FlexEControl: Flexible and Efficient Multimodal Control for Text-to-Image Generation — Appendix

Anonymous authors
Paper under double-blind review

.

This Appendix is organized as follows:

- Appendix A contains our detailed model architectures;
- Appendix B contains additional implementation details;
- Appendix C contains additional results;
- Appendix D contains additional study on text-to-image pretraining;
- Appendix E contains additional related works;
- Appendix F contains details for the human evaluation setup;

A Detailed Model Architecture

In ControlNet (Zhang and Agrawala, 2023) and Uni-ControlNet (Zhao et al., 2023), the weights of Stable Diffusion (SD) (Rombach et al., 2022) are fixed and the input conditions are fed into zero-convolutions and added back into the main Stable Diffusion backbone. Specifically, for Uni-ControlNet, they uses a multi-scale condition injection strategy that extracts features at different resolutions and uses them for condition injection referring to the implementation of Feature Denormalization (FDN):

$$\begin{aligned}\mathrm{FDN}\left(Z, c\right) = \mathrm{norm}\left(Z\right) \cdot \left(1 + \Phi\left(\mathrm{zero}\left(h_r\left(c\right)\right)\right)\right) \\ + \Phi\left(\mathrm{zero}\left(h_r\left(c\right)\right)\right),\end{aligned} \tag{1}$$

where Z denotes noise features, c denotes the input conditional features, Φ denotes learnable convolutional layers, and zero denotes zero convolutional layer. The zero convolutional layer contains weights initialized to zero. This ensures that during the initial stages of training, the model relies more on the knowledge from the backbone part, gradually adjusting these weights as training progresses. The use of such layers aids in preserving the architecture's original behavior while introducing structure-conditioned inputs. We use the similar model architecture while we perform efficient training proposed in the main paper. We show the model architecture in Figure 1.

B Additional Implementation Details

In this section, we provide further details about the implementation aspects of our approach.

B.1 Additional Details of Structural Input Conditions Extraction

- **Edge Maps**: For generating edge maps, we utilized two distinct techniques:
  - Canny Edge Detector (Canny, 1986) - A widely used method for edge detection in images.
  - HED Boundary Extractor (Xie and Tu, 2015) - Holistically-Nested Edge Detection, an advanced technique for identifying object boundaries.

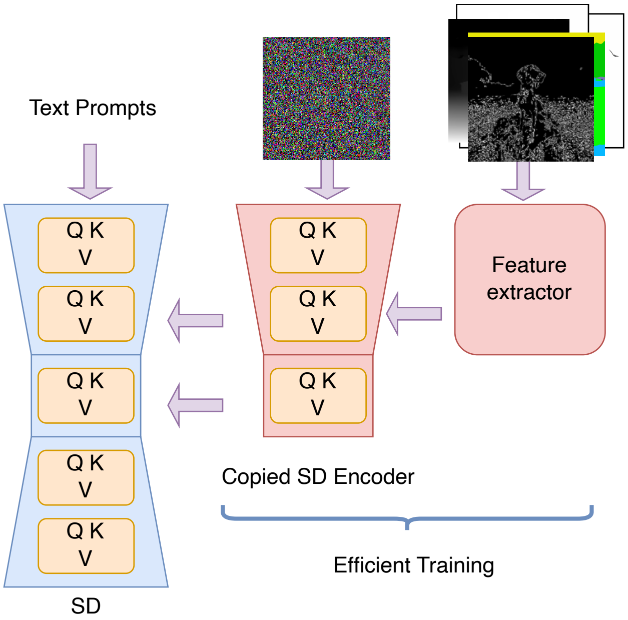


Figure 1: Detailed model architecture in FlexEControl. The Stable Diffusion part is fixed and others are trainable.

  - MLSD (Gu et al., 2022) - A method particularly designed for detecting multi-scale line segments in images.

