A4A: Adapter for Adapter Transfer via All-for-All Mapping for Cross-Architecture Models

Keyu Tu, Mengqi Huang, Zhuowei Chen, Zhendong Mao*
University of Science and Technology of China
{kytu, huangmq, chenzw01}@mail.ustc.edu.cn, {zdmao}@ustc.edu.cn

Abstract

*Large-scale text-to-image models evolve rapidly in size and architecture. The existing adapters struggle to keep pace with these models, requiring extensive retraining. This paper proposes a novel adapter transfer framework, **A4A** (Adapter for Adapter), which uses an all-for-all mapping approach to seamlessly transfer attention-based adapters across different model architectures (*e.g.*, U-Net to transformer). The framework consists of Coupling Space Projection and Upgraded Space Mapping. During Coupling Space Projection, all attention features of the pretrained adapter are aggregated to fully capture the coupling relationship before being projected into a unified space. The unified space maintains coupling features in a consistent dimension, effectively and efficiently addressing feature scale discrepancies arising from the base model's architecture. In the Upgraded Space Mapping Module, randomly initialized learnable features are introduced to connect the unified and upgraded spaces by integrating reference features via the attention mechanism. The learned features are adaptively injected into the upgrade model through the Alignment module, which bridges the discrepancies between the models using the all-for-all mapping. Experimental results on personalized image generation tasks demonstrate that A4A outperforms previous methods in transferring adapters while being the first to achieve adapter transfer across model architectures.*

1. Introduction

Recent advancements in large-scale text-to-image diffusion models [5, 11, 23, 26] have significantly improved their ability to generate high-quality, realistic images based on user-friendly textual prompts. Building on these generative capabilities, numerous *adapters* have been developed upon these pretrained models to further endow them with new control conditions, such as pose and human identity control, thereby fostering the growth of downstream real-world applications like personalized image creation. As pretrained models rapidly evolve, with increasing parameters (*e.g.*, from SD1.5's 860M to SDXL's 2600M) and developing architectures (*e.g.*, from the convolution U-Net [27] to the transformer [22]), the original adapters built on *base models* require substantial resources for retraining and significant effort for redesign to accommodate upgraded models. This leads to a lag in adapter development compared to the progression of *upgraded models* [1]. Therefore, the *adapter transfer* task, *i.e.*, effectively and efficiently transferring existing adapters from base models to upgraded models to leverage the strong control capabilities of the original well-developed adapters and the superior generative abilities of the upgraded models, has become an increasingly important and urgent requirement in both academia and industry.

Given the substantial potential benefits of adapter transfer, several prior studies have been conducted in this field. For instance, Ctrl-Adapter [19] has been proposed to transfer the ControlNet [41] architecture by fusing the output of the zero-convolution from the pretrained ControlNet into the corresponding layer in the upgraded model. Concurrently, X-Adapter [25] has been explored for mapping the latent from the base model's decoder block and adding them to the corresponding location within the upgraded model's decoder. In summary, existing adapter transfer methods primarily focus on addition-based adapters (*i.e.*, the control conditions are injected into the pretrained models by simple addition, typically, ControlNet), through a layer-by-layer mapping, *i.e.*, the output of each layer in the base model's adapters are mapped to the semantically equivalent layer in the upgraded model.

However, in this study, we argue that the existing layer-by-layer mapping fails to fully exploit the *coupling between original adapters and base models* to effectively bridge the

*Zhendong Mao is the corresponding author.

[1] In this study, we define "base models" as the original pretrained text-to-image models with well-developed adapters (e.g., SD1.5), and "upgraded models" as those pretrained models that require adapter transfer (e.g., SDXL). Upgraded models typically have more parameters and advanced architectural designs compared to the base models.

discrepancies between base models and upgraded models. Here, coupling refers to the already well-trained compatibility between the original adapters and base models, while differences refer to the architectural discrepancies between base models and upgraded models. The reason behind this is that the original adapters and base models function as an integrated whole, and the layer-by-layer mapping disrupts this holistic consistency by isolating the output of each layer. Consequently, during the subsequent mapping to the upgraded models, this overall consistency cannot be effectively utilized. As a result, existing adapter transfer methods suffer from **limited transfer scopes**. On the one hand, they can only transfer addition-based adapters (typically, ControlNet) but fail to generalize to the attention-based adapters, which are more widely used. This is technically more challenging because attention-based adapters (*i.e*. the control conditions are injected into pretrained models by attention mechanisms) involve more complex interactions and dependencies across different layers of the model. On the other hand, these methods are limited to transferring between similar pretrained model architectures (*e.g*., from a U-Net model to another U-Net model) but fail when transferring from a U-Net model to a transformer model. This limitation is particularly critical, as the latest state-of-the-art pretrained text-to-image models [1, 6] are predominantly transformer-based, while the current most mature adapters remain developed on the U-Net architecture.

To address this challenge, we propose a novel adapter transfer framework, A4A (Adapter for Adapter), which utilizes an innovative *all-for-all* mapping approach to seamlessly transfer the intrinsic coupling between all layers of the original adapters and base models to all layers of the upgraded models, thereby enabling the transfer of more difficult and widely used attention-based adapters and facilitating cross-architecture model transfers. Specifically, A4A achieves all-for-all mapping through the Coupling Space Projection and Upgraded Space Mapping. In the Coupling Space Projection phase, all attention features of the pretrained adapter are collected, capturing the complete coupling relationship between the adapter and the base model, and then projected into a unified space. The coupling relationship treats the features of all layers of the adapter as a unified whole, conveying a continuous representation of the new control conditions throughout the generation process, distinguishing it from isolated layer mappings. Upgraded space refers to the coupled feature space that corresponds to the upgraded model, where we randomly initialize learnable attention features to transfer the coupling relationship from the base model to the upgraded model. By integrating the reference features through the attention mechanism and aligning them with the upgraded architecture, the learnable features bridge the discrepancies between the models.

