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

Supplementary Material

7. Experimental Settings

7.1. Learning Rate

The trainable network in the A4A framework is divided into three components, each with distinct learning rate hyperparameters L. These components, highlighted in Fig. 1, correspond to the following:

- L_1: Learnable features ($\bar{\boldsymbol{K}}$ and $\bar{\boldsymbol{V}}$), along with the feature-learning mechanism (depicted within the black dashed rectangle that repeats R times).
- L_2: Networks responsible for alignment and projection (represented by the orange blocks in Fig. 1).
- L_3: The PTA[3], which can be optionally fine-tuned within A4A (represented by the pink blocks).

In all the complete A4A experiments mentioned above, we set $L_1 = 1 \times 10^{-4}$, $L_2 = 1 \times 10^{-5}$, and $L_3 = 1 \times 10^{-5}$. In the following, we analyze the ablation experiments regarding the learning rate. Based on insights from previous research, L_1 selects a value between 1×10^{-4} and 1×10^{-5}. Firstly, we set L_1 to 1×10^{-4} while keeping the PTA frozen (*i.e.*, without fine-tuning). For L_2, we evaluate three values: 1×10^{-4}, 1×10^{-5}, and 1×10^{-6}. The results in Fig. 9 indicate that the A4A (without fine-tuning PTA), achieves better convergence when L_2 is set to a smaller value. Thus, L_2 selects from 1×10^{-5} and 1×10^{-6}.

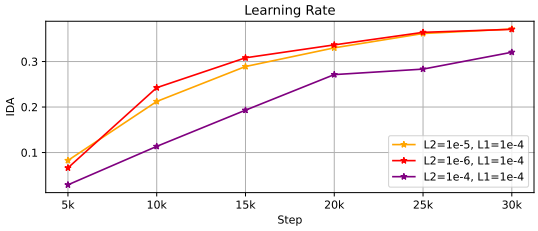

Figure 9. The performance of A4A (ours) without fine-tuning the PTA is shown with $L_1 = 1 \times 10^{-4}$.

As shown in Fig. 10, the configuration with $L_2 = 1 \times 10^{-5}$ and $L_1 = 1 \times 10^{-4}$ achieves optimal performance. For fine-tuning the PTA, which in our work corresponds to the IP-Adapter [40], the learning rate of the PTA is set to $L_3 = 1 \times 10^{-5}$ following the configuration of the IP-Adapter. The above experiments are conducted with a resolution of 1024 and a batch size of 8.

[3]pretrained Adapter from the base model

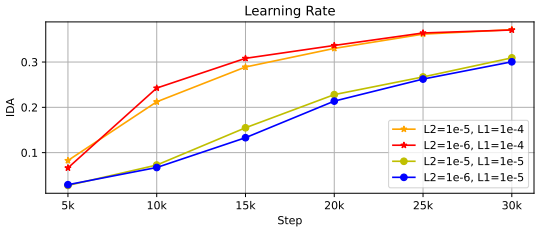

Figure 10. The performance of A4A (ours) without fine-tuning the PTA is presented with $L_1 = 1 \times 10^{-4}$ and 1×10^{-5}, as well as $L_2 = 1 \times 10^{-5}$ and 1×10^{-6}.

7.2. The Balancing Parameter λ

λ in the Eq. (13) is a hyperparameter that balances the mapped features and original features. During the training process, we set λ as 1.0 for all experiments. For inference, in the task of ID customization, the balancing parameter is also set to 1.0, consistent with the setting used for the PTA. In the task of IP customization, we report the text-image similarity (CLIP-I) and image-image similarity (CLIP-T) with different values of λ, ranging from 0.4 to 1.0 as shown in Fig. 11. For inference, we report the results with $\lambda = 1.0$ and $L_3 = 1 \times 10^{-5}$ in Tab. 1. At the point of reporting, the CLIP-I of A4A (ours) is comparable to that of the pretrained adapter from the upgraded model. The above experiments are conducted with a resolution of 1024 with a batch size of 8.