- **Sketch Maps**: We adopted a sketch extraction technique detailed in Simo-Serra et al. (2016) to convert images into their sketch representations.
- **Pose Information**: OpenPose (Cao et al., 2017) was employed to extract human pose information from images, which provides detailed body joint and keypoint information.
- **Depth Maps**: For depth estimation, we integrated Midas (Ranftl et al., 2020), a robust method for predicting depth information from single images.
- **Segmentation Maps**: Segmentation of images was performed using the method outlined in Xiao et al. (2018), which focuses on accurately segmenting various objects within an image.

Each of these conditions plays a crucial role in guiding the text-to-image generation process, helping FlexEControl to generate images that are not only visually appealing but also semantically aligned with the given text prompts and structural conditions.

B.2 Additional Details of Evaluation Metrics

mIoU (Rezatofighi et al., 2019): Mean Intersection over Union, a metric that quantifies the degree of overlap between predicted and actual segmentation maps.

SSIM (Wang et al., 2004): Structural Similarity, a metric evaluating the structural similarity in generated outputs, applied to Canny edges, HED edges, MLSD edges, and sketches.

mAP: Mean Average Precision, utilized for pose maps, measuring the precision of localization across multiple instances.

MSE: Mean Squared Error, employed for depth maps, MSE quantifies the pixel-wise variance, providing an assessment of image fidelity.

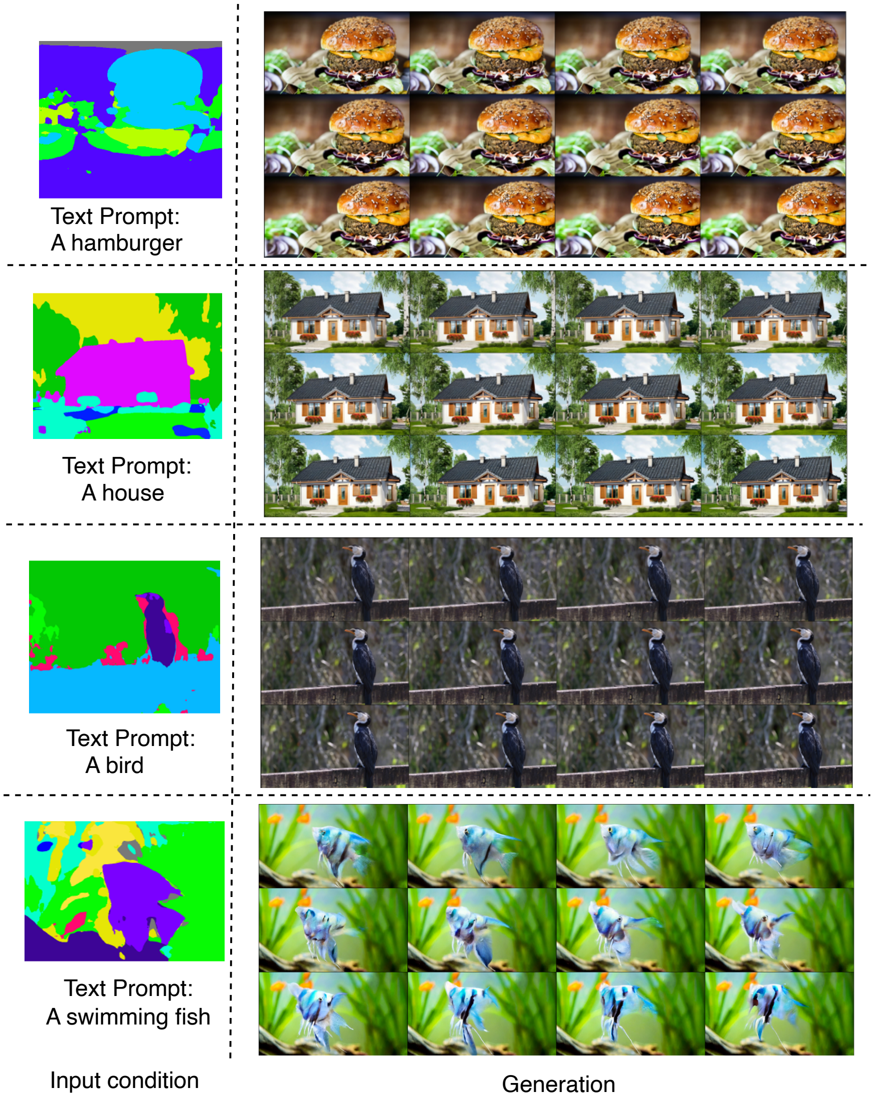


Figure 2: Results from FlexEControl on controllable text-to video generation (single condition).