The contributions of this work are as follows:

1. To alleviate the limited transfer scopes, we introduce a novel all-for-all mapping approach that enables the transfer of attention-based adapters and facilitates cross-architecture model transfers.
2. A4A projects the adapter's complete features into the unified coupling space and bridges it with the upgraded space by fusing these features with randomly initialized learnable features through the attention mechanism.
3. Experiments in various types of personalized image creation demonstrate that A4A is an effective attention-based adapter transfer approach for cross-architecture models, achieving better performance than the pretrained adapter from the upgraded model with minimal training costs.

2. Related work

2.1. Latent Diffusion Models

Recent diffusion-based text-to-image models [11, 23, 26, 29] have received wide acclaim for their outstanding image fidelity and diversity. Ho *et al*. [11] introduce the denoising diffusion process into generation models in the seminal work DDPM. Diffusion models learn the generation process through iterative denoising steps. Latent Diffusion Model (LDM) [26] proposed to perform the diffusion process in the latent space of a Variational Autoencoder [31]. Under the LDM architecture, two primary backbone models are employed: the convolutional U-Net [27] and the transformer [22]. While these models share a similar generation process, there are significant differences between the two models. The most notable example of U-Net models is the StableDiffusion [23, 26] series, excluding SDv3.0 [6], which consists of the symmetric encoder and decoder. The encoder is composed of multiple blocks of diverse dimensions, interconnected by down-sampling layers. Each block incorporates several attention layers [32] to fuse latents and conditions. Another series of models [4, 6, 9, 28] adopts DiT blocks [22] for denoising. The transformer model has been widely used in recent video generation tasks [20], where it has achieved state-of-the-art results. After patchification, the resulting features are progressively processed through a series of DiT blocks. In the DiT blocks, the features are scaled and shifted using AdaLN, maintaining the same dimension, which distinguishes them from U-Net-based models. Transformer-based models, with their flexible structure and impressive generative capabilities, have garnered significant attention due to their great potential.

2.2. Adapters for Text-to-Image Diffusion Models

Given the inefficiencies of fine-tuning large pretrained models, an alternative strategy is to utilize adapters, which introduce a limited number of trainable parameters while keeping the pretrained model frozen. Due to their flexibility and greater efficiency compared to fine-tuning, adapters have gained significant interest. By designing conditional modules, an adapter can introduce new control conditions, such as personalized characters, objects, layouts, and style information, to a pretrained text-to-image model. Downstream tasks for these adapters include personalized character generation (ID customization) [7, 34, 40, 42, 44], personalized object generation (IP customization) [2, 7, 13, 21, 30, 36, 37, 40, 42], attribute and layout control [3, 16, 18, 35, 38, 43], and stylization [10, 39]. As discussed in Sec. 1, these adapters can be categorized into two main types: attention-based and addition-based. In the addition-based adapter (typically, ControlNet [41]), the encoded new conditions are directly added to the output of the sub-blocks of the generation model. In contrast, the attention-based adapter processes the encoded conditions through attention layers, modifying the original attention values based on both the new conditions and the text prompts. Attention-based adapters [7, 10, 30, 34, 36–40, 42–44] have received widespread attention due to their efficient and precise condition control capabilities, dominating the field of adapters.

2.3. Adapter Transferring

The rapid evolution and diverse architectures of text-to-image generation models place constraints on transferring the aforementioned adapters. Consequently, to accommodate new models, adapters often require retraining from scratch on these pretrained diffusion models. X-Adapter [25] designates models equipped with the well-trained adapter as base models, with the upgraded version referred to as the upgraded model. It establishes manual connections between decoder blocks of the same dimensions in both the base and upgraded models. Specifically, the decoder of SD1.5 and SDXL consists of blocks with dimensions 1280, 640, and 320. X-Adapter [25] maps the output of the base model's blocks to the corresponding blocks of identical dimensions in the upgraded model. As a result, X-Adapter is specifically suited for SD-series models and faces challenges when transferring to transformer-based models [4, 6, 9, 22, 28]. Ctrl-Adapter [19] aims to transfer the addition-based adapter, ControlNet [41], for video generation models or upgraded image generation models. It connects the output of the zero-convolutional layers of both models through mappers to transfer the ControlNet.

3. Method

We propose a novel framework for transferring well-trained adapters from the base model to the upgraded model with architectural discrepancies, specifically U-Net model and transformer model. For the sake of brevity, we define **PTA** as the **P**re-**T**rained **A**dapter. Specifically, *pretrained* refers to the version that has been officially published. Additionally, we denote M_{base} and M_{up} as the previously mentioned base model and upgraded model. A4A first projects the extracted attention features into the unified coupling to maintain the coupling relationship between PTAs and M_{base}. Then, the coupling space is mapped to the upgraded space, where learnable features integrate the reference coupling features with attention layers. These learned features are then adaptively aligned with M_{up} through the Alignment component.

3.1. Preliminaries

Before presenting our method, we introduce the diffusion model with various backbones. The Latent Diffusion Model (LDM), which perform noise addition and denoising in the latent space z of the VAE encoder, is the most prominent open-source community for text-to-image generation. The objective of training LDMs is:

$$\min_{\theta} \mathcal{L}_{LDM} = \mathbb{E}_{z,\epsilon\sim\mathcal{N}(0,I),t} ||\epsilon - \epsilon_\theta(z_t, t, E_t(y_t))||_2^2, \quad (1)$$

where t is uniformly sampled from the time steps $\{1, \cdots, T\}$, y_t denotes the conditional text prompt, and E_t represents the text encoder. The parameterized denoising network denoted as ϵ_θ, may take the form of either a U-Net model or a transformer model. The U-Net architecture consists of blocks with varying dimensions. Each U-Net block includes down-sampling or up-sampling layers along with attention layers. The transformer model primarily consists of multiple DiT blocks grounded on the transformer architecture. Since our method aims to transfer the widely used attention-based adapters, we define the cross-attention process and its associated signals as follows:

$$\boldsymbol{q} = \boldsymbol{W}^q \cdot \boldsymbol{i}, \quad \boldsymbol{k} = \boldsymbol{W}^k \cdot \boldsymbol{c}, \quad and \quad \boldsymbol{v} = \boldsymbol{W}^v \cdot \boldsymbol{c} \quad (2)$$

where $\boldsymbol{i}$ denotes the latents of the image, and $\boldsymbol{c}$ signifies the embeddings of the condition, such as the text prompt in the original model. Additionally, $\boldsymbol{W}$ represents the weights for attention projection. Attention is conducted as follows:

$$\text{Attention}(\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v}) = \text{Softmax}(\frac{\boldsymbol{q} \cdot \boldsymbol{k}^T}{\sqrt{d}})\boldsymbol{v} \quad (3)$$

where d represents the dimensions of $\boldsymbol{k}$ and $\boldsymbol{v}$. Through cross-attention, the image latents and condition embeddings are comprehensively integrated.

