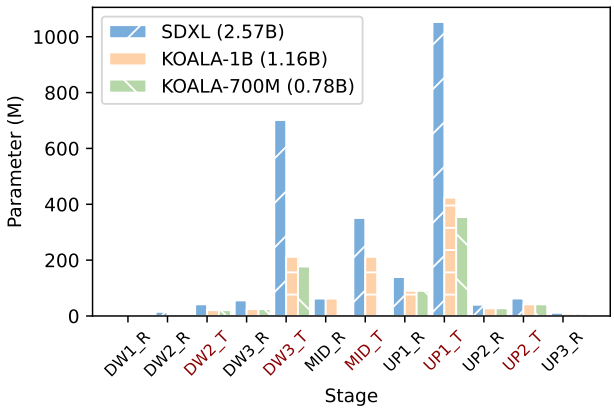

Figure 2. **Dissection of U-Net in SDXL.** DWi and UPi indicate i-*th* stage of the down and the up block, and R and T denote the Residual block and Transformer block, respectively.

SDXL-Base	Text Encoder [22, 43]	VAE Decoder [4]	U-Net
#Parameters	817M	83M	2,567M
Latency (s)	0.008	0.002	3.133

Table 1. **SDXL-Base-1.0 model budget.** Latency is measured under the image scale of 1024 × 1024, FP16-precision, and 25 denoising steps in NVIDIA 4090 GPU (24GB).

U-Net	SDM-v2.0	SDXL-Base	**KOALA-1B**	**KOALA-700M**
Param.	865M	2,567M	1,161M	782M
CKPT size	3.46GB	10.3GB	4.4GB	3.0GB
Tx blocks	[1, 1, 1, 1]	[0, 2, 10]	[0, 2, 6]	[0, 2, 5]
Mid block	✓	✓	✓	✗
Latency	1.131s	3.133s	1.604s	1.257s

Table 2. **U-Net Comparison.** Tx means Transformer. SDM-v2.0 [48] uses 768 × 768 resolution, while SDXL and KOALA models use 1024 × 1024 resolution. Latency is measured with FP16-precision, and 25 denoising steps in NVIDIA 4090 GPU (24GB). CKPT means the trained checkpoint file.

3. Analysis: Stable Diffusion XL

SDXL [41], the latest version of the SDM series [46–48], exerts a significant influence on both the academic community and the industry due to its unprecedented quality and open source resources. It has several key improvement points from the previous SDM-v2.0 [48], *e.g.*, multiple sizes- & crop-conditioning, improved VAE [27, 45], and much larger U-Net [51], and an ad hoc style of refinement module, which leads to significantly improving generation

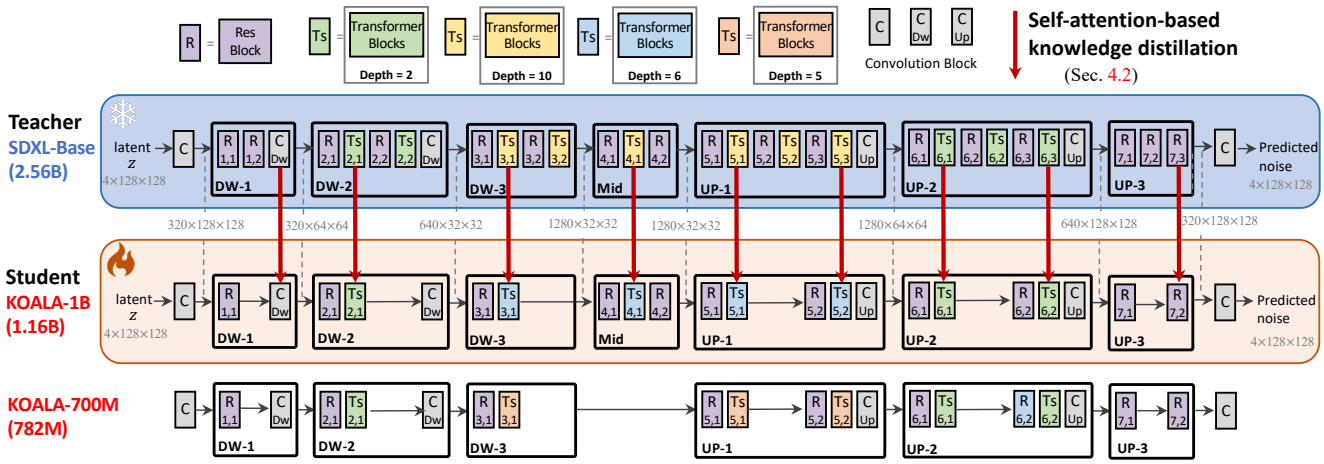

Figure 3. **Overview of KnOwledge-DistillAtion in LAtent diffusion model based on SDXL and architecture of KOALA.** We omit skip connections for simplicity. We perform feature distillation in transformer blocks using the output of the self-attention layer.

quality. However, the significant enlargement of U-Net in model size results in increased computational costs and significant memory (or storage) requirements, hampering the accessibility of SDXL. Thus, we investigate the U-Net in SDXL regarding model size and latency to design a more lightweight U-Net for knowledge distillation. We dissect the components of SDXL, quantifying its size and latency during the denoising phase, as detailed in Tab. 1. The enlarged U-Net (2.56B) is the primary cause of the increasing SDXL model size (vs. SDM-v2.0 (865 M)). Furthermore, the latency of U-Net is the main inference time bottleneck in SDXL. Therefore, it is necessary to reduce U-Net's model budget for better efficiency.

The SDXL's U-Net architecture varies in the number of transformer blocks for each stage, unlike SDM-v2.0, which employs a transformer block for each stage (see Tab. 2). At the highest feature levels (*e.g.*, `DW-1&UP-3` in Fig. 3), SDXL uses only residual blocks [14] without transformer blocks, instead distributing more transformer blocks to lower-level features. So, in Fig. 2, we analyze the parameter distribution of each stage in the U-Net. Most parameters are concentrated on the transformers with ten blocks in the lowest feature map (*e.g.*, 32 $\times$ 32 of `DW-3`, `Mid`, `UP-1` in Fig. 3), making the main parameter bottleneck. Thus, it is essential to address this bottleneck when designing an efficient U-Net.

4. Approach

In this section, we first propose a simple yet efficient U-Net architecture in Sec. 4.1. Then, we explore how to effectively distill the knowledge from U-Net in SDXL [41] into the proposed efficient U-Net in Sec. 4.2.

4.1. Efficient U-Net architecture

Based on the investigation of the U-Net model budget of SDXL in Sec. 3, we propose a simple yet efficient U-Net architecture. BK-SDM [26] also aimed to compress the U-Net of SDM-v1.4 [47] by removing a pair of a residual block and a transformer block at each stage. However, the BK-SDM's approach is only suitable for SDM-v1.4, which has only one transformer block (*i.e.*, depth=1) at each stage. As SDXL's U-Net has a different number of transformer blocks at each stage, as shown in Tab. 2, simple block-level removal (one block pair) can only reduce SDXL's U-Net to at most 1.3B model parameters.

In this work, we devise a compressed U-Net that is more suitable for SDXL [41] compared to that of BK-SDM [26]. Similar to BK-SDM, we first remove the residual-transformer blocks pair at each stage. Specifically, in the encoder part (`DW-i`), each stage has two alternating pairs of a residual block and transformer blocks. We remove the last pair of residual-transformer blocks at each stage. In the decoder part (`UP-i`), we remove the intermediate pair of residual-transformer blocks. Furthermore, focusing on the fact that the majority of the parameters are concentrated on the transformer blocks at the lowest features (Fig. 2), we reduce the depth of the transformer blocks from 10 to 5 or 6 at the lowest features (*i.e.*, `DW-3`, `Mid` and `UP-1` in Fig. 3). As a result, we design two types of compressed U-Net, KOALA-1B and KOALA-700M. More details of the proposed U-Nets are demonstrated in Tab. 2 and Fig. 3. Note that we remove `Mid` block in KOALA-700M for additional model compression. Our KOALA-1B model has 1.16B parameters, making it twice as compact as SDXL (2.56B). Meanwhile, KOALA-700M, with its 782M parameters, is comparable in size to SDM-v2.0 (865M). It is noteworthy that KOALA-700M achieves almost twice the

Distill type	HPSv2
SD-loss	25.53
SA	**26.74**
CA	26.11
Res	26.27
FFN	26.48
LF (BK [26])	26.63

(a) **Distillation type.**

Distill loc.	HPSv2
SD-loss	25.53
DW-2	25.32
DW-3	25.57
Mid	25.66
UP-1	**26.52**
UP-2	26.05

(b) **Distill stage.** We distill the SA feature at only each stage.

SA loc.	HPSv2
SA-bottom	**26.74**
SA-interleave	26.58
SA-up	26.48

(c) **SA locations** to distill from a transformer block with a depth of 10 in Teacher U-Net in SDXL.

Combination	HPSv2
Baseline (SA only)	26.74
SA + LF at DW-1 & UP-3	**26.98**
SA + Res at DW-1 & UP-3	26.94
SA + LF all	26.83
SA + Res all	26.80
SA+CA+Res+FFN+LF all	26.39

(d) **Combination.** DW-1 & UP-3 are the highest feature resolution in U-Net.

Table 3. **Analysis of feature level knowledge distillation of U-Net in SDXL [41].** SA, CA, and FFN denote self-attention, cross-attention, and feed-forward net in the transformer block. Res is a convolutional residual block and LF denotes the last feature (same in BK [26]). For the ablation study, we train our KOALA-1B as student U-Net for 30K iterations with a batch size of 32.

faster inference speed over SDXL, as well as comparable to SDM-v2.0, which generates lower-resolution images.