FID (Heusel et al., 2017): Fréchet Inception Distance, which serves as a metric to quantify the realism and diversity of the generated images. A lower FID value indicates higher quality and diversity of the output images.

CLIP Score (Hessel et al., 2021; Radford et al., 2021): Employing CLIP Score, we gauge the semantic similarity between the generated images and the input text prompts.

C Additional Results

C.1 Additional Qualitative Results on Video Generation

FlexEControl can be further extended to accommodate video generation. In training the controllable video generation model with multiple input conditions, a straightforward strategy is employed to mask out conditions during the training process. In each iteration, a random sample, denoted as N_s, is drawn from $[1, N]$ to

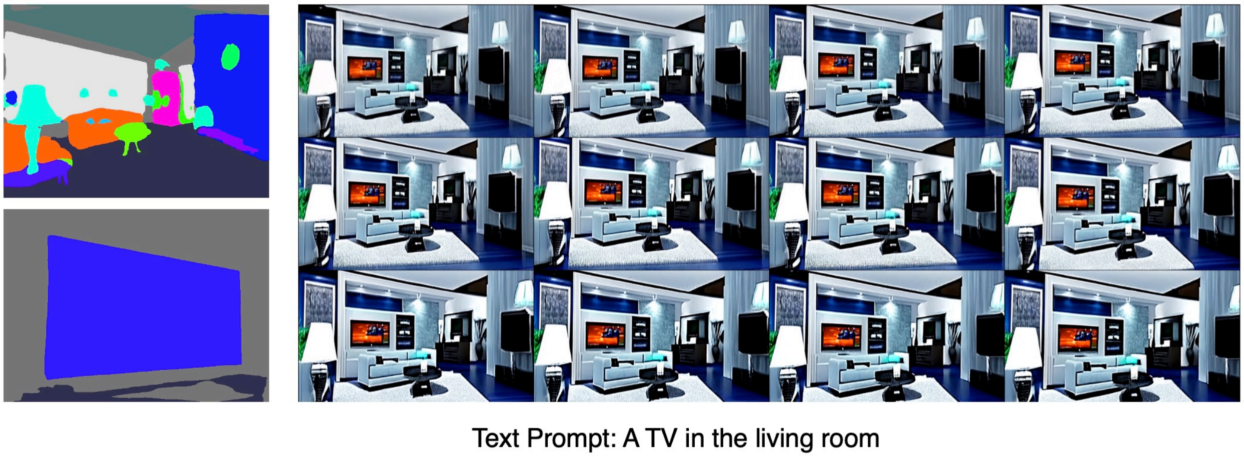


Figure 3: FlexEControl on using multiple conditions for video generation.

determine the number of frames that will incorporate the conditions. Subsequently, N_s unique values are drawn from the set $1, 2, ..., N$, and the conditions are retained for the corresponding frames.

In this section, we showcase the extensibility of FlexEControl in controllable video generation. The results are presented in Figure 2 and Figure 3, where results for providing one condition and multiple conditions are demonstrated.

D Training a Small U-Net Backbone

In this section, we discuss further methods to refine the training of a lightweight Stable Diffusion backbone within FlexEControl, aiming to further curtail the number of trainable parameters and minimize memory usage end-to-end. The resulting pre-trained Stable Diffusion backbone, which we denote as FlexEControl-pretraining, offers a more lightweight alternative to the original model while retaining versatility for application in a variety of tasks.

Building upon the strategies delineated in the main paper, we architect a streamlined U-Net structure utilizing low-rank decomposition. This design is complemented by the implementation of knowledge distillation techniques throughout the training process to cultivate an efficient text-to-image generative model. Our training regimen unfolds in two distinct phases: Initially, we focus on establishing a lightweight T2I diffusion model founded on a conventional U-Net framework, with knowledge distillation enhancing this foundational stage. Subsequently, we move to fine-tuning introduced in the main paper, enabling the model to adeptly manage controlled T2I generation tasks. This bifurcated approach yields significant resource savings both in fine-tuning and in the overall model parameter count, setting a new benchmark for efficiency in generative modeling.