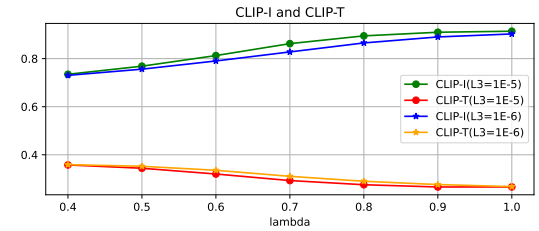

Figure 11. The performance of λ is evaluated across a range from 0.4 to 1.0. The line with points represents the setting where $L_3 = 1 \times 10^{-5}$, while the line with stars corresponds to the setting where $L_3 = 1 \times 10^{-6}$. $L_1 = 1 \times 10^{-4}$, and $L_2 = 1 \times 10^{-5}$.

7.3. Analysis of Coupling Space Dimensions

As outlined in section **Coupling Space**, the module projects adapter features into a coupling space with a dimension

Figure 12. The visualization of the training process when transferring the pretrained IP-Adapter to Pixart-α, referred to as A4A (ours), is compared with that of the IP-Adapter trained from scratch, denoted as IP-Adapter*. The left and right sides present the generated results for males and females, respectively.

equal to the **S**mallest **C**ommon **M**ultiple (d_{scm}), ensuring both efficiency and effectiveness. To validate this design, we conduct experiments with coupling space dimensions scaled to $0.5 \times scm$ and $2 \times scm$. The results, illustrated in Fig. 13, demonstrate that using $2 \times scm$ not only increases computational overhead but also reduces the adapter's transfer efficiency. Conversely, reducing the dimension to $0.5 \times scm$ compresses the features excessively, causing information loss and degraded learning performance compared to d_{scm}.

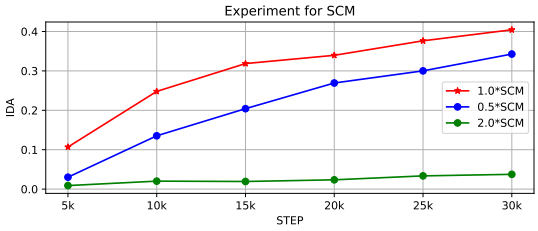

Figure 13. Results for coupling space dimensions of $0.5 \times scm$, scm, and $2 \times scm$ are presented, with all other settings consistent with those described in the Learning Rate section.

8. Ablation Experiments

8.1. The Ablation Study of Fine-tuning PTA

In Fig. 8, we present the experiments involving fine-tuning PTA or not, with SDXL as the upgraded model. We present similar experiments with Pixart-α as the upgraded model, comparing the effects of fine-tuning the PTA. As shown in Fig. 14, whether fine-tuning the PTA or not does not show a significant difference in terms of performance on IDA. Both configurations are notably better than the baseline (IP-Adapter*, the IP-Adapter trained from scratch). This indicates that the Coupling Space Projection and Upgraded Space Mapping modules we proposed are indeed beneficial for the transfer of the adapter.

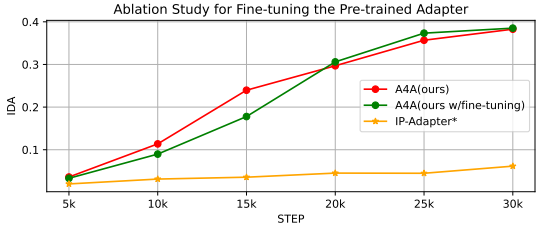

Figure 14. The ablation study of fine-tuning the PTA (IP-Adapter) is presented. The A4A without fine-tuning the PTA is labeled as *A4A(ours w/fine-tuning)*.

8.2. The Variation of Projection And Alignment

To demonstrate the necessity of each technical aspect in A4A (ours), we compare it with a naive approach. The experiments are conducted at a resolution of 512 with a batch size of 8. In A4A, we use a linear layer for Projection and Alignment to ensure dimensional compatibility between the coupling space and the upgraded space. We evaluate variations of Projection and Alignment that achieve dimensional

alignment through training-free methods, specifically interpolation for Projection and average pooling for Alignment. As shown in Fig. 15, aligning the key and value features to the coupling space aids in the adapter's transfer. Additionally, the training-based Alignment and Projection methods outperform the training-free ones.