4.2. Exploring Knowledge distillation for SDXL

Now we explore how to effectively distill the knowledge of U-Net in SDXL [41] into the proposed compact U-Net described in Sec. 4.1. As a latent diffusion model [46], SDXL encodes input $x \in \mathbb{R}^{3\times1024\times1024}$ to latent representation $z \in \mathbb{R}^{4\times128\times128}$ via VAE [27, 45] to reduce computation cost for high-resolution generation. The latent representation z is then fed into the U-Net [51] to predict the noise (ϵ), which is the most essential part of generating the image and also the most computationally intensive. We replace this U-Net in SDXL with the proposed efficient U-Net and then train the replaced U-Net through knowledge-distillation-based learning.

Kim *et al*. [26] adopted knowledge distillation for text-to-image pre-training by using the U-Net in SDM-v1.4 [47] as a teacher model (T_θ) through the following objectives $\mathcal{L}_{\text{task}} + \mathcal{L}_{\text{outKD}} + \mathcal{L}_{\text{featKD}}$:

$$\mathcal{L}_{\text{task}} = \min_{S_\theta} \mathbb{E}_{z_t,\epsilon,t,c}||\epsilon_t - \epsilon_{S_\theta}(z_t,t,c)||_2^2, \quad (1)$$

$$\mathcal{L}_{\text{outKD}} = \min_{S_\theta} \mathbb{E}_{z,\epsilon,t,c}||\epsilon_{T_\theta}(z,t,c) - \epsilon_{S_\theta}(z,t,c)||_2^2, \quad (2)$$

$$\mathcal{L}_{\text{featKD}} = \min_{S_\theta} \mathbb{E}_{z,\epsilon,c,t}||\sum_i f_T^i(z_t,t,c) - f_S^i(z_t,t,c)||_2^2, \quad (3)$$

where S_θ is the compressed U-Net as a student model, ϵ_t is the ground-truth sampled Gaussian noise at timestep t, c is text embedding as a condition, $\epsilon_{S_\theta}(\cdot)$ and $\epsilon_{T_\theta}(\cdot)$ denote the predicted noise from each U-Net in teacher and student model, respectively. $\mathcal{L}_{\text{task}}$ is the task loss for the reverse denoising process [20], $\mathcal{L}_{\text{out}}$ is the output-KD loss [18] computed between the predicted noises from teacher and student respectively, and $\mathcal{L}_{\text{feat}}$ is the feature-wise KD loss [15, 50] computed between the last features $f_T^i(\cdot)$ and $f_T^i(\cdot)$ at i-stage from teacher and student models, respectively.

The feature-wise distillation literature [6, 13, 15, 39, 50] shows that intermediate features play a more critical role in knowledge distillation than output KD [18]. For the feature-wise KD-loss, BK-SDM [26] considers **only the last feature map** at each stage. However, the denoising U-Net consists of several types of features, such as self-attention (`SA`), cross-attention (`CA`), and feedforward net (`FFN`) in the transformer block, and convolutional residual block (`Res`). This means that the feature distillation approach for text-to-image diffusion models has *not been sufficiently explored*, leaving room for further investigation.

In this work, we have performed an in-depth analysis of feature distillation in the U-Net of SDXL [41] as shown in Tab. 3 and observed ***four important findings***. To this end, we ablate feature distillation strategies by using our efficient U-Net (KOALA-1B) as the student model and the U-Net of SDXL as the teacher model. More training details are described in Sec. 5.1. We start from a baseline trained only by $\mathcal{L}_{\text{task}}$ and add $\mathcal{L}_{\text{featKD}}$ without $\mathcal{L}_{\text{outKD}}$ to validate the effect of feature distillation.

F1. Which feature type is effective for distillation? BK-SDM [26] demonstrated that distilling the last features (`LF`) at U-Net stages benefits overall performance, when applied to shallow U-Net of early SDM-v1.4 [47]. However, with the increasing complexity of U-Net and its stage, relying solely on `LF` may not be sufficient to mimic the intricate behavior of the teacher U-Net. To this end, we revisit which features provide the richest guidance for effective knowledge distillation. We focus on key intermediate features from each stage: outputs from the `SA`, `CA`, and `FFN` layers in the transformer block, as well as outputs from `Res` and `LF`. Tab. 3a summarizes the experimental results. While all types of features help obtain higher performance over the naïve baseline with only the task loss, distilling ***self-attention features*** achieves the most performance gain. Considering the prior studies [29, 60, 62] which suggest that `SA` plays a vital role in capturing semantic affinities and the overall structure of images, the results emphasize that such information is crucial for the distillation process.

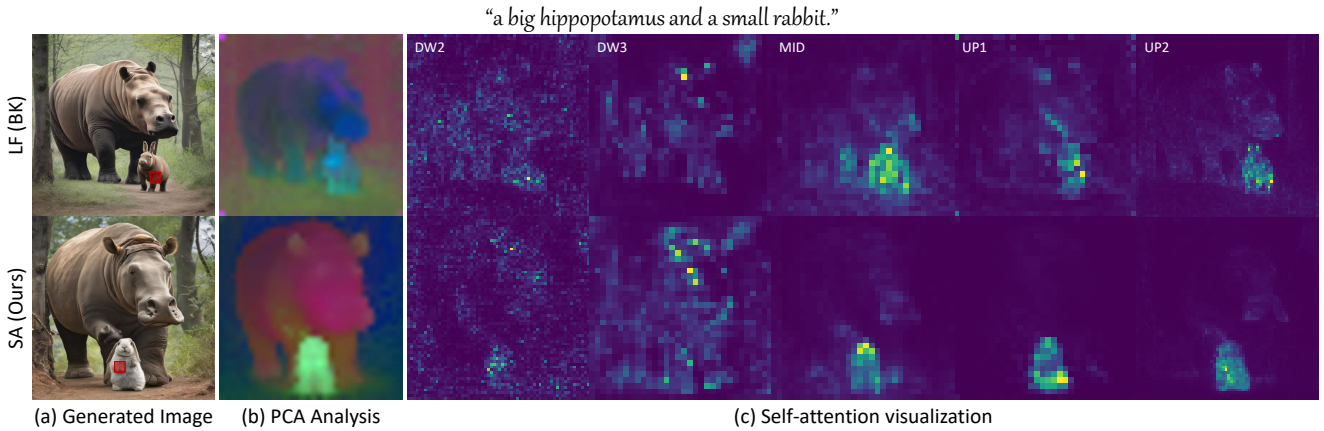

Figure 4. **Analysis on self-attention maps of distilled student U-Nets.** (a) Generated images of LF- and SA-based distilled models, which are BK-SDM [26] and our proposal, respectively. In BK-SDM's result, a rabbit is depicted like a hippopotamus (*i.e.*, appearance leakage). (b) Visualization of PCA analysis results on self-attention maps of `UP-1` stage. (c) Representative visualization of self-attention map from different U-Net stages. Red boxes denote the query patches. Note that from the `MID` stage, the `SA`-based model ***attends*** to the rabbit more ***discriminatively*** than the `LF` model, demonstrating that self-attention-based KD allows to generate objects more distinctly.

To understand the effects more clearly, we illustrate a representative example in the Fig. 4. To reason about how the distilled student U-Net captures self-similarity, we perform a PCA analysis [23, 63] on self-attention maps. Specifically, we apply PCA on self-attention maps from `SA`- and `LF`-based models and show the top three principal components in Fig. 4-(b). Interestingly, in the `SA`-based model, each principal component distinctly represents individual objects (*i.e.*, unique color assignments to each object). This indicates that the `SA`-based model effectively distinguishes different objects in modeling self-similarity, which plays a crucial role in accurately rendering the distinct appearance of each object. In contrast, the `LF`-based model exhibits less distinction between objects, resulting in *appearance leakage* between them (*e.g.*, small hippo with rabbit ears).

F2. Which stage is most effective for distillation? Based on the first finding (F1), we further explore the role and significance of each self-attention stage. To this end, we first visualize the self-attention map in Fig. 4-(c). The self-attention maps initially capture general contextual information (*e.g.*, `DW-2&DW-3`) and gradually focus on localized semantics (*e.g.*, `MID`). In the decoder, self-attentions increasingly correlate with higher-level semantics (*e.g.*, object) to accurately model appearances and structures. Notably, in this stage, the `SA`-based model attends corresponding object regions (given the query patch, red box) more *discriminatively* than the `LF`-based model, which results in improved compositional image generation performance.

In addition, we ablate the significance of each self-attention stage in the distillation process. Specifically, we adopt an `SA`-based loss at a single stage alongside the task loss. As shown in Tab. 3b, the results align with the above understanding: distilling self-attention knowledge within the ***decoder*** stages significantly enhances generation quality. In comparison, the impact of self-attention solely within the encoder stages is less pronounced. Consequently, we opt to retain more `SA` layers within the decoder (see Fig. 3).

F3. Which SA's location is effective in the transformer blocks? At the lowest feature level, the depth of the transformer blocks is 6 for KOALA-1B, so we need to decide which locations to distill from the 10 transformer blocks of teacher U-Net. We assume three cases for each series of transformer blocks; (1) `SA-bottom`: $\{f_T^l \mid l \in \{1,2,3,4,5\}\}$, (2) `SA-interleave`: $\{f_T^l \mid l \in \{1,3,5,7,9,10\}\}$, and (3) `SA-up`: $\{f_T^l \mid l \in \{6,7,8,9,10\}\}$ where l is the number of block. Tab. 3c shows that `SA-bottom` performs the best while `SA-up` performs the worst. This result suggests that the features of the early blocks are more significant for distillation. A more empirical analysis is described in Appendix B.2. Therefore, we adopt the `SA-bottom` strategy in all experiments.