3.2. Coupling Space Projection

Condition Encoder of PTA. For new control conditions y_n beyond the original text prompt, adapters typically incorporate a condition encoder as illustrated in Fig. 1. We denote E_n to distinguish it from the original text encoder E_t of the pretrained large-scale T2I models:

$$\boldsymbol{c}_n = E_n(y_n), \quad (4)$$

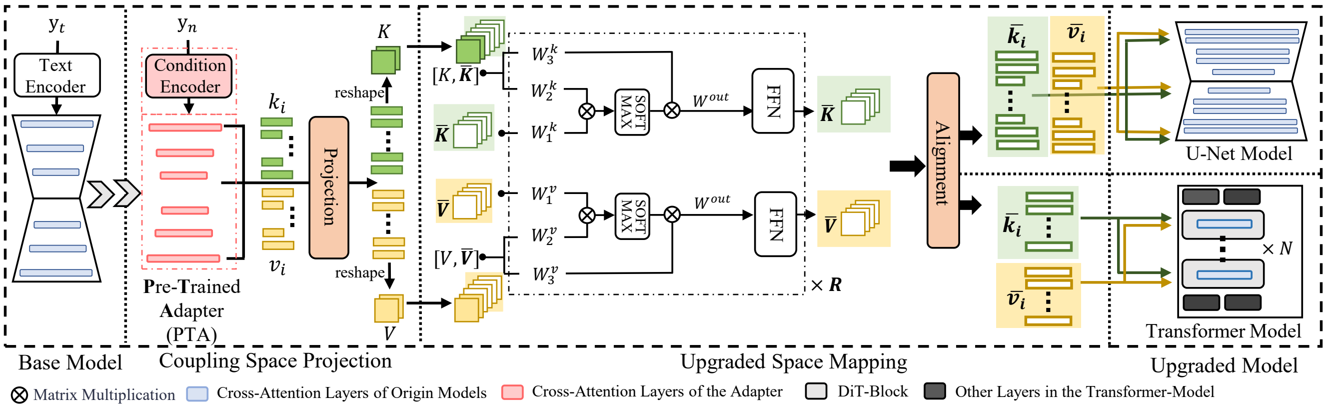

Figure 1. The illustration of the Adapter for Adapter (A4A) framework. Both the base model and the upgraded model are kept frozen. (a) Coupling Space Projection: The pretrained Adapter, consisting of the condition encoder and attention layers (highlighted in pink), is loaded. The adapter features k_i and v_i are projected into a unified coupling space, reshaping them as K and V. (b) Upgraded Space Mapping: Randomly initialized learnable upgraded features, $\bar{K}$ and $\bar{V}$, are concatenated with K and V as references. The learning process of $\bar{K}$ and $\bar{V}$ bridges the discrepancy between the base model and the upgraded model. These features are then aligned with the original cross-attention layers of the upgraded model through Alignment, which can be a U-Net or Transformer model. Best viewed in color.

where $\boldsymbol{c}_n$ denotes the new condition embeddings. Taking IP-Adapter [40] as an example, the Image Encoder external to the original generation model serves as the condition encoder. And we directly integrate the pretrained condition encoder E_n from the adapter to efficiently transfer the well-trained adapter to the upgraded model.

Attention Layers of PTA. As illustrated in Fig. 1, the encoded new condition embeddings $\boldsymbol{c}_n$ are processed through the attention mechanism coupling with the base model. We explicitly depict the attention layers associated with the pretrained adapter in the figure, denoting the weights of the i-th attention layer of the adapter as $\boldsymbol{W}_{A,i}$, distinguishing them from the original attention weights $\boldsymbol{W}$ in the base model. Subscript $_A$ represents the adapter, and $_i$ represents the i-th adapter attention layer. For instance, the fine-tuned cross-attention layers in the Decoupled Cross-Attention module of IP-Adapter [40] exemplify this. $\boldsymbol{c}_n$ are sequentially fed into the aforementioned adapter's attention layers $\boldsymbol{W}_{A,i}$:

$$\boldsymbol{k}_i = \boldsymbol{W}^k_{A,i}(\boldsymbol{c}_n), \quad \boldsymbol{k}_i \in \mathbb{R}^{N \times d_i}, \tag{5}$$

$$\boldsymbol{v}_i = \boldsymbol{W}^v_{A,i}(\boldsymbol{c}_n), \quad \boldsymbol{v}_i \in \mathbb{R}^{N \times d_i}, \tag{6}$$

where d_i is the dimension of the feature, and N denotes the number of tokens for feature $\boldsymbol{k}_i$ and $\boldsymbol{v}_i$. To extract the attention features from $\boldsymbol{c}_n$, we employ the attention layers of the adapter rather than utilizing the entire base model.