D.1 Background on Low-rank Training

Background on Training in Low-dimensional Space Let $\theta^D = \left[{\theta_0}^D \ldots {\theta_m}^D\right]$ be a set of m D-dimensional parameters that parameterize the U-Net within the Stable Diffusion. Instead of optimizing the noise prediction loss in the original parameter space $\left(\theta^D\right)$, we are motivated to train the model in the lower-dimensional space $\left(\theta^d\right)$ (Aghajanyan et al., 2020). Our overall pipeline is trying to train the controllable text-to-image diffusion model in such a lower-dimension space to improve the overall efficiency.

An overview of our proposed two-stage pipeline is shown in Figure 6. We first train the U-Net of a text-to-image model with a low-rank schema. Specifically, we employ matrix factorization techniques that decompose high-dimensional matrices into smaller matrices, capturing essential features with reduced computational overhead. This process is augmented through knowledge distillation, visually represented in green on Figure 6.

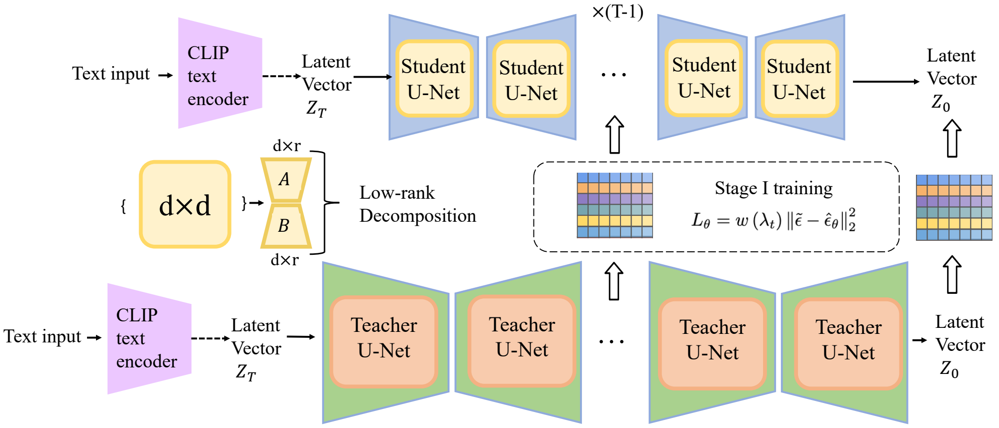


Figure 4: Overview of the Stage-1 training: Training a low-rank U-Net using knowledge distillation from a teacher model (green) to the student model (blue). This process involves initializing the student U-Net with a decomposition into low-rank matrices and minimizing the loss between the predicted noise representations from the student and teacher.

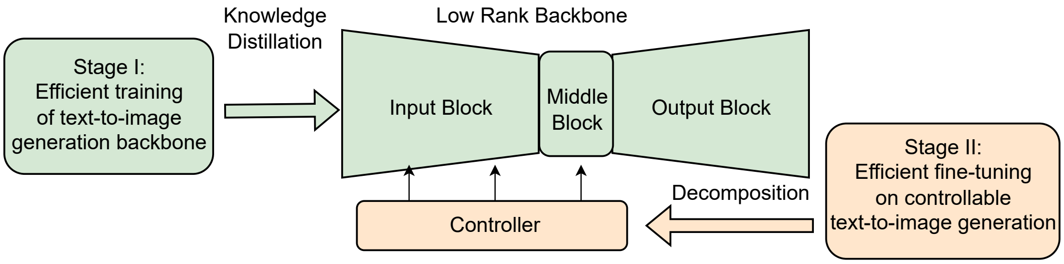

Knowledge Distillation
Low Rank Backbone
Stage I: Efficient training of text-to-image generation backbone
Input Block
Middle Block
Output Block
Controller
Decomposition
Stage II: Efficient fine-tuning on controllable text-to-image generation

Figure 5: The overview pipeline of our method. Our method improves the efficiency of controllable text-to-image generation from two aspects. At pretraining stage, we propose an efficient pretraining method for the standard text-to-image generation via knowledge distillation. For the finetuning stage introduced in the main paper, we propose to resort to low-rank and Kronecker decomposition to reduce the tunable parameter space.