F4. Which combination is the best? In SDXL's U-Net, as shown in Fig. 3, there are no transformer blocks at the highest feature levels (*e.g.*, `DW-1&UP-3`); consequently, self-attention features cannot be distilled at this stage. Thus, we try two options: the residual block (`Res at DW-1&UP-3`) and the last feature (`LF at DW-1&UP-3`) as BK-SDM [26]. To this end, we perform SA-based feature distillation at every stage except for `DW-1` and `UP-3`, where we use the above two options, respectively. In addition, we try additional combinations: `SA+LF all`, `SA+Res all`, and `SA+CA+Res+FFN+LF all` where `all` means all stages). Tab. 3d demonstrates that adding more feature distillations to the SA-absent stage (*e.g.*,

Model	#Param.	HPSv2					Attribute			Object Relationship		Complex	Average
	Whole/U-Net	Anime	Concept-art	Paintings	Photo	Average	Color	Shape	Texture	Spatial	Non-spatial		
SDM-v1.4 [47]	1.04B 860M	27.26	26.61	26.66	27.27	26.95	0.3765	0.3576	0.4156	0.1246	0.3079	0.308	0.3150
SDM-v2.0 [48]	1.28B 865M	27.48	26.89	26.86	27.27	27.13	0.5065	0.4221	0.4922	0.1342	0.3096	0.3386	0.3672
DALLE-2 [44]	6.5B -	27.34	26.54	26.68	27.24	26.95	0.5750	0.5464	0.6374	0.1283	0.3043	0.3696	0.4268
SDXL-Base-1.0 [41]	3.46B 2.6B	27.69	27.44	27.50	28.29	**27.73**	0.6369	0.5408	0.5637	0.2032	0.3110	0.4091	**0.4441**
BK-SDM-S [26]	655M 483M	26.64	26.77	26.87	26.61	26.72	0.3984	0.3783	0.4225	0.0731	0.3003	0.3695	0.3237
Ours-SDM-S	655M 483M	26.73	26.95	27.00	26.74	**26.86**	0.4386	0.3950	0.4549	0.0832	0.3007	0.3777	**0.3417**
BK-SDM-B [26]	752M 580M	27.01	26.64	27.06	26.63	26.84	0.4192	0.4096	0.4409	0.0979	0.3077	0.3052	0.3301
Ours-SDM-B	752M 580M	26.79	27.12	27.11	26.79	**26.95**	0.4436	0.4338	0.4680	0.1077	0.3090	0.3872	**0.3582**
BK-SDXL-700M	1.68B 782M	27.59	27.13	27.17	27.14	27.26	0.5202	0.4506	0.4564	0.1360	0.3008	0.3699	0.3723
KOALA-700M	1.68B 782M	27.65	27.28	27.58	27.21	**27.43**	0.5068	0.4731	0.4674	0.1535	0.3008	0.3731	**0.3791**
BK-SDXL-1B	2.06B 1.16B	27.52	26.90	27.17	26.87	27.12	0.4876	0.4498	0.4578	0.1551	0.3035	0.3777	0.3719
KOALA-1B	2.06B 1.16B	27.73	27.26	27.61	27.16	**27.44**	0.5223	0.5108	0.4864	0.1563	0.3019	0.3697	**0.3912**

Table 4. **Visual aesthetics evaluation** using HPSv2 [67] (Left) and **Image-text alignment evaluation** using T2I-CompBench [21] (Right). Note that BK-SDXL-1B/700M are implemented by ourselves using our efficient U-Net with the BK-SDM [26]'s distillation method. For fair comparison with other methods, we use 50 denoising steps with Euler discrete scheduler [24] same as the huggingface's SDXL-Base-1.0 model [66].

`DW-1&UP-3`) consistently boots performance, and especially `LF at DW1&UP3` shows the best. Interestingly, both `+LF all` and `+Res all` are worse than the ones at only `DW-1&UP-3` and `SA+CA+Res+FFN+LF all` is also not better, demonstrating that the SA features are not complementary to the other features.

With these findings, we build a **KnOwledge-distillAtion-based LAtent diffusion model** with our efficient U-Net, called KOALA. We train our KOALA models with the following objectives: $\mathcal{L}_{\text{task}} + \mathcal{L}_{\text{outKD}} + \mathcal{L}_{\text{featKD}}$ where we apply our findings to $\mathcal{L}_{\text{featKD}}$. As shown in Tab. 2, we design two models, KOALA-1B and KOALA-700M, based on SDXL [41], with U-Net model sizes of 1.16B and 782M, respectively.

5. Experiments

5.1. Implementation details

Dataset. Since the dataset used to train SDXL [41] is not publicly available (*i.e.*, internal data), we train the proposed efficient U-Net in SDXL on *publicly available* LAION-Aesthetics V2 6+ [55, 57, 58] for reproducibility. As the dataset contains some blank text and corrupted images, we filter the data and collect 8,483,623 image-text pairs.

Training. We use the officially released SDXL-Base-1.0 [40] and the training settings, while only replacing its U-Net with our efficient U-Net. We use the same two text encoders used in SDXL, which are OpenCLIP ViT-bigG [22] and CLIP ViT-L [43]. For VAE, we use `sdxl-vae-fp16-fix` [4], which enables us to use FP16 precision for VAE computation. We initialize the weights of our U-Net with the teacher's U-Net weights at the same block location. We train our KOALA models for 100K iterations with a batch size of 128 using four NVIDIA A100 (80GB) GPUs. More details are described in A. For a fair comparison to our counterpart BK-SDM [26], we train our efficient U-Nets with their distillation method under the same data setup (*e.g.*, BK-SDXL-1B and -700M in Tab. 4). Furthermore, we also train SDM-Base and SDM-Small in BK-SDM [26] with our approach (Ours-SDM-Base & Ours-SDM-Small in Tab. 4), following the BK-SDM training recipe.

Evaluation metric. Recently, several works [2, 41, 67, 68] have claimed that FID [17] is not closely correlated with visual fidelity because a feature extractor for FID is pre-trained on the ImageNet [11] dataset, which does not overlap much with the datasets used to train recent text-to-image models (*e.g.*, style, types, resolution, etc.). Therefore, instead of FID, we use **Human Preference Score (HPSv2)** [67] as a visual aesthetics metric, which allows us to evaluate visual quality in terms of more specific types. For image-text alignment, we use the **T2I-compbench** [21], which is a more comprehensive benchmark for evaluating the compositional text-to-image generation capability than the single CLIP score [16].

5.2. Main results

Visual aesthetics. We compare our KOALA-700M/1B models against state-of-the-art text-to-image models, including popular open-sourced Stable diffusion models series [41, 47, 48] and DALLE-2 [44]. Tab. 4 summarizes the results. Our KOALA-700M & KOALA-1B models based on SDXL [41] consistently achieve a higher HPS average score than the BK [26] models (BK-SDXL-700M & 1B) equipped with our efficient U-Net. Moreover, for SDM, Ours-SDM-Base & Smalll models using BK's compressed U-Net in SDM-v1.4 [47] still outperform BK-SDM-Base & Smalll [26]. These results demonstrate that the proposed distillation of the self-attention layer is more helpful for visual aesthetics than the last layer feature distillation by BK [26]. In addition, our KOALA models achieve a higher quality score than DALLE-2 [44], which has a much larger model size (6.5B). Furthermore, our KOALA-700M surpasses SDM-v2.0 [48] with a comparable U-Net size, which is widely used in the community. In Appendix C, we provide the qualitative comparisons to DALLE-2, SDM-v2.0,

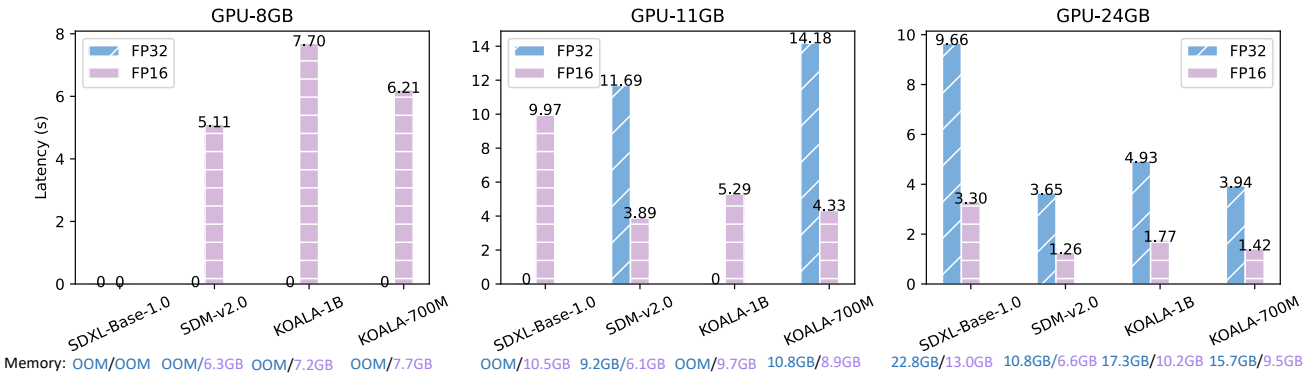

Figure 5. **Latency and memory usage comparison on different GPUs**: NVIDIA 3060Ti (8GB), 2080Ti (11GB), and 4090 (24GB). OOM means *Out-of-Memory*. We measure the inference time of SDM-v2.0 with 768×768 resolution and the other models with 1024×1024. We use 25 denoising steps and FP16/FP32 precisions. Note that SDXL-Base cannot operate in the 8GB-GPU.

A 3d art character of a cute kitty holding a sign that says "Let there be peace", pixar style

A baby penguin wearing a blue hat and red gloves

Figure 6. **Failure cases of KOALA-700M**

and SDXL-Base-1.0, supporting the quantitative results.