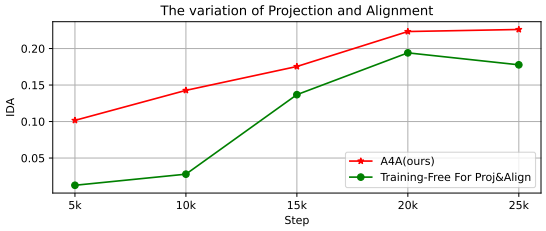

Figure 15. The results using training-free layers (*i.e.*, interpolation and average pooling for Projection and Alignment Module) are shown, labeled with green line.

8.3. Feature Learning with Linear Layers

A4A (ours) adopts Attention-driven architectures, consisting of cross-attention layers and feed-forward networks (FFN). To demonstrate the necessity of the Attention-driven method, we replace the cross-attention block with linear layers that map the original feature K and V to $\bar{\boldsymbol{K}}$ and $\bar{\boldsymbol{V}}$. The following experiment is conducted with a resolution of 512 and a batch size of 8. We conduct two experiments

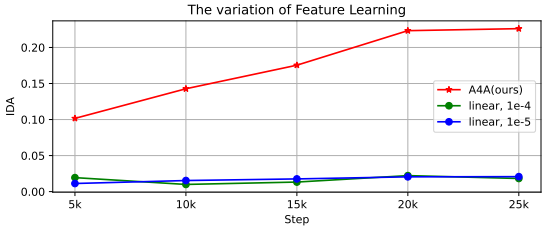

Figure 16. The results of replacing the Attention-driven feature learning mechanism with linear layers (labeled with green and blue lines) are compared to those of A4A (ours) (labeled with red).

with learning rates for the linear layer set to 1×10^{-4} and 1×10^{-5}, labeled as *linear, 1e-4* and *linear, 1e-5*, respectively. As shown in Fig. 16, the results obtained from linear feature learning are noticeably weaker compared to those from the Attention-driven blocks.

9. Visualization of Training Process

We visualize the training process in Fig. 12, where time is denoted by STEP. The experiment is conducted on two A100 GPUs with a batch size of 8, so SC=240K corresponds to STEP=30K. We show the training processes for

Figure 17. The visualization of the training process of transferring PTA to SDXL.

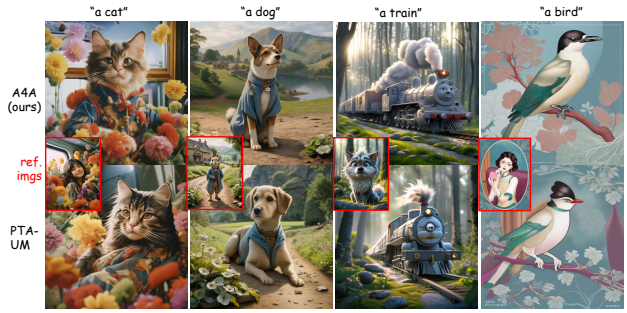

Figure 18. Visulization of transferring InstantStyle for stylization.

a woman (left) and a man (right). The image within the orange frame represents the generated reference image, while the text on either side corresponds to the text prompts. We generate images based on reference images and text prompts with 4 different random seeds. Through this visualization in Fig. 12, it is clear that the training of the IP-Adapter has significantly accelerated convergence with the assistance of A4A. We also visualize the training process of transferring to SDXL as the upgraded model. To more clearly visualize facial features, we include the phrase **close-up face** in the text prompt. As shown in Fig. 17, starting from STEP=30K, the facial similarity between the reference image and generated images shows an improvement. As training progresses, the facial feature similarity continues to improve, and the details appear more natural.

10. Style Transferring

We conduct adapter transferring on InstantStyle [33] using A4A, with settings identical to those of IP customization. Fig. 18 shows that our approach achieves comparable results with limited training cost.