We then conduct efficient fine-tuning using the methods (shown in the yellow part on Figure 6) with the methods introduced in the main paper, where we employ low-rank decomposition and Kronecker decomposition to streamline the parameter space.

Low-rank Text-to-image Diffusion Model To establish a foundational understanding of our model, it's crucial to first comprehend the role of U-Nets in the diffusion process. In diffusion models, there exists an input language prompt y that is processed by a encoder τ_θ. This encoder projects y to an intermediate representation $\tau_\theta(y) \in \mathbb{R}^{M\times d_\tau}$, where M is denotes the token length, and d_τ denotes the dimension of the embedding space . This representation is subsequently mapped to the intermediate layers of the U-Net through a cross-attention layer given by

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V, \tag{2}$$

with $Q = \mathbf{W}_Q\varphi_i(z_t)$, $\quad K = \mathbf{W}_K\tau_\theta(y)$, $\quad V = \mathbf{W}_V\tau_\theta(y)$. In this context, $\varphi_i(z_t) \in \mathbb{R}^{N\times d_\epsilon}$ is an intermediate representation of the U-Net. The terms $\mathbf{W}_V \in \mathbb{R}^{d\times d_\epsilon}, \mathbf{W}_Q \in \mathbb{R}^{d\times d_\tau}, \mathbf{W}_K \in \mathbb{R}^{d\times d_\tau}$ represent learnable projection matrices.

Shifting focus to the diffusion process, during the t-timestep, we can represent:

$$K = \mathbf{W}_K \tau_\theta(y) = AB\tau_\theta(y), \tag{3}$$

$$V = \mathbf{W}_V \tau_\theta(y) = AB\tau_\theta(y), \tag{4}$$

where A and B are decomposed low-rank matrices from the cross-attnetion matrices, d_τ and d_ϵ denote the dimension for the text encoder and noise space respectively. Conventionally, the diffusion model is trained via minimizing $\mathcal{L}_\theta = \left\|\epsilon - \epsilon_\theta\right\|_2^2$, where ϵ is the groundtruth noise and ϵ_θ is the predicted noise from the model.

Central to our strategy is a knowledge distillation process. This involves guiding a novice or 'Student' diffusion model using feature maps that draw upon the wisdom of a more seasoned 'Teacher' model. A pivotal insight from our study lies in the mathematical congruence between the low-rank training processes across both training phases, unveiling the symmetries in low-rank training trajectories across both phases.

To fully exploit the prior knowledge from the pretrained teacher model while exploiting less data and training a lightweight diffusion model, we propose a new two-stage training schema. The first one is the initialization strategy to inherit the knowledge from the teacher model. Another is the knowledge distillation strategy. The overall pipeline is shown in Figure 4.

D.2 Initialization

Directly initializing the student U-Net is not feasible due to the inconsistent matrix dimension across the Student and teacher U-Net. We circumvent this by decomposing U-Net into two low-rank matrices, namely A and B for the reconstruction. We adopt an additional transformation to adapt the teacher's U-Net weights to the Student, which leverages the Singular Value Decomposition (SVD) built upon the teacher U-Net. The initialization process can be expressed as:

1. Compute the SVD of the teacher U-Net: Starting with the teacher U-Net parameterized by θ_0, we compute its SVD as $\theta_0 = U\Sigma V^T$.

2. Extract Low-Rank Components: to achieve a low-rank approximation, we extract the first k columns of U, the first k rows and columns of Σ, and the first k rows of V^T. This results in matrices U_k, Σ_k, and V_k^T as follows:

$$U_k = \text{first } k \text{ columns of } U, \tag{5}$$

$$\Sigma_k = \text{first } k \text{ rows \& columns of } \Sigma, \tag{6}$$

$$V_k^T = \text{first } k \text{ rows of } V^T \tag{7}$$

3. We then initialize the student U-Net with $U_k\Sigma_k$ and V_k^T that encapsulate essential information from the teacher U-Net but in a lower-rank format.