Image-text alignment. As shown in Tab. 4 (Right), our approach with both SDXL and SDM consistently surpasses the counterpart BK method [26] in terms of text-image alignment. We conjecture that this is because our self-attention-based KD approach allows the model to learn more discriminative representations between objects or attributes, as demonstrated in Sec. 4.2 and Fig. 4. Meanwhile, unlike the aesthetics results (HPSv2), our models lag behind DALLE-2 regarding the average score of the CompBench. In attribute binding (color, shape, and texture), our model lags behind DALLE-2 but outperforms in object-relationship metrics. We speculate that the different tendency between DALLE-2 and our model may stem from data used for training. Because the LAION-Aesthetics V2 6+ [57] data we used focuses on higher aesthetic images than multiple objects with various attributes, our model is vulnerable to texts with different attribute properties.

5.3. Model budget comparison

We further validate the efficiency of our model by measuring its inference speed on a variety of *consumer-grade* GPUs with different memory sizes, such as 8GB (3060Ti), 11GB (2080Ti), and 24GB (4090), because the GPU environment varies for each user. Fig. 5 illustrates inference speed and GPU memory usage on different GPUs with both FP16 and FP32 precisions. For this experiment, we compare against the most popular open-sourced models, SDM-v2.0 [48] and SDXL-Base-1.0 [41]. On the 8GB GPU, *SDXL does not fit*, but the other models can run in FP16 precision. Notably, KOALA-700M generates higher-resolution images of superior quality at a comparable inference speed to SDM-v2.0. On the 11GB GPU, SDXL can run with FP16 precision, and on the 24 GB, it can run at 9.66s and 3.3s with both FP16 and FP32 precision, respectively. On the other hand, our KOALA-700M runs at 3.94s and 1.42s, which is $2\times$ faster than SDXL. Overall, our KOALA-700M is the best alternative for high-quality image generation that can replace SDM-v2.0 and SDXL in resource-constrained GPU environments.

6. Limitations and Future Work

While our KOALA models generate images with impressive aesthetic quality, such as the photo-realistic or 3d-art renderings shown in Fig. 1, it still shows limitations in several specific cases:

Rendering long legible text. Our models have difficulty in synthesizing legible texts in the generated image. For example, it renders unintended letters or generates unintelligible letters, as shown in Fig. 6 (Left).

Complex prompt with multiple attributes. When attempting to compose an image using prompts that include various attributes of an object or scene, KOALA sometimes generates instances that do not perfectly follow the intended description. For example, as shown in Fig. 6 (Right), when we configure the penguin to wear a blue hat and red gloves, only the blue hat attribute is applied, while the red gloves are not.

We conjecture that these limitations may stem from the dataset, LAION-aesthetics-V2 6+ [57], we used to train, whose text prompts are relatively shorter (lacking detail) and messy (*e.g.*, HTML code). Recent works [1, 69] also pointed out this issue and showed that utilizing machine-generated detailed captions (*i.e.*, synthesized captions) improves the fine-grained text-alignment of T2I models. For future work, it will also be interesting to see the synergies between our efficient T2I model and such large multimodal models [5, 33, 34]-based recaptioning techniques. More failure cases are illustrated in Appendix C.4.

7. Conclusion

In this work, we propose KOALA, an efficient text-to-image synthesis model, offering a compelling alternative between SDM-v2.0 and SDXL in resource-limited environments. To achieve this, we devise more compact U-Nets by effectively compressing the main computational bottlenecks present in SDXL. In doing so, we demonstrate that self-attention-based knowledge distillation is one of the most crucial components to enhance the quality of generated images. With these contributions, our KOALA-700M model substantially reduces the model size (69%↓) and the latency (60%↓) of SDXL, while exhibiting decent aesthetic generation quality.

8. Acknowledgments

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. RS-2022-00187238, Development of Large Korean Language Model Technology for Efficient Pre-training).

References

[1] James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, Wesam Manassra, Prafulla Dhariwal, Casey Chu, Yunxin Jiao, and Aditya Ramesh. Improving image generation with better captions. https://cdn.openai.com/papers/dall-e-3.pdf, 2023. 9

[2] Eyal Betzalel, Coby Penso, Aviv Navon, and Ethan Fetaya. A study on the evaluation of generative models. *arXiv preprint arXiv:2206.10935*, 2022. 2, 7

[3] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *CVPR*, 2023. 1

[4] Ollin Boer Bohan. Sdxl-vae-fp16-fix. https://huggingface.co/madebyollin/sdxl-vae-fp16-fix, 2023. 3, 7, 12

[5] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020. 9

[6] Hanting Chen, Yunhe Wang, Chang Xu, Chao Xu, and Dacheng Tao. Learning student networks via feature embedding. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):25–35, 2020. 5

[7] Yu-Hui Chen, Raman Sarokin, Juhyun Lee, Jiuqiang Tang, Chuo-Ling Chang, Andrei Kulik, and Matthias Grundmann. Speed is all you need: On-device acceleration of large diffusion models via gpu-aware optimizations. In *CVPR-Workshop*, 2023. 1

[8] Jooyoung Choi, Yunjey Choi, Yunji Kim, Junho Kim, and Sungroh Yoon. Custom-edit: Text-guided image editing with customized diffusion models. In *CVPR-Workshop*, 2023. 1

[9] Benjamin Consolvo. Text-to-image stable diffusion with stability ai and compvis on the latest intel gpu. https://medium.com/intel-analytics-software, 2022. 1

[10] DeepFloyd. Deepfloyd. https://www.deepfloyd.ai/, 2022. 3

[11] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 7

[12] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit Haim Bermano, Gal Chechik, and Daniel Cohen-or. An image is worth one word: Personalizing text-to-image generation using textual inversion. In *ICLR*, 2023. 1

[13] Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International Journal of Computer Vision*, 2021. 5

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4

[15] Byeongho Heo, Jeesoo Kim, Sangdoo Yun, Hyojin Park, Nojun Kwak, and Jin Young Choi. A comprehensive overhaul of feature distillation. In *ICCV*, 2019. 1, 3, 5

[16] Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. Clipscore: A reference-free evaluation metric for image captioning. In *EMNLP*, 2021. 7

[17] 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. In *NeurIPS*, 2017. 2, 7

[18] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. 1, 5

[19] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 12, 13, 15, 16, 17, 18

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 3, 5, 12

[21] Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2i-compbench: A comprehensive benchmark for open-world compositional text-to-image generation. In *NeurIPS*, 2023. 2, 7

[22] Gabriel Ilharco, Mitchell Wortsman, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip. https://github.com/mlfoundations/open_clip, 2021. 3, 7, 12

[23] Ian T Jolliffe and Jorge Cadima. Principal component analysis: a review and recent developments. *Philosophical transactions of the royal society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202, 2016. 6, 12

[24] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models. *NeurIPS*, 2022. 2, 7, 12, 13, 15, 16, 17, 18

[25] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. In *ICCV*, 2023. 1

[26] Bo-Kyeong Kim, Hyoung-Kyu Song, Thibault Castells, and Shinkook Choi. On architectural compression of text-to-image diffusion models. *arXiv preprint arXiv:2305.15798*, 2023. 1, 2, 3, 4, 5, 6, 7, 8, 12, 13, 14

[27] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 3, 5

[28] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. In *ICCV*, 2023. 2

[29] Nicholas Kolkin, Jason Salavon, and Gregory Shakhnarovich. Style transfer by relaxed optimal transport and self-similarity. In *CVPR*, 2019. 5

[30] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *CVPR*, 2023. 1

[31] Yanyu Li, Huan Wang, Qing Jin, Ju Hu, Pavlo Chemerys, Yun Fu, Yanzhi Wang, Sergey Tulyakov, and Jian Ren. Snapfusion: Text-to-image diffusion model on mobile devices within two seconds. *arXiv preprint arXiv:2306.00980*, 2023. 1

[32] Chen-Hsuan Lin, Jun Gao, Luming Tang, Towaki Takikawa, Xiaohui Zeng, Xun Huang, Karsten Kreis, Sanja Fidler, Ming-Yu Liu, and Tsung-Yi Lin. Magic3d: High-resolution text-to-3d content creation. In *CVPR*, 2023. 1

[33] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning, 2023. 9

[34] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning, 2023. 9

[35] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 12

[36] Cheng Lu, Yuhao Zhou, Fan Bao, Jianfei Chen, Chongxuan Li, and Jun Zhu. Dpm-solver: A fast ode solver for diffusion probabilistic model sampling in around 10 steps. *Advances in Neural Information Processing Systems*, 35:5775–5787, 2022. 14

[37] Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. On distillation of guided diffusion models. In *CVPR*, 2023. 1, 3

[38] Chong Mou, Xintao Wang, Liangbin Xie, Yanze Wu, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. 1

[39] Peyman Passban, Yimeng Wu, Mehdi Rezagholizadeh, and Qun Liu. Alp-kd: Attention-based layer projection for knowledge distillation. In *AAAI*, 2021. 5

[40] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Stable-diffusion-xl-base-1.0. https://huggingface.co/stabilityai/stable-diffusion-xl-base-1.0, 2023. 7, 12

[41] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 1, 2, 3, 4, 5, 7, 8, 12, 13

[42] Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022. 1

[43] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 3, 7, 12

[44] 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. 3, 7, 13

[45] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, 2014. 3, 5

[46] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 3, 5

[47] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. Stable-diffusion-v1.4. https://github.com/CompVis/stable-diffusion, 2022. 1, 3, 4, 5, 7

[48] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. Stable-diffusion-v2.0. https://github.com/Stability-AI/stablediffusion, 2022. 1, 3, 7, 8