Projection onto the Coupling Space. The features obtained from multiple layers form a sequence of length l, $[\boldsymbol{k}_1, \boldsymbol{k}_2, \cdots, \boldsymbol{k}_l]$ and $[\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_l]$, where l represents the number of attention layers of PTA. The dimensions d_i of the features vary, as shown in Fig. 1. To achieve the all-for-all mapping for the attention-based adapter from M_{base} to M_{up}, the features of all cross-attention layers need to be projected into a unified coupling space S_{co} which is defined by the **s**mallest **c**ommon **m**ultiple d_{scm} of all dimensions d_i:

$$\boldsymbol{K} = \mathrm{Proj}([\boldsymbol{k}_1, \boldsymbol{k}_2, \cdots, \boldsymbol{k}_l]), \quad \boldsymbol{K} \in \mathbb{R}^{l \times N \times d_{scm}}, \tag{7}$$

$$\boldsymbol{V} = \mathrm{Proj}([\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_l]), \quad \boldsymbol{V} \in \mathbb{R}^{l \times N \times d_{scm}}. \tag{8}$$

The projection module consists of several linear layers designed to map dimensions d_i to d_{scm}. After the sequences are projected to S_{co}, they are reshaped into matrices $\boldsymbol{K}$, and $\boldsymbol{V}$. Mapping to S_{co} defined by d_{scm} strikes the best balance between efficiency and effectiveness. Sec. 7.3 in the supplementary material demonstrates this through experiments. It reduces computational complexity by aligning features to a common dimension, avoiding the overhead of larger spaces. At the same time, it maintains sufficient representational capacity, preventing the loss of important information, which can happen with smaller spaces. This ensures effective feature alignment without excessive resource usage, making it an optimal choice for both computational efficiency and model performance.

3.3. Upgraded Space Mapping

Similarly, we define the number of the cross-attention layers in the upgraded model as $\bar{l}$, and the space for upgraded models as S_{up}. To transfer the unified adapter features to the upgraded space, inspired by BLIP-2 [17], we randomly initialize two learnable parameters $\bar{\boldsymbol{K}} \in \mathbb{R}^{\bar{l} \times N \times d_{scm}}$ and $\bar{\boldsymbol{V}} \in \mathbb{R}^{\bar{l} \times N \times d_{scm}}$. Given that the attention layer is effective for integrating features, we adopt this architecture to learn the aforementioned parameters. Consider the learning of $\bar{\boldsymbol{K}}$ as an example. To enhance the robustness, we first normalize $\bar{\boldsymbol{K}}$ and $\boldsymbol{K}$ using *layer normalization*. And then, the following operation is performed:

$$\bar{\boldsymbol{K}} = \mathrm{FFN}[\boldsymbol{W}^{out} \cdot \mathrm{Attention}(\bar{\boldsymbol{K}}\boldsymbol{W}_1^k, [\boldsymbol{K}, \bar{\boldsymbol{K}}]\boldsymbol{W}_2^k, [\boldsymbol{K}, \bar{\boldsymbol{K}}]\boldsymbol{W}_3^k)]. \quad (9)$$

where $[\boldsymbol{K}, \bar{\boldsymbol{K}}]$ denotes the concatenation of $\boldsymbol{K}$ and $\bar{\boldsymbol{K}}$ along dimension N. The projection weights $\boldsymbol{W}_1^k, \boldsymbol{W}_2^k, \boldsymbol{W}_3^k \in \mathbb{R}^{d_{scm} \times d_{in}}$ map $\bar{\boldsymbol{K}}$ and $\boldsymbol{K}$, respectively, to the intermediate space with dimension d_{in}. Following the processing of features using the Attention as described in Eq. (3), they are subsequently transformed back to the original space via the $\boldsymbol{W}^{out}$. The **F**eed **F**orward **N**etwork (FFN) is composed of layers arranged sequentially, including *layer normalization, linear transformations, GELU activation, and additional linear layers*. The aforementioned process of Eq. (9) is iterated **R** times, with **R** serving as a hyperparameter. The learning process of $\bar{\boldsymbol{V}}$ is similar to $\bar{\boldsymbol{K}}$:

$$\bar{\boldsymbol{V}} = \mathrm{FFN}[\boldsymbol{W}^{out} \cdot \mathrm{Attention}(\bar{\boldsymbol{V}}\boldsymbol{W}_1^v, [\boldsymbol{V}, \bar{\boldsymbol{V}}]\boldsymbol{W}_2^v, [\boldsymbol{V}, \bar{\boldsymbol{V}}]\boldsymbol{W}_3^v)]. \quad (10)$$

We clarify that two distinct modules with an identical structure are responsible for learning $\bar{\boldsymbol{K}}$ and $\bar{\boldsymbol{V}}$, respectively. Throughout this process, the learnable features $\bar{\boldsymbol{K}}$ and $\bar{\boldsymbol{V}}$ are seamlessly integrated with the adapter features $\boldsymbol{K}$ and $\boldsymbol{V}$ of the PTA, effectively bridging the unified coupling space to the upgraded space.

Alignment with the Upgraded Model. The learned features in the upgraded space should be aligned with the attention layers of M_{up} to fully leverage the adapter's capabilities. Specifically, we first fetch the dimensions $\bar{d}_i$ of the cross-attention layers within the original upgraded model. Then, we employ the linear layer to align the i-th row of the matrix $\bar{\boldsymbol{K}}$ to $\bar{d}_i$ dimensions through Alignment, which we denote as $\bar{\boldsymbol{k}}_i$:

$$[\bar{\boldsymbol{k}}_1, \cdots, \bar{\boldsymbol{k}}_{\bar{l}}] = \mathrm{Align}(\bar{\boldsymbol{K}}). \quad (11)$$

Similarly, the identical operation is applied to the $\bar{\boldsymbol{V}}$ matrix in order to derive the vector $\bar{\boldsymbol{v}}_i$:

$$[\bar{\boldsymbol{v}}_1, \cdots, \bar{\boldsymbol{v}}_{\bar{l}}] = \mathrm{Align}(\bar{\boldsymbol{V}}). \quad (12)$$

For the i-th cross-attention layer, let $\boldsymbol{q}_i^{up}$, $\boldsymbol{k}_i^{up}$, and $\boldsymbol{v}_i^{up}$ be the original features. The extracted features, $\bar{\boldsymbol{k}}_i$ and $\bar{\boldsymbol{v}}_i$, are combined through linear weighting and summation with the prior values of the upgraded model using Attention Eq. (3):

$$\bar{\boldsymbol{Z}} = \mathrm{Attention}(\boldsymbol{q}_i^{up}, \boldsymbol{k}_i^{up}, \boldsymbol{v}_i^{up}) + \lambda \mathrm{Attention}(\boldsymbol{q}_i^{up}, \bar{\boldsymbol{k}}_i, \bar{\boldsymbol{v}}_i). \quad (13)$$

The result of Eq. (13), $\bar{\boldsymbol{Z}}$, serves as the output for the i-th cross-attention layer and will be forwarded to the next layer of the upgraded model. The parameter λ serves as a balancing factor, fixed at 1.0 during training and subsequently adjusted for downstream tasks during inference.