We observe in practice that such initialization effectively retains the prior knowledge inherited from Teacher U-Net while enabling the student U-Net to be represented in a compact form thus computationally more efficient for later training.

D.3 Loss Function

We propose to train our Student U-Net with knowledge distillation (Meng et al., 2023) to mimic the behavior of a teacher U-Net. This involves minimizing the loss between the student's predicted noise representations and those of the teacher. To be specific, our training objective can be expressed as:

$$\mathcal{L}_\theta = w\left(\lambda_t\right)\left\|\tilde{\epsilon} - \hat{\epsilon}_\theta\right\|_2^2, \tag{8}$$

where $\tilde{\epsilon}$ denotes the predicted noise in the latent space of Stable Diffusion from the teacher model, $\hat{\epsilon}_\theta$ is the corresponding predicted noise from the student model, parameterized by θ, and $w\left(\lambda_t\right)$ is a weighting

Table 1: Comparing U-Net models: Original, decomposed, with and without Knowledge Distillation. FlexEControl-Pretraining showcases a promising balance between performance and efficiency. Note that compared with Stable Diffusion, FlexEControl-Pretraining is only trained on 5 million data. FlexEControl-Pretraining beats Decomposed U-Net w/o Distillation interms of FID and CLIP Score, suggesting the effectiveness of our distillation strategy in training the decomposed U-Net.

Methods	FID↓	CLIP Score↑	# Parameters ↓
Stable Diffusion	27.7	0.824	1290M
Standard U-Net w/o Distill.	66.7	0.670	1290M
Decomposed U-Net w/o Distill.	84.3	0.610	790M
FlexEControl-Pretraining	45.0	0.768	790M

Table 2: Performance and resource metrics comparison of FlexEControl with the baseline Uni-ControlNet. The FlexEControl approach with distillation shows a significant reduction in resource consumption while providing competitive image quality and outperforming in controllability metrics, especially in segmentation maps. The Δ column shows the improvement of FlexEControl (w/o distillation) compared with no distillation.

	Metrics	Uni-ControlNet	FlexEControl		Δ
			w/o Distill.	w/ Distill.	
Efficiency	Memory Cost ↓	20GB	**11GB**	**11GB**	**0**
	# Params. ↓	1271M	**536M**	**536M**	**0**
Image Quality	FID ↓	27.7	84.0	43.7	**- 40.3**
	CLIP Score ↑	0.82	0.61	0.77	**+0.16**
Controllability	Sketch Maps (CLIP Score)↑	0.49	0.40	0.46	**+0.06**
	Edge Maps (NMSE) ↓	0.60	0.54	0.57	**+0.03**
	Segmentation Maps (IoU) ↑	0.70	0.40	0.74	**+0.34**

function that may vary with the time step t. Such an objective encourages the model to minimize the squared Euclidean distance between the teacher and Student's predictions thus providing informative guidance to the Student. We also tried combining the loss with the text-to-image Diffusion loss but using our training objective works better.