[49] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. Stable-diffusion-v2.0. https://huggingface.co/stabilityai/stable-diffusion-2, 2022. 13, 15, 16, 17, 18

[50] Adriana Romero, Nicolas Ballas, Samira Ebrahimi Kahou, Antoine Chassang, Carlo Gatta, and Yoshua Bengio. Fitnets: Hints for thin deep nets. *arXiv preprint arXiv:1412.6550*, 2014. 3, 5

[51] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 3, 5

[52] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *CVPR*, 2023. 1, 14

[53] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. 2022. 3

[54] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. In *ICLR*, 2022. 1, 3

[55] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-aestheics v2. https://laion.ai/blog/laion-aesthetics/, 2022. 7

[56] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-aesthetics v2 6.5+. https://huggingface.co/datasets/ChristophSchuhmann/improved_aesthetics_6.5plus, 2022. 12

[57] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-aesthetics v2 6+. https://huggingface.co/datasets/ChristophSchuhmann/improved_aesthetics_6plus, 2022. 2, 7, 8, 9, 12

[58] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. In *NeurIPS*, 2022. 7

[59] Yuzhang Shang, Zhihang Yuan, Bin Xie, Bingzhe Wu, and Yan Yan. Post-training quantization on diffusion models. In *CVPR*, 2023. 1

[60] Eli Shechtman and Michal Irani. Matching local self-similarities across images and videos. In *CVPR*, 2007. 5

[61] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 3, 12, 13, 15, 16, 17, 18

[62] Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. In *CVPR*, 2022. 5

[63] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*, 2023. 1, 6

[64] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers, 2022. 12

[65] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers/blob/main/examples/text_to_image/train_text_to_image_sdxl.py, 2023. 12

[66] Patrick von Platen, Suraj Patil, Anton Lozhkov, Pedro Cuenca, Nathan Lambert, Kashif Rasul, Mishig Davaadorj, and Thomas Wolf. Stabilityai: Sdxl-base-1.0. https://huggingface.co/stabilityai/stable-diffusion-xl-base-1.0, 2023. 2, 7, 13, 15, 16, 17

[67] Xiaoshi Wu, Yiming Hao, Keqiang Sun, Yixiong Chen, Feng Zhu, Rui Zhao, and Hongsheng Li. Human preference score v2: A solid benchmark for evaluating human preferences of text-to-image synthesis. *arXiv preprint arXiv:2306.09341*, 2023. 2, 7, 13

[68] Xiaoshi Wu, Keqiang Sun, Feng Zhu, Rui Zhao, and Hongsheng Li. Better aligning text-to-image models with human preference. In *ICCV*, 2023. 2, 7

[69] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv preprint arXiv:2206.10789*, 2022. 9

[70] Xiaohui Zeng, Arash Vahdat, Francis Williams, Zan Gojcic, Or Litany, Sanja Fidler, and Karsten Kreis. Lion: Latent point diffusion models for 3d shape generation. *arXiv preprint arXiv:2210.06978*, 2022. 1

[71] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 1

Appendix

A. Implementation details

A.1. Training

We use the officially released SDXL-Base-1.0 [40] with `Diffusers` library [64, 65], while only replacing its U-Net with our efficient U-Net. We use the same two text encoders used in SDXL, which are OpenCLIP ViT-bigG [22] and CLIP ViT-L [43]. For VAE, we use `sdxl-vae-fp16-fix` [4], which enables us to use FP16 precision for VAE computation. We initialize the weights of our U-Net with the teacher's U-Net weights at the same block location. We freeze the text encoders, VAE, and the teacher U-Net of SDXL and only fine-tune our U-Net. We train our KOALA models on LAION-Aesthetics V2 6+ [57] dataset (about 800M text-image pairs) for 100K iterations using four NVIDIA A100 (80GB) GPUs with a resolution of 1024×1024, a discrete-time diffusion schedule [20], size- and crop-conditioning as in SDXL [41], a batch size of 128, AdamW optimizer [35], a constant learning rate of 10^{-5}, and FP16 precision. For a fair comparison to our counterpart BK-SDM [26], we train our efficient U-Nets with their distillation method under the same data setup (*e.g.*, BK-SDXL-1B and -700M in Tab. 4). Furthermore, we also train SDM-Base and SDM-Small in BK-SDM [26] with our approach (Ours-SDM-Base & Ours-SDM-Small in Tab. 4), following the BK-SDM training recipe on LAION-Aesthetics V2 6.5+ [56]. For the ablation studies in Tab. 3, we train all models for 30K iterations with a batch size of 32 on LAION-Aesthetics V2 6.5+ datasets for fast verification.

A.2. Inference

When generating samples, we also use FP16-precision and `sdxl-vae-fp16-fix` [4] for VAE-decoder. Note that in the SDXL original paper [41], authors used DDIM sampler [61] to generate samples in the figures while the diffuser's official SDXL code [65] used Euler discrete scheduler [24] as the default scheduler. Therefore, we also use the Euler discrete scheduler for generating samples. With the Euler discrete scheduler, we set the denoising step to 50 only for quantitative evaluation in Tab. 3 and Tab. 4, and set it to 25 for other qualitative results or latency measurements. we set classifier-free guidance [19] to 7.5.

A.3. Implementation details of `SA-bottom`

Fig. 7 illustrates how to choose transformer blocks when distilling self-attention (SA) features at `DW3 & MID & UP1` as described in Sec. 4.2 (F.3) and Tab. 3c. In Fig. 7, the Transformer blocks (yellow) with a depth of 10 is from the original SDXL's U-Net teacher model, and the Transformer blocks (blue) with a depth of 6 is from our KOALA-1B's U-Net student model. For `SA-bottom` in Tab. 3c, we perform feature distillation by selecting consecutive blocks from the teacher model's transformer blocks, starting with the first one, and comparing to each transformer's self-attention (SA) features from the student model's transformer blocks.

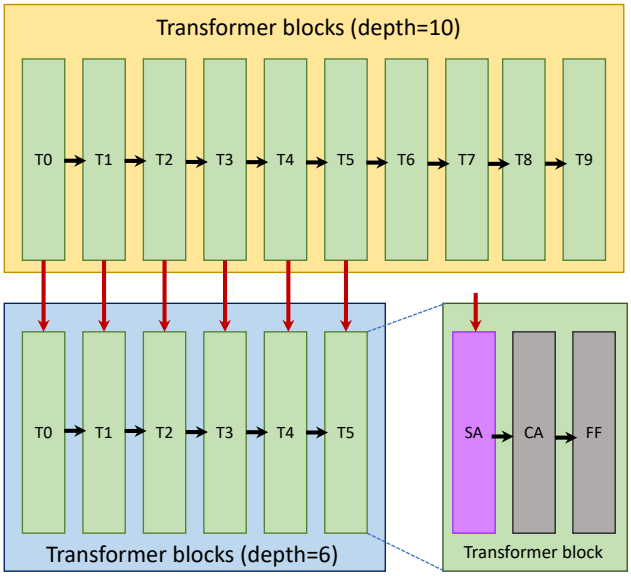

Figure 7. **`SA-bottom` illustration** in Tab. 3c.

B. Additional Analysis

B.1. Attention visualization for Tab. 3a and Tab. 3b

In Section 4.3 of the main paper, we provide empirical evidence demonstrating the paramount importance of self-attention features in the distillation process. Our findings particularly highlight the significant impact of specific self-attention (`SA`) stages (*e.g.*, `UP-1&UP-2`) on enhancing performance. To support these results, we extensively analyze self-attention maps in the main paper. To complete the analysis, we expand our Principal Component Analysis [23] (PCA) on self-attention maps to encompass all layers in Fig. 9.

As elaborated in the main paper, self-attention begins by capturing broad contextual information (*e.g.*, `DW-2&DW-3`) and then progressively attends to localized semantic details (*e.g.*, `MID`). Within the decoder, self-attentions are increasingly aligned with higher-level semantic elements `UP-1&UP-2`), such as objects, for facilitating a more accurate representation of appearances and structures. Notably, at this stage, the `SA`-based model focuses more on specific object regions than the `LF`-based model. This leads to a marked improvement in compositional image generation performance.

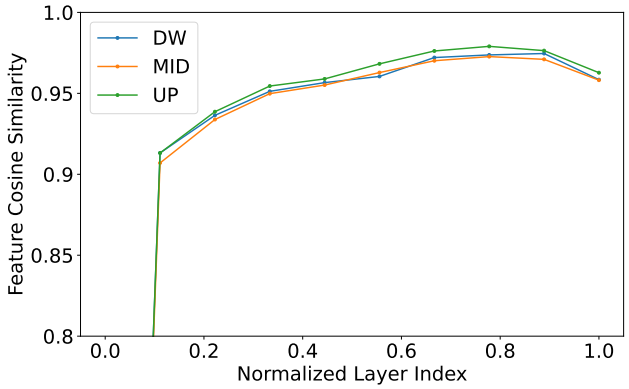

Figure 8. **Feature cosine similarity analysis.** We plot the cross-layer cosine similarity against the normalized layer indexes of transformer block.

B.2. Feature cosine similarity analysis for Tab. 3c

KOALA models compress the computationally intensive transformer blocks in the lowest feature levels (*i.e.*, `DW-3&Mid&UP-1` stages). Specifically, we reduce the depth of these transformer blocks from 10 to 5 for KOALA-1B and to 6 for KOALA-700M. For this purpose, we demonstrate that distilling knowledge from the consecutive bottom layers of transformer blocks is a simple yet effective strategy (see third finding (F3) in the main paper).

To delve deeper into the rationale behind this strategy, we conducted a thorough feature analysis of the original SDXL model [41]. In particular, we investigate the evolution of the features within the transformer blocks. We compute the cross-layer cosine similarity between the output features of each block and those of its predecessors. A lower similarity score indicates a significant contribution of the current block, whereas a higher score implies a marginal contribution.