3.4. Optimization Loss Function

To optimize the A4A framework, we employ the loss function $\mathcal{L}_{LDM}$ of the upgraded model M_{up} as defined in Equation Eq. (1):

$$\mathcal{L}_{LDM} = \mathbb{E}_{z,\epsilon\sim\mathcal{N}(0,I),t}||\epsilon - \epsilon_\theta^{up}(z_t, t, c^{up})||_2^2, \quad (14)$$

The condition c^{up} includes the original text prompt y_t and new control conditions y_n. The upgraded model ϵ_θ^{up} acquires new control conditions and capabilities by injecting learned adapter features.

The base model and upgraded model remain frozen. The loss function is exclusively used to update the A4A module, which consists of the network with learnable features and the PTA for fine-tuning. To optimize model training, given the varying numbers of parameters in each trainable module, we adopt an asynchronous training strategy. Specifically, for training the projection in S_{co} and alignment in S_{up}, a learning rate of 1×10^{-5} is employed to avoid overfitting, while a learning rate of 1×10^{-4} is applied to the other components. If the pretrained adapter is fine-tuned, a smaller learning rate of 1×10^{-6} is used to effectively retain PTA's conditional encoding capabilities.

4. Experiments

4.1. Experimental Settings

Datasets. In our study, we utilize the CelebAMask-HQ dataset [15], which consists of approximately 20,000 high-quality facial images, each with a resolution of 1024×1024 pixels, and the OpenImages dataset [14], which offers a diverse collection of images featuring a wide range of clearly identifiable objects. We employ the BLIP-2 model [17] to generate captions for the aforementioned datasets, which serve as text prompts paired with the images. For validation, we randomly select 100 images from CelebAMask-HQ, ensuring they are distinct from those in the training set, and generate four images for each reference image.

Implementation Details. We implement A4A using SD1.5 [26] as the base model, and SDXL [23] and Pixart-Alpha (XL) [4] as the upgraded models. Both SDXL and Pixart-Alpha are significantly larger text-to-image models compared to SD1.5, and we use them as upgraded models for the U-Net and transformer architectures, respectively. In this paper, we utilize the IP-Adapter [40] as our pretrained adapter. The IP-Adapter series has recently demonstrated remarkable capabilities, garnering significant interest for its ability to enhance personalization in generative models. Its performance across a variety of tasks highlights its increasing potential to advance text-to-image generation. Due to variations in GPU types across compared methods and the absence of comprehensive GPU hour reports, we use **S**ample **C**ount (**SC**), which represents the number of samples processed up to a specific time point, as a metric

Figure 2. The visualization of personalized human generation using SDXL with U-Net architecture as the upgraded model. A4A(ours) compares with the previous work X-Adapter and pretrained adapter from the upgraded model (PTA-UM).

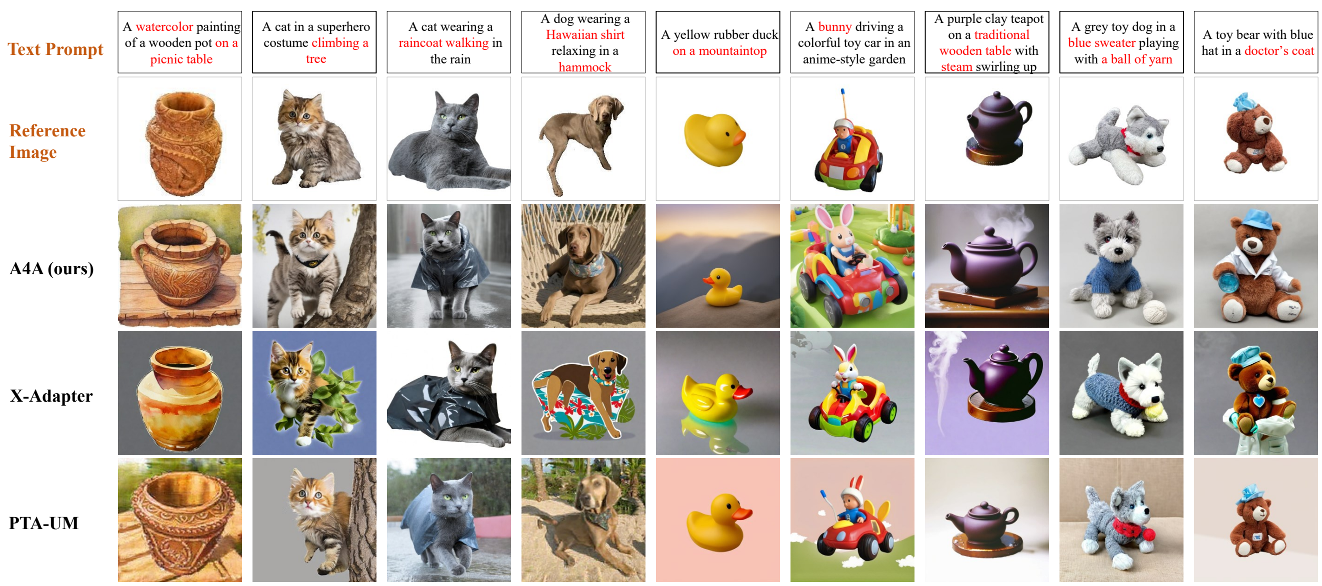

Figure 3. The visualization of personalized object generation using SDXL with U-Net architecture as the upgraded model. A4A(ours) compares with the previous work X-Adapter and pretrained adapter from the upgraded model (PTA-UM).

for evaluating training costs and efficiency. For example, the officially published PTA has an SC of 64M (*i.e.*, the PTA is trained with 8 V100 GPUs for 1M steps with a batch size of 8 per GPU). The terms in the following charts are defined as: (1) **A4A (ours)**: transferring the PTA from the base model to the upgraded model using our method A4A; (2) **X-Adapter**: transferring the PTA using the published X-Adapter [25] checkpoint; (3) **PTA-UM**: the officially published pretrained adapter from the upgraded model.