D.4 Experimental Settings

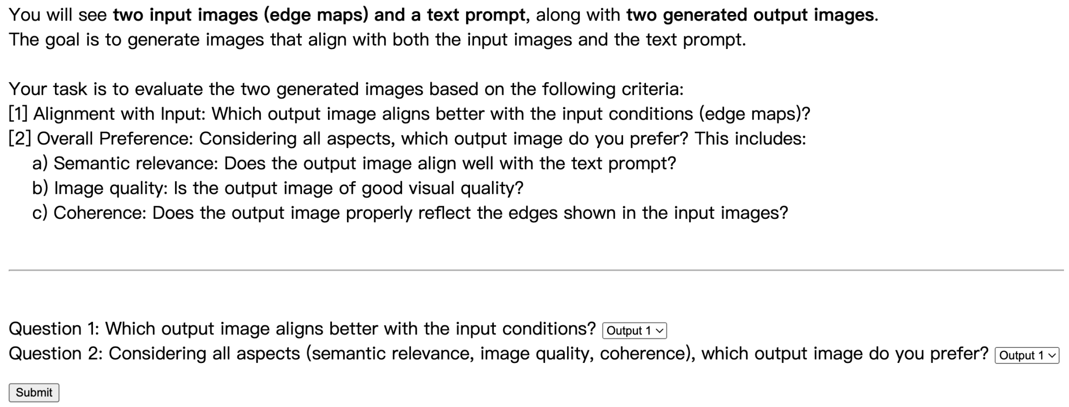
In the pretraining stage, we used the standard training scheme of Stable Diffusion (Rombach et al., 2022) with the classifier-free guidance (Ho and Salimans, 2022). We employed the Stable Diffusion 2.1 [1] model in conjunction with xFormers (Lefaudeux et al., 2022) and FlashAttention (Dao et al., 2022) using the implementation available in HuggingFace Diffusers [2].

D.5 Results

Table 1 illustrates the comparison between different variations of our method in the pretraining stage, including original U-Net, decomposed low-rank U-Net, and their respective performance with and without knowledge distillation. It is observed that the decomposed low-rank U-Net models demonstrate efficiency gains, with a reduction in the total number of parameters to 790M, although at the cost of some fidelity in metrics such as FID and CLIP Score. Employing distillation helps to mitigate some of these performance reductions.

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

[2] `https://huggingface.co/docs/diffusers/index`



You will see **two input images (edge maps) and a text prompt**, along with **two generated output images**.
The goal is to generate images that align with both the input images and the text prompt.

Your task is to evaluate the two generated images based on the following criteria:
[1] Alignment with Input: Which output image aligns better with the input conditions (edge maps)?
[2] Overall Preference: Considering all aspects, which output image do you prefer? This includes:
a) Semantic relevance: Does the output image align well with the text prompt?
b) Image quality: Is the output image of good visual quality?
c) Coherence: Does the output image properly reflect the edges shown in the input images?

Question 1: Which output image aligns better with the input conditions?
Question 2: Considering all aspects (semantic relevance, image quality, coherence), which output image do you prefer?

Figure 6: Screenshot for human evaluation tasks on the Amazon Mechanical Turk crowdsource evaluation platform.

Table 2 illustrates the comparison between FlexEControl including pretraining and the baseline training end-to-end. It is observed that the decomposed low-rank U-Net models demonstrate efficiency gains, with a reduction in the total number of parameters to 536M, although at the cost of some fidelity in metrics such as FID and CLIP Score. Employing distillation helps to mitigate some of these performance reductions.

These collective results affirm our method's capability to not only enhance efficiency but also improve or maintain performance across various aspects of text-to-image generation.

E Additional Related Works

Knowledge Distillation for Vision-and-Language Models Knowledge distillation (Gou et al., 2021), as detailed in prior research, offers a promising approach for enhancing the performance of a more streamlined "student" model by transferring knowledge from a more complex "teacher" model (Hinton et al., 2015; Sanh et al., 2019; Hu et al.; Gu et al., 2021; Li et al., 2021). The crux of this methodology lies in aligning the predictions of the student model with those of the teacher model. While a significant portion of existing knowledge distillation techniques leans towards employing pretrained teacher models (Tolstikhin et al., 2021), there has been a growing interest in online distillation methodologies (Wang and Jordan, 2021). In online distillation (Guo et al., 2020), multiple models are trained simultaneously, with their ensemble serving as the teacher. Our approach is reminiscent of online self-distillation, where a temporal and resolution ensemble of the student model operates as the teacher. This concept finds parallels in other domains, having been examined in semi-supervised learning (Peters et al., 2017), label noise learning (Bengio et al., 2010), and quite recently in contrastive learning (Chen et al., 2020). Our work on distillation for pretrained text-to-image generative diffusion models distinguishes our method from these preceding works. (Salimans and Ho, 2022; Meng et al., 2023) propose distillation strategies for diffusion models but they aim at improving inference speed. Our work instead aims to distill the intricate knowledge of teacher models into the student counterparts, ensuring both the improvements over training efficiency and quality retention.