For this analysis, we leverage the diverse domain of prompts in the HPSv2 dataset [67]. We compute the cross-layer cosine similarity for each stage (`DW&Mid&UP`) and average these values across all prompts. The results are illustrated in Fig. 8. For all stages, transformer blocks exhibit a tendency of feature saturations: While early transformer blocks generally show a significant contribution, later blocks have less impact. This motivates us to distill the learned knowledge of consecutive bottom layers of transformer blocks for minimal performance degradation.

C. Qualitative results

C.1. Comparison to other methods

We show generated samples with various types of prompt style (*e.g.*, portrait photo, 3d art animation, and paintings) comparing with DALLE-2 [44], SDM-v2.0[1] and SDXL[2] in Figs. 10 to 12. When generating samples, we set the same random seed for all models except DALLE-2 because for DALLE-2 API, we cannot set the random seed. Overall, our method surpasses DALLE-2 and SDM-v2.0 in terms of visual aesthetic quality and demonstrates decent results when compared to SDXL.

We also compare our model to Stable diffusion models, SDM-v2.0 and SDXL, with four random seeds in Fig. 13. We follow the official inference setups of each model (SDM-v2.0 [49] and SDXL-Base-1.0 [66]) using `huggingface`. Specifically, SDM-v2.0 is set to generate with DDIM scheduler [61] with 25 steps and SDXL and ours are set to use Euler discrete scheduler [24] with 25 steps. And we set all models to use the classifier-free guidance [19] with 7.5. The results highlight the robustness of the KOALA model against varying random seed selection.

C.2. Comparison to BK-SDM

In addition to the quantitative comparisons in the main paper, we also provide a qualitative comparison with BK-SDM [26]. As illustrated in Fig. 14, BK-SDM occasionally overlooks specific attributes or objects mentioned in the text prompt and generates structurally invalid images. On the contrary, our proposed model consistently generates images with enhanced adherence to the text, showcasing a superior ability to capture the intended details accurately.

C.3. Self-Comparison on KOALA models

KOALA-1B vs. KOALA-700M To assess the influence of model size on performance, we conducted a comparative analysis between KOALA-1B and KOALA-700M. The results, as illustrated in Fig. 15 and Fig. 16, reveal that KOALA-1B is able to capture finer details and exhibits a marginally superior adherence to text prompts. However, despite its smaller size, KOALA-700M still delivers impressive results, with the added advantage of a significantly faster inference time.

Controllability of KOALA The KOALA-700M model exhibits not only faithful visual quality but also remarkable controllability in response to the given text prompts. Fig. 17 highlights the impressive controllability for challenging cases, including various painter's styles (see the first row), diverse color reflections (see the second row), a range of seasons (see the third row), and different times of day (see the fourth row).

C.4. Failure cases

As noted in the main paper, the KOALA models have certain limitations despite their great aesthetic quality. To thor-

[1] https://huggingface.co/stabilityai/stable-diffusion-2

[2] https://huggingface.co/stabilityai/stable-diffusion-xl-base-1.0

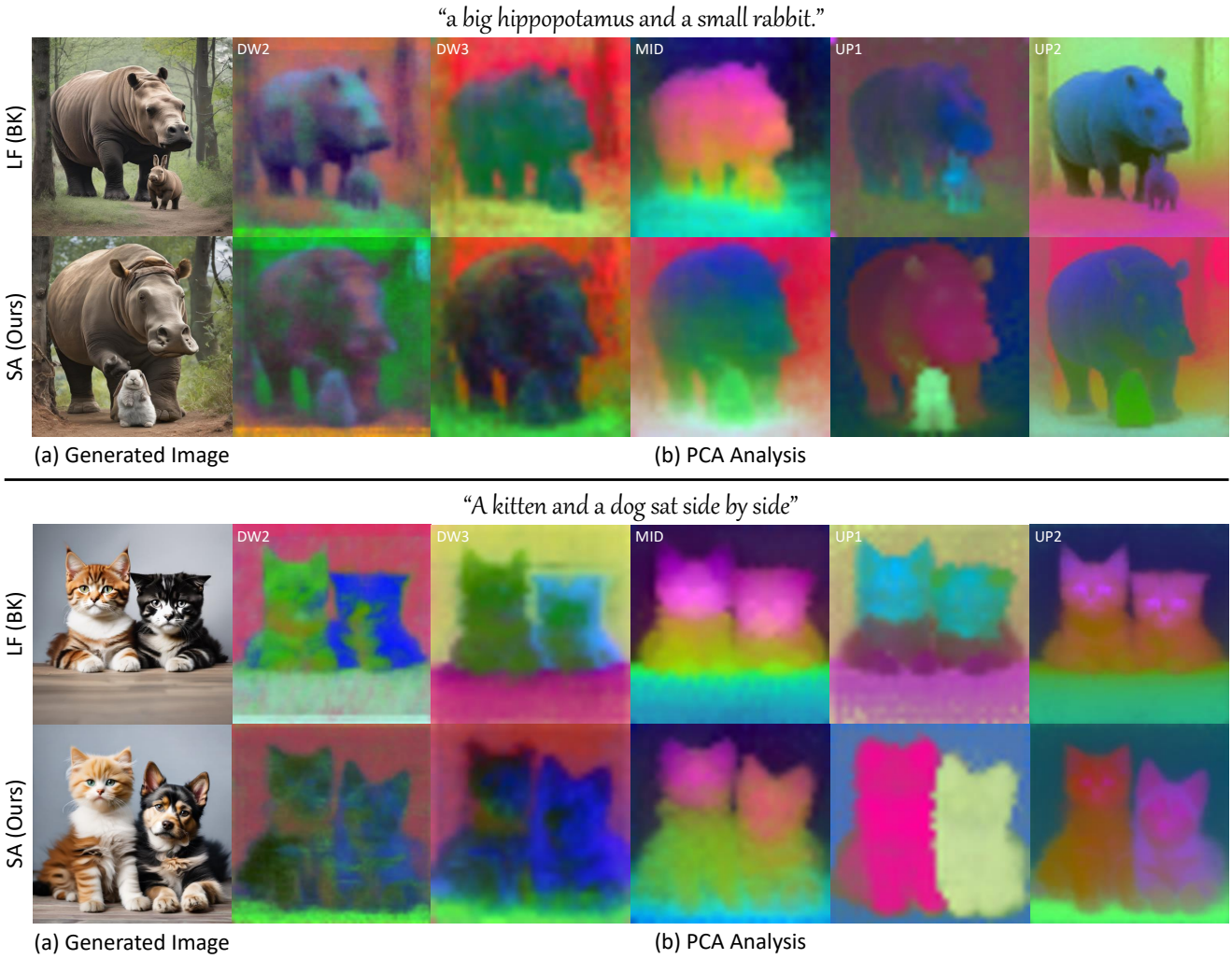

Figure 9. **Extended analysis on self-attention maps of distilled student U-Nets.** (a) Generated images of LF- and SA-based distilled models, which are BK-SDM [26] and our proposal, respectively. In BK-SDM's result, a rabbit or dog is depicted like a hippopotamus or cat, repectively (*i.e.*, appearance leakage). (b) Visualization of PCA analysis results. Note that from the `UP-1` stage, the `SA`-based model ***attends*** to the corresponding object (*i.e.*, rabbit or dog) more ***discriminatively*** than the `LF` model, demonstrating that self-attention-based KD allows to generate objects more distinctly.

oughly understand these constraints, we present additional examples and categorize them into distinct cases. Fig. 18 clearly demonstrates that the KOALA-700M model faces challenges in complex scenarios, such as complex compositional prompts with multiple attributes (the first row), rendering legible text (the second row), capturing intricate structural details (the third row), and accurately depicting human hands (the fourth row).

D. Downstream task: Dreambooth

To validate the transferability and generation capability of our KOALA model, we apply our KOALA-700M model to a custom text-to-image (T2I) downstream task, Dreambooth [52], which is a popular custom model for personalized T2I generation. We fine-tune our KOALA-700M model on the Dreambooth dataset using resizing 1024, the 8-bit Adam optimizer, a constant learning rate of 5e-5, and a batch size of 4 for 500 iterations without the incorporation of a class-preservation loss. The number of steps for gradient accumulation is set to 2. For generating images, we use DPM-Solver [36] with 25 denoising steps. As shown in Fig. 19, with about 5-6 photographs provided, subject training is conducted alongside an identifier token, taking approximately 20 minutes per subject on an NVIDIA RTX A6000 GPU. The results demonstrate that the images are generated seamlessly, without any inconsistencies between the text and the object.

Figure 10. **Qualitative comparison with DALLE-2, SDM-v2.0, and SDXL in terms of portrait photo**. We follow the official inference setups of each model (SDM-v2.0 [49] and SDXL-Base-1.0 [66]) using `huggingface`. Specifically, SDM-v2.0 is set to generate with DDIM scheduler [61] with 25 steps and SDXL and ours are set to use Euler discrete scheduler [24] with 25 steps. And we set all models to use the classifier-free guidance [19] with 7.5.