Evaluation Metrics. To verify the effectiveness and efficiency of our method, we evaluate A4A on two tasks: personalized human generation (ID customization) and personalized object generation (IP customization). For ID customization, we utilize **ID**entity **A**lignment scores (**IDA**) to measure the similarity between generated and reference facial features, alongside the OMG method [12]. Specifically, we employ the Antelopev2 model from the InsightFace library [8] to detect faces and extract facial embeddings from both reference and generated images. For IP customization, we extract image embeddings using pretrained CLIP

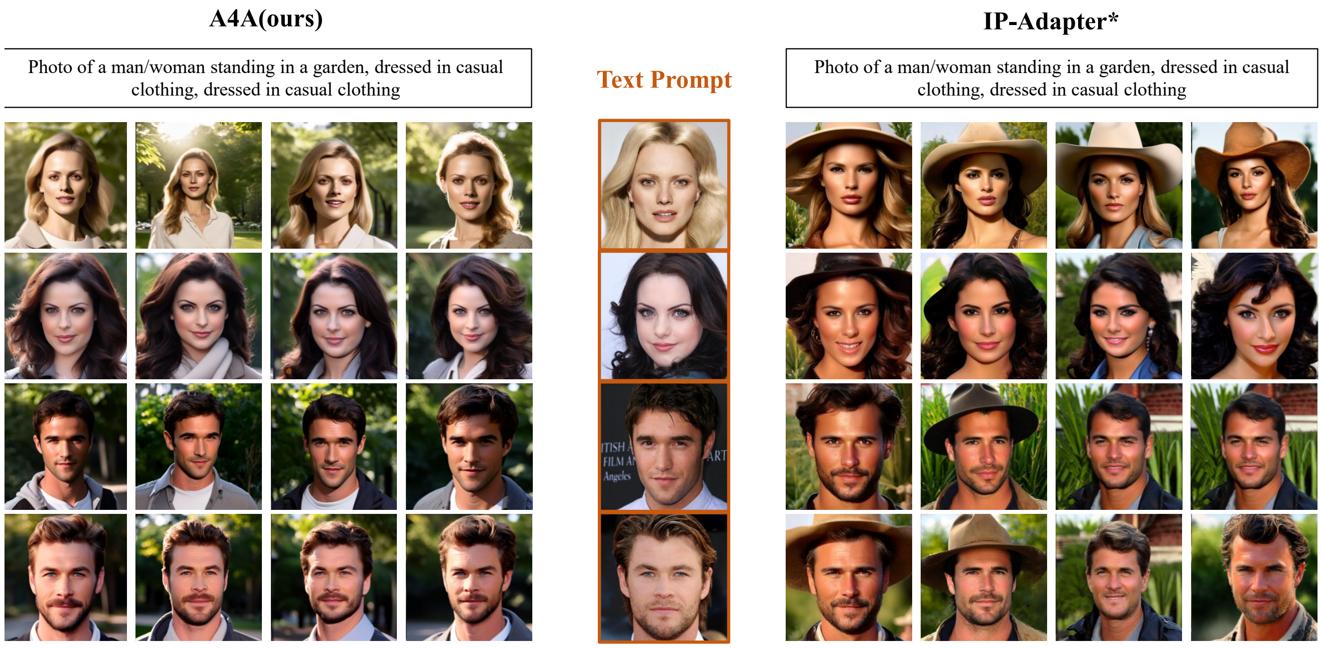

Figure 4. The visualization of transferring the IP-Adapter from SD1.5 (U-Net architecture) to Pixart-Alpha (Transformer architecture). The middle line, framed in orange, serves as the reference for comparison. The left side shows the A4A effect, which closely resembles the reference, while the right side (IP-Adapter*) displays the results of training IP-Adapter from scratch.

model [24] and report the embedding similarity between generated and reference images, referred to as the **CLIP-score**. In the graph, we use the previously defined SC as the horizontal axis to represent the training process.

4.2. Transferring to the U-Net model

We use SDXL as the upgraded model and compare our results with the officially published PTA-UM [40] (represented by the green star in Fig. 5). To facilitate comparison, we extend a horizontal line from this point to represent the performance of PTA-UM, rather than the actual training curve. As illustrated in Fig. 5, transferring the PTA using A4A achieves an IDA comparable to that of the PTA-UM at approximately SC 0.5M[2]. Furthermore, as shown in Fig. 2, the lines “A4A (ours)” and “PTA-UM” demonstrate that A4A not only preserves the intellectual property of the characters but also maintains the editing capabilities of the upgraded model with minimal training cost.

For personalized object generation (IP customization), as shown in Fig. 6, transferring the PTA using A4A achieves a CLIP-score comparable to that of PTA-UM at approximately 2.5M SC, compared to 64M SC for PTA-UM. As illustrated in Fig. 3, the lines labeled “A4A (ours)” and “PTA-UM” demonstrate that control ability, as guided by the text prompt, is also preserved. When the attention-based adapter is transferred to the upgraded U-Net model, A4A effectively retains and transfers the adapter’s capabilities with minimal training cost. The data of the quantitative indica-

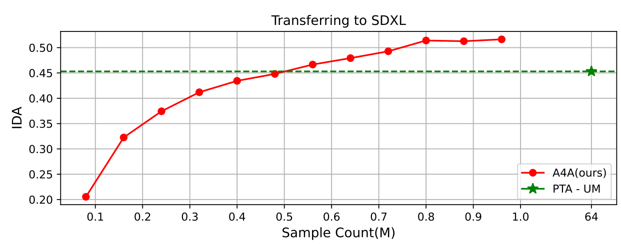

Figure 5. The graph shows the IDA of ID preservation when transferring to SDXL, compared to the pretrained adapter from SDXL (PTA-UM). The horizontal axis is in units of M.

