

## APPENDICES

### A NETWORK DETAILS

Due to space constraints in the main paper, we only present a brief overview of the EPI process. Here, in Fig. 8, we provide a more detailed explanation of the pose transformation in EPI, along with additional case examples. First, we sample a driving pose  $I^p$  and then randomly select an anchor pose  $I_{anchor}^p$  from the pose pool (two examples are shown in Fig. 8). The driving pose  $I^p$  is aligned to the anchor pose  $I_{anchor}^p$ , resulting in the aligned pose  $I_{realign}^p$ . Next, we apply several rescaling operations randomly chosen from the rescale pool to further modify the aligned pose  $I_{realign}^p$ . By combining different rescaling options, we can obtain multiple transformed poses  $I_n^p$ . However, it is important to note that in each training step, only one anchor pose  $I_{anchor}^p$  and one rescaling combination are selected, so only one transformed pose  $I_n^p$  is used for training. As shown in the Fig. 8, the transformed pose  $I_n^p$  retains the same motion as the sampled pose  $I^p$  but has a body shape similar to the anchor pose  $I_{anchor}^p$ . This simulates scenarios during inference where there are body shape differences between the reference image and the driving pose, enabling the model to generalize to such cases.

In the experiments, we use the visual encoder of the multi-modal CLIP-Huge model [Radford et al. \(2021\)](#) in Stable Diffusion v2.1 [Rombach et al. \(2022\)](#) to encode the CLIP embedding of the reference image and driving videos. The pose encoder, composed of several convolutional layers, follows a similar structure to the STC-encoder in VideoComposer [Wang et al. \(2023c\)](#). For model initialization, we employ a pre-trained video generation model [Wang et al. \(2024c\)](#), as done in previous approaches [Xu et al. \(2023a\)](#); [Hu et al. \(2023\)](#); [Zhu et al. \(2024\)](#); [Wang et al. \(2024b\)](#). The experiments are carried out using 8 NVIDIA A100 GPUs. During training, videos are resized to a spatial resolution of 768x512 pixels, and we feed the model with uniformly sampled video segments of 32 frames to ensure temporal consistency. We use the AdamW optimizer [Loshchilov & Hutter \(2017\)](#) with learning rates of 5e-7 for the implicit pose indicator and 5e-5 for other modules. For noise sampling, DDPM [Ho et al. \(2020\)](#) with 1000 steps is applied during training. In the inference phase, we adjust the length of the driving pose to align roughly with the reference pose and used the DDIM sampler [Song et al. \(2021\)](#) with 50 steps for faster sampling.

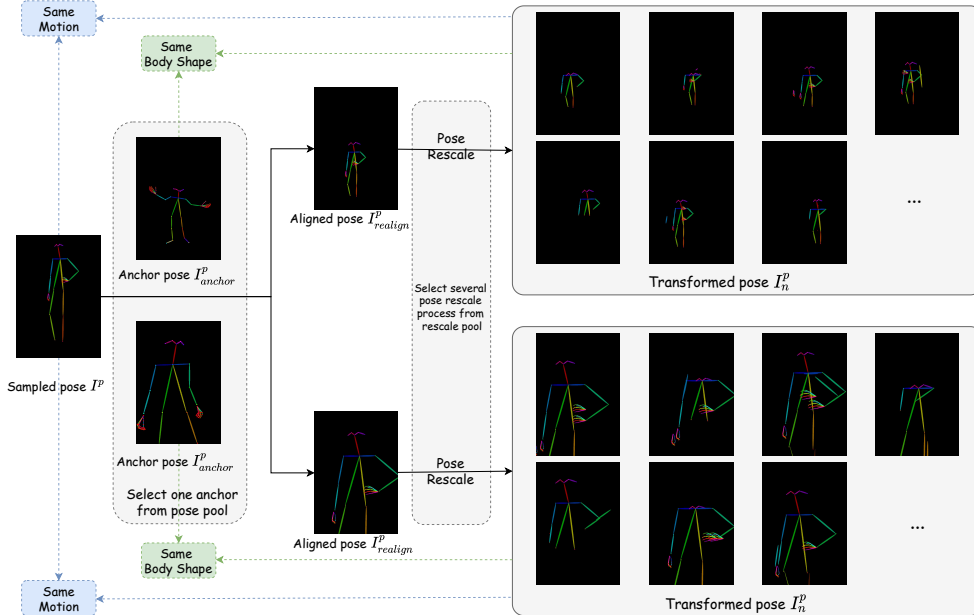


Figure 8: More example for EPI.