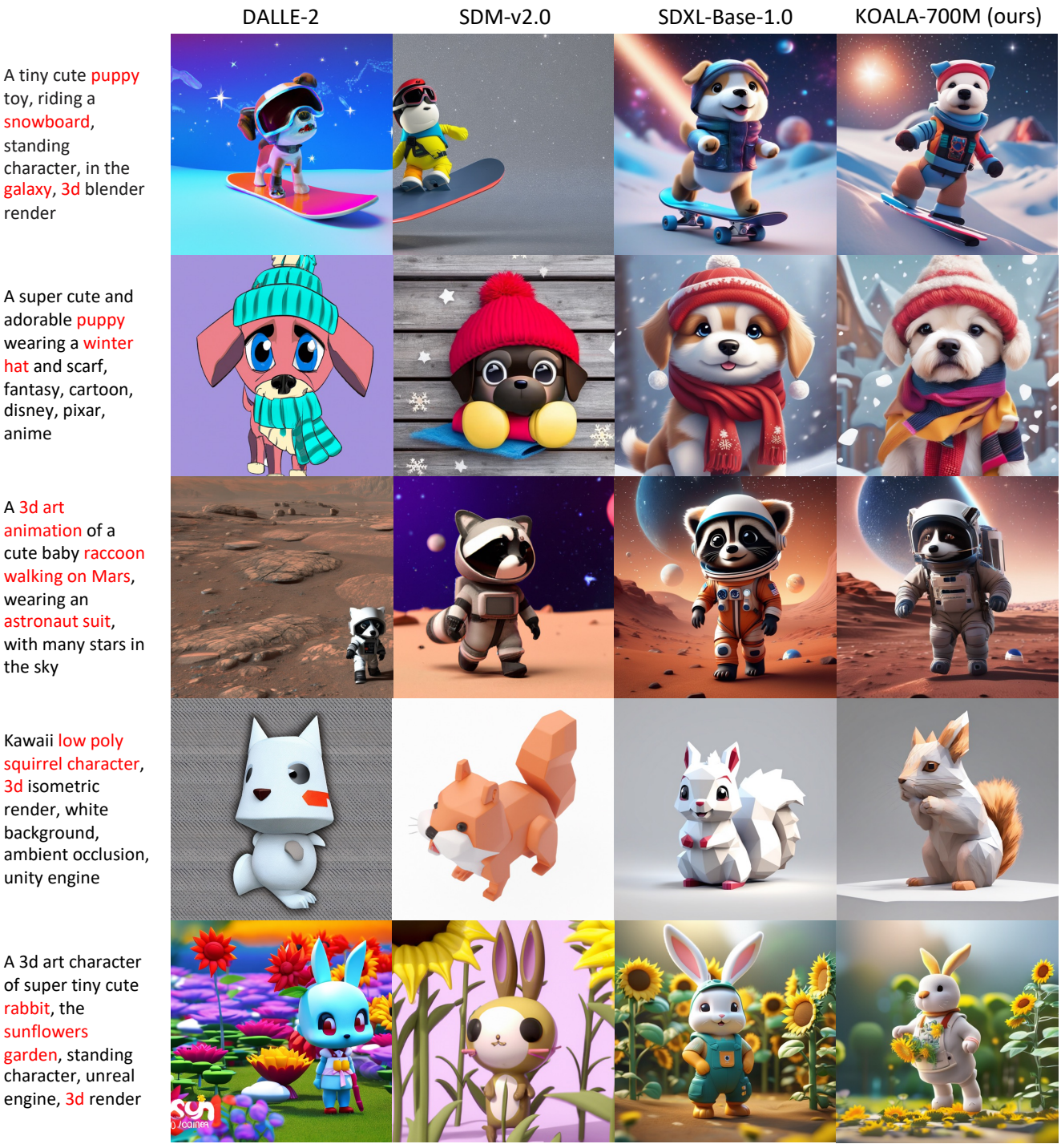

Figure 11. **Qualitative comparison with DALLE-2, SDM-v2.0, and SDXL in terms of 3D art**. We follow the official inference setups of each model (SDM-v2.0 [49] and SDXL-Base-1.0 [66]) using `huggingface`. Specifically, SDM-v2.0 is set to generate with DDIM scheduler [61] with 25 steps and SDXL and ours are set to use Euler discrete scheduler [24] with 25 steps. And we set all models to use the classifier-free guidance [19] with 7.5.

Figure 12. **Qualitative comparison with DALLE-2, SDM-v2.0, and SDXL in terms of painting**. We follow the official inference setups of each model (SDM-v2.0 [49] and SDXL-Base-1.0 [66]) using `huggingface`. Specifically, SDM-v2.0 is set to generate with DDIM scheduler [61] with 25 steps and SDXL and ours are set to use Euler discrete scheduler [24] with 25 steps. And we set all models to use the classifier-free guidance [19] with 7.5.

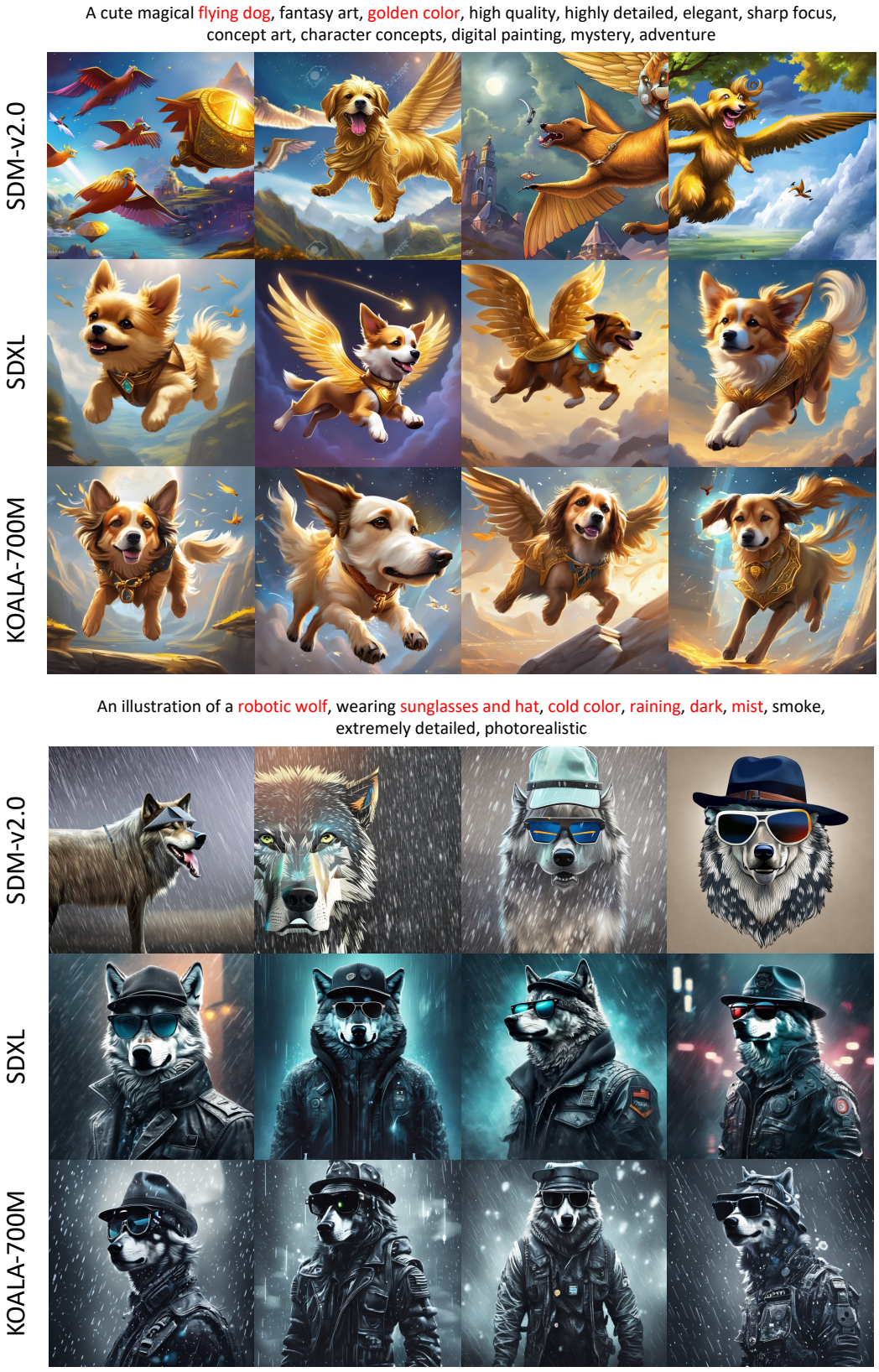

Figure 13. **Qualitative comparison between SDM-v2.0 vs. SDXL-Base-1.0 vs. KOALA-700M (ours).** For each prompt, we use 4 random seeds to generate images, while all models are generated with the same seed for each image. SDM-v2.0 [49] is set to generate with DDIM scheduler [61] and SDXL and ours are set to use Euler discrete scheduler [24]. All samples are generated with 25 denoising steps and the cfg-scale [19] 7.5.