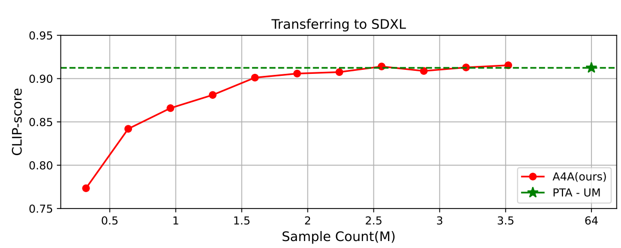

Figure 6. The graph shows the CLIP-score of personalized object generation when transferring to SDXL, compared to PTA-UM. The horizontal axis is in units of M.

[2] A4A is trained on 2 A100 GPUs with a batch size of 4 per GPU.

tors corresponding to the line chart are shown in Tab. 1.

4.3. Transferring to Transformer model

We use Pixart-Alpha [4] with the transformer architecture as the upgraded model. Given the absence of a corresponding published version of the IP-Adapter for Pixart-Alpha, we train it from scratch using the CelebAMask-HQ dataset as our baseline, represented by the orange line labeled "IP-Adapter*" in Fig. 7 and Fig. 4. As illustrated in Fig. 7, employing A4A to transfer pretrained adapters to transformer models offers a significant advantage over training adapters directly on the transformer models. It is worth noting that our work is the first to transfer the adapter from the U-Net model to the transformer model, achieving strong results. Fig. 4 presents a visualization comparing our method with training the IP-Adapter from scratch, both at 30k steps with a batch size of 8. The images generated by A4A show significant facial similarity to the reference image, while the image on the right does not yield comparable results.

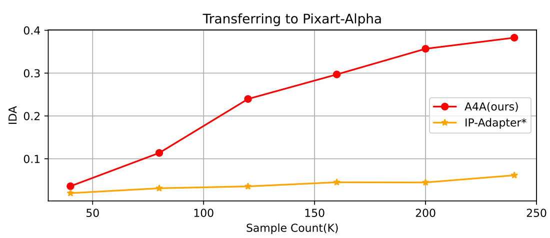

Figure 7. The IDA of transferring the pretrained IP-Adapter from SD1.5 to Pixart-Alpha using A4A (red line with dots), compared to training the IP-Adapter from scratch (orange line with stars). The horizontal axis is in units of K. Best viewed in color.

4.4. Ablation Study

In the previous section, we adopted the A4A paradigm, which includes training our modules, namely Coupling Space Projection and Upgraded Space Mapping, along with fine-tuning the PTA. To demonstrate that fine-tuning is not the core driving force of our method, we conducted the following experiment. As shown, the two curves are very close, with the fine-tuning paradigm showing only a slight improvement over the non-fine-tuning version.

We present experiments on the hyperparameter settings of the learning rates for each component in Sec. 7. It is worth mentioning that, since Projection and Alignment have similar numbers of parameters, we group them together. The ablation study of the two core modules is presented in Sec. 8, which demonstrates that our design achieves a satisfactory transfer effect with an efficient structure.

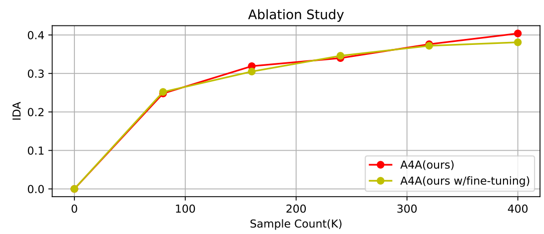

Figure 8. The yellow line labeled "A4A(ours w/fine-tuning)" represents training A4A without fine-tuning the PTA, while the red line represents the full A4A approach with fine-tuning the PTA.

PTA: IP-Adapter	IDA↑	CLIP-score↑
X-Adapter	0.062	0.7894
PTA-UM	0.4531	0.9124
A4A(ours)	<u>0.5127</u>	<u>0.9154</u>

Table 1. The evaluation metrics for IP customization (CLIP-score) and ID customization (IDA) are presented. Using SDXL as the upgraded model and IP-Adapter as the pretrained adapter, A4A (ours) is compared with the transfer method X-Adapter and the pretrained adapter from the upgraded model (PTA-UM).

4.5. Comparison with X-Adapter

The previous work X-Adapter [25] is designed specifically for transferring adapters from U-Net models. To demonstrate the effectiveness of A4A, we also compare it with X-Adapter. As shown in Tab. 1, our method achieves better results in generation using the transferred adapter for both IP and ID customization. The visualizations in Figure 2 and Fig. 3, particularly the row labeled "X-Adapter", further substantiate this when compared to the adjacent rows. It is also worth noting that our method requires only the adapter for training and inference, without the need for denoising using the base model as in X-Adapter, which makes it more efficient.

5. Conclusion

We propose A4A (Adapter for Adapter), a novel framework designed to address the challenges of transferring pretrained adapters across rapidly evolving model architectures. By employing an all-for-all mapping approach, A4A seamlessly transfers attention-based adapters from U-Net to transformer models without the need for extensive retraining. The framework's two key components, Coupling Space Projection and Upgraded Space Mapping, enable effective bridging of adapter features with upgraded model structures. Our experimental results demonstrate that A4A preserves both the generative power of upgraded models and the controllability of the original adapters. This work offers a scalable solution for cross-architecture adapter transfer.