F Human Evaluation Interface

We give the human evaluation interface in Figure 6. The human evaluators are mainly asked to finish two tasks and choose their preference from three perspectives.

References

Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. 2020. Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning. *arXiv:2012.13255 [cs]*.

Samy Bengio, Jason Weston, and David Grangier. 2010. Label embedding trees for large multi-class tasks. In *NIPS*.

John Canny. 1986. A computational approach to edge detection. *IEEE Transactions on pattern analysis and machine intelligence*, (6):679–698.

Zhe Cao, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2017. Realtime multi-person 2d pose estimation using part affinity fields. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7291–7299.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. FlashAttention: Fast and memory-efficient exact attention with IO-awareness. In *Advances in Neural Information Processing Systems*.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. 2021. Knowledge distillation: A survey. *International Journal of Computer Vision*, 129:1789–1819.

Geonmo Gu, Byungsoo Ko, SeoungHyun Go, Sung-Hyun Lee, Jingeun Lee, and Minchul Shin. 2022. Towards light-weight and real-time line segment detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 726–734.

Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. 2021. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*.

Qiushan Guo, Xinjiang Wang, Yichao Wu, Zhipeng Yu, Ding Liang, Xiaolin Hu, and Ping Luo. 2020. Online knowledge distillation via collaborative learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11020–11029.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: a reference-free evaluation metric for image captioning. In *EMNLP*.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30.

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

Jonathan Ho and Tim Salimans. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.

Xinting Hu, Kaihua Tang, Chunyan Miao, Xian-Sheng Hua, and Hanwang Zhang. Distilling Causal Effect of Data in Class-Incremental Learning. page 10.

Benjamin Lefaudeux, Francisco Massa, Diana Liskovich, Wenhan Xiong, Vittorio Caggiano, Sean Naren, Min Xu, Jieru Hu, Marta Tintore, Susan Zhang, Patrick Labatut, and Daniel Haziza. 2022. xformers: A modular and hackable transformer modelling library. `https://github.com/facebookresearch/xformers`.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. *Advances in neural information processing systems*, 34:9694–9705.

Chenlin Meng, Robin Rombach, Ruiqi Gao, Diederik Kingma, Stefano Ermon, Jonathan Ho, and Tim Salimans. 2023. On distillation of guided diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14297–14306.

Matthew Peters, Waleed Ammar, Chandra Bhagavatula, and Russell Power. 2017. Semi-supervised sequence tagging with bidirectional language models. In *ACL*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*.

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. 2020. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637.

Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. 2019. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 658–666.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Tim Salimans and Jonathan Ho. 2022. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. In *EMC2 @ NeurIPS*.

Edgar Simo-Serra, Satoshi Iizuka, Kazuma Sasaki, and Hiroshi Ishikawa. 2016. Learning to simplify: fully convolutional networks for rough sketch cleanup. *ACM Transactions on Graphics (TOG)*, 35(4):1–11.

Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. 2021. MLP-Mixer: An all-MLP Architecture for Vision. *arXiv:2105.01601 [cs]*.

Yixin Wang and Michael I. Jordan. 2021. Desiderata for Representation Learning: A Causal Perspective. *arXiv:2109.03795 [cs, stat]*.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612.

Tete Xiao, Yingcheng Liu, Bolei Zhou, Yuning Jiang, and Jian Sun. 2018. Unified perceptual parsing for scene understanding. In *Proceedings of the European conference on computer vision (ECCV)*, pages 418–434.

Saining Xie and Zhuowen Tu. 2015. Holistically-nested edge detection. In *Proceedings of the IEEE international conference on computer vision*, pages 1395–1403.

Lvmin Zhang and Maneesh Agrawala. 2023. Adding conditional control to text-to-image diffusion models. *arXiv preprint arXiv:2302.05543*.

Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K Wong. 2023. Uni-controlnet: All-in-one control to text-to-image diffusion models. *arXiv preprint arXiv:2305.16322*.