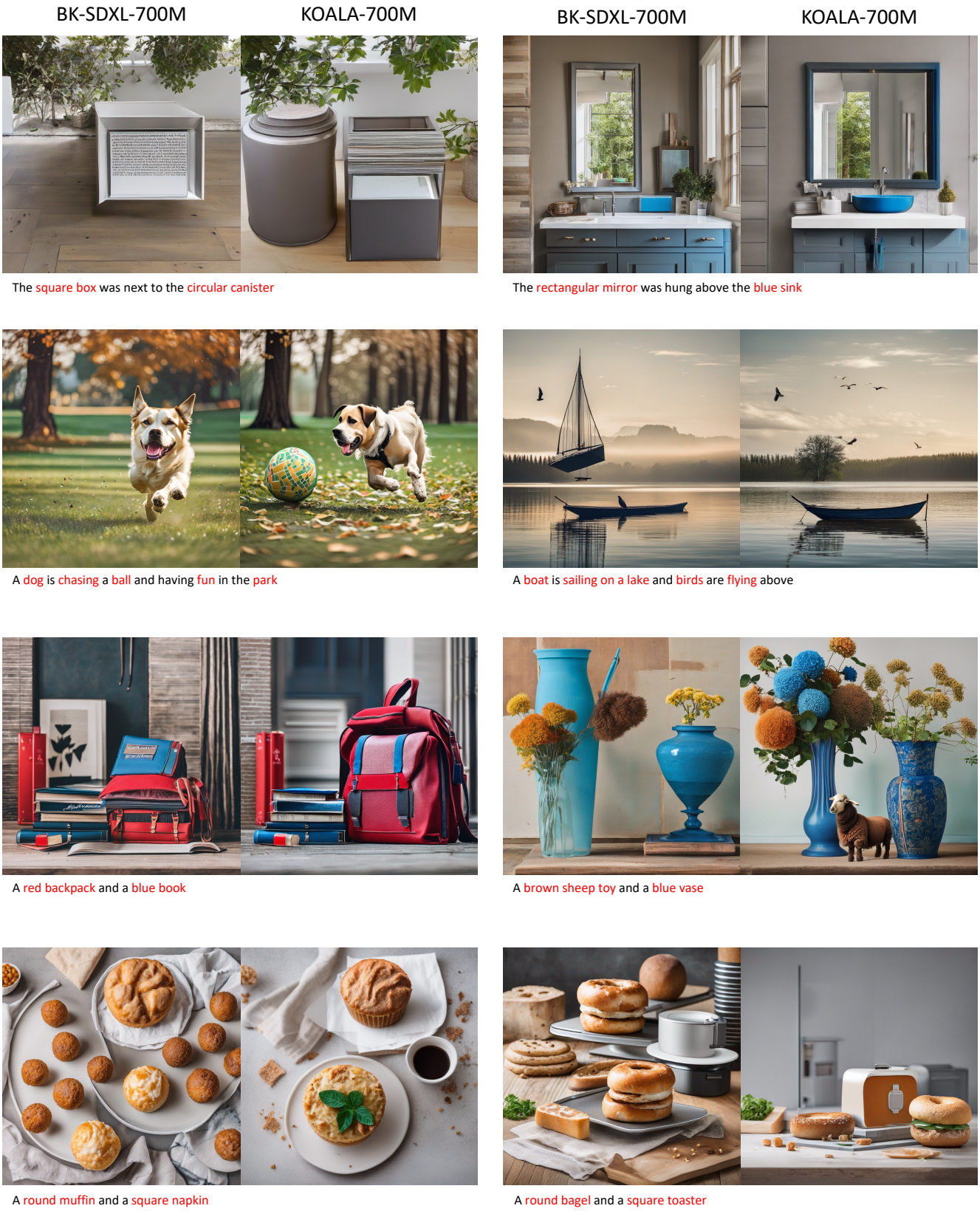

Figure 14. **Qualitative comparison between BK-Base-700M vs. KOALA-700M (ours).**

KOALA-1B KOALA-700M

Pencil painting of young girl

KOALA-1B KOALA-700M

Renaissance-style portrait of an astronaut in space, detailed starry background, reflective helmet

Abstract watercolor anime art of a magical girl surrounded by flowers, 8k, stunning intricate details, by artgerm

A blue haired girl, with blowing bubbles, with a sophisticated intellectual style, anime. dark, cold color

A 3d art character of a tiny cute rabbit, big reflect eyes, wearing a hoodie, in the city, full body shot, 3D, character, 3d rendering, realistic, adorable, physically based rendering

Vibrant painting of a flowerpunk owl, dramatic lighting, abstract flowers, hightly detailed digital painting, 8K

Cute Cat in a Variety of Colors, Universe fulfilling the body, fantasy, renaissance aesthetic, Star Trek aesthetic, pastel colors aesthetic, intricate fashion clothing, highly detailed, surrealistic, digital painting, concept art, sharp focus, illustration

A vitral window, Art Noveau style, with colorful nature motives and a big fox in the middle, rounded upside, photorealistic

Figure 15. **Generated samples of KOALA-1B and KOALA-700M**.

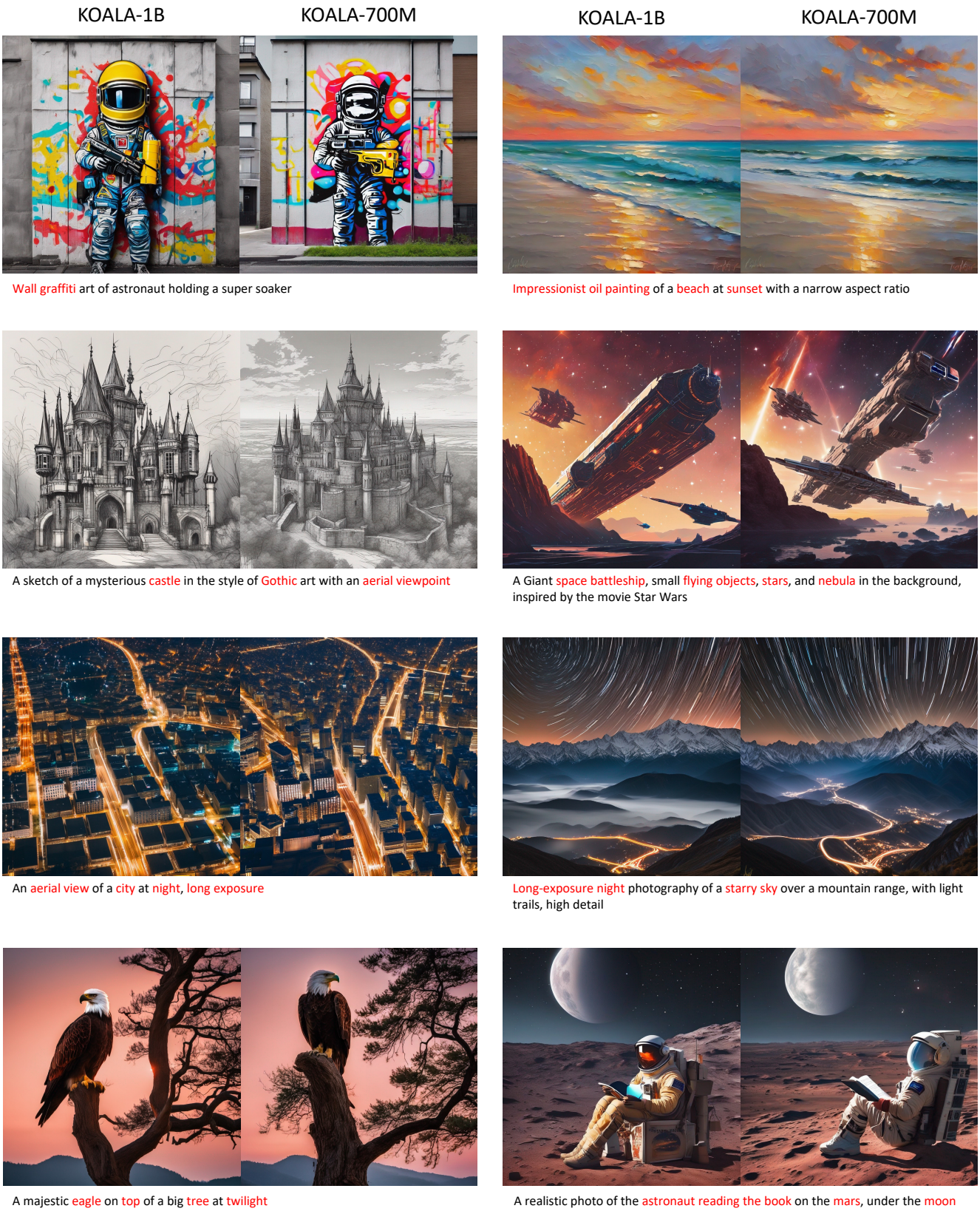

Figure 16. **Generated samples of KOALA-1B and KOALA-700M**.

Figure 17. **Generated samples of KOALA-700M**. Our KOALA-700M model demonstrates faithful visual quality and remarkable controllability across various painter's styles (see the first row), diverse color reflections (see the second row), a range of seasons (see the third row), and different times of day (see the fourth row), in response to the given text prompts.

A corgi wearing a red bowtie and a purple party hat

A white dog (left) and a black cat (right) are running together with smiling

A close-up photo of a young boy with a white hat, blue t-shirt, and denim jacket

A photo of a black handbag and a green wallet on the wooden floor

A neon sign that reads “Backprop”

A photo of a store with the name "koala grocery store"

A picture of a sticky note on desk that says "I can do it."

A photo of a red sign that says "No Trespassing" in the desert.

A photo of the Burj Khalifa, extremely detailed, night, best quality, professional photography, light, strong contrast

A photo of the Tokyo Tower in the spring

A professional photo of a cultural heritage landscape with a traditional Korean house, in the background of a colorful sky, best quality, RAW photo, UHD, DSLR, soft lighting, high quality, film grain, night, light, strong contrast, contrast, glass reflection, high detailed building, building, time-lapse photography, indoor light, reflective, half aerial view, water, bridge

A professional photo of the Colosseum at sunset

A close-up photo of hand full of maple leaves.

A photorealistic image of a young girl blowing bubbles in a park, with colorful flowers and a big blue sky in the background. Shot from a close-up angle to capture the sense of playfulness and innocence

A photo of an old lady with her hand raised in greeting.

A woman is speaking on the phone and making plans with a friend.

Figure 18. **Failure cases of KOALA-700M**. KOALA-700M model faces challenges in complex scenarios, such as complex compositional prompts with multiple attributes (1st row), rendering legible text (2nd row), capturing intricate structural details (3rd row), and accurately depicting human hands (4th row).

Figure 19. **Image Generations with Dreambooth+KOALA-700M**.