6. Acknowledgment

This research is supported by Artificial Intelligence National Science and Technology Major Project 2023ZD0121200, and National Natural Science Foundation of China under Grant 62222212 and 623B2094.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 2

[2] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Qinsheng Zhang, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, et al. ediff-i: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 3

[3] Marco Bellagente, Manuel Brack, Hannah Teufel, Felix Friedrich, Björn Deiseroth, Constantin Eichenberg, Andrew M Dai, Robert Baldock, Souradeep Nanda, Koen Oostermeijer, et al. Multifusion: Fusing pre-trained models for multi-lingual, multi-modal image generation. *Advances in Neural Information Processing Systems*, 36:59502–59521, 2023. 3

[4] Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Yue Wu, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, et al. Pixart-alpha: Fast training of diffusion transformer for photorealistic text-to-image synthesis. *arXiv preprint arXiv:2310.00426*, 2023. 2, 3, 5, 8

[5] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 1

[6] Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024. 2, 3

[7] Rinon Gal, Moab Arar, Yuval Atzmon, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Encoder-based domain tuning for fast personalization of text-to-image models. *ACM Transactions on Graphics (TOG)*, 42(4):1–13, 2023. 3

[8] Jia Guo, Jiankang Deng, Xiang An, Jack Yu, and Baris Gecer. Insightface: 2d and 3d face analysis project, 2019. 6

[9] Thomas A Halgren, Robert B Murphy, Richard A Friesner, Hege S Beard, Leah L Frye, W Thomas Pollard, and Jay L Banks. Glide: a new approach for rapid, accurate docking and scoring. 2. enrichment factors in database screening. *Journal of medicinal chemistry*, 47(7):1750–1759, 2004. 2, 3

[10] Amir Hertz, Andrey Voynov, Shlomi Fruchter, and Daniel Cohen-Or. Style aligned image generation via shared attention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4775–4785, 2024. 3

[11] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1, 2

[12] Zhe Kong, Yong Zhang, Tianyu Yang, Tao Wang, Kaihao Zhang, Bizhu Wu, Guanying Chen, Wei Liu, and Wenhan Luo. Omg: Occlusion-friendly personalized multi-concept generation in diffusion models. *arXiv preprint arXiv:2403.10983*, 2024. 6

[13] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023. 3

[14] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *International Journal of Computer Vision*, 128(7):1956–1981, 2020. 5

[15] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo. Maskgan: Towards diverse and interactive facial image manipulation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 5

[16] Yuseung Lee and Minhyuk Sung. Reground: Improving textual and spatial grounding at no cost. *arXiv preprint arXiv:2403.13589*, 2024. 3

[17] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *International conference on machine learning*, pages 19730–19742. PMLR, 2023. 4, 5

[18] Yuheng Li, Haotian Liu, Qingyang Wu, Fangzhou Mu, Jianwei Yang, Jianfeng Gao, Chunyuan Li, and Yong Jae Lee. Gligen: Open-set grounded text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22511–22521, 2023. 3

[19] Han Lin, Jaemin Cho, Abhay Zala, and Mohit Bansal. Ctrl-adapter: An efficient and versatile framework for adapting diverse controls to any diffusion model. *arXiv preprint arXiv:2404.09967*, 2024. 1, 3

[20] Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024. 2

[21] Xichen Pan, Li Dong, Shaohan Huang, Zhiliang Peng, Wenhu Chen, and Furu Wei. Kosmos-g: Generating images in context with multimodal large language models. *arXiv preprint arXiv:2310.02992*, 2023. 3

[22] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4195–4205, 2023. 1, 2, 3

[23] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: Improving latent diffusion mod-

els for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 1, 2, 5

[24] 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 *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 7

[25] Lingmin Ran, Xiaodong Cun, Jia-Wei Liu, Rui Zhao, Song Zijie, Xintao Wang, Jussi Keppo, and Mike Zheng Shou. X-adapter: Adding universal compatibility of plugins for upgraded diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8775–8784, 2024. 1, 3, 6, 8

[26] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 1, 2, 5

[27] 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. 1, 2

[28] 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. *Advances in neural information processing systems*, 35:36479–36494, 2022. 2, 3

[29] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. 2

[30] Yoad Tewel, Rinon Gal, Gal Chechik, and Yuval Atzmon. Key-locked rank one editing for text-to-image personalization. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 3

[31] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. 2

[32] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. 2

[33] Haofan Wang, Matteo Spinelli, Qixun Wang, Xu Bai, Zekui Qin, and Anthony Chen. Instantstyle: Free lunch towards style-preserving in text-to-image generation. *arXiv preprint arXiv:2404.02733*, 2024. 3

[34] Qixun Wang, Xu Bai, Haofan Wang, Zekui Qin, Anthony Chen, Huaxia Li, Xu Tang, and Yao Hu. Instantid: Zero-shot identity-preserving generation in seconds. *arXiv preprint arXiv:2401.07519*, 2024. 3

[35] Xudong Wang, Trevor Darrell, Sai Saketh Rambhatla, Rohit Girdhar, and Ishan Misra. Instancediffusion: Instance-level control for image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6232–6242, 2024. 3

[36] Yuxiang Wei, Yabo Zhang, Zhilong Ji, Jinfeng Bai, Lei Zhang, and Wangmeng Zuo. Elite: Encoding visual concepts into textual embeddings for customized text-to-image generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15943–15953, 2023. 3

[37] Zhichao Wei, Qingkun Su, Long Qin, and Weizhi Wang. Mm-diff: High-fidelity image personalization via multi-modal condition integration. *arXiv preprint arXiv:2403.15059*, 2024. 3

[38] Yinwei Wu, Xianpan Zhou, Bing Ma, Xuefeng Su, Kai Ma, and Xinchao Wang. Ifadapter: Instance feature control for grounded text-to-image generation. *arXiv preprint arXiv:2409.08240*, 2024. 3

[39] Peng Xing, Haofan Wang, Yanpeng Sun, Qixun Wang, Xu Bai, Hao Ai, Renyuan Huang, and Zechao Li. Csgo: Content-style composition in text-to-image generation. *arXiv preprint arXiv:2408.16766*, 2024. 3

[40] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 3, 4, 5, 7, 1

[41] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 1, 3

[42] Yuxuan Zhang, Yiren Song, Jiaming Liu, Rui Wang, Jinpeng Yu, Hao Tang, Huaxia Li, Xu Tang, Yao Hu, Han Pan, et al. Ssr-encoder: Encoding selective subject representation for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8069–8078, 2024. 3

[43] Dewei Zhou, You Li, Fan Ma, Xiaoting Zhang, and Yi Yang. Migc: Multi-instance generation controller for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6818–6828, 2024. 3

[44] Zhengguang Zhou, Jing Li, Huaxia Li, Nemo Chen, and Xu Tang. Storymaker: Towards holistic consistent characters in text-to-image generation. *arXiv preprint arXiv:2409.12576*, 2024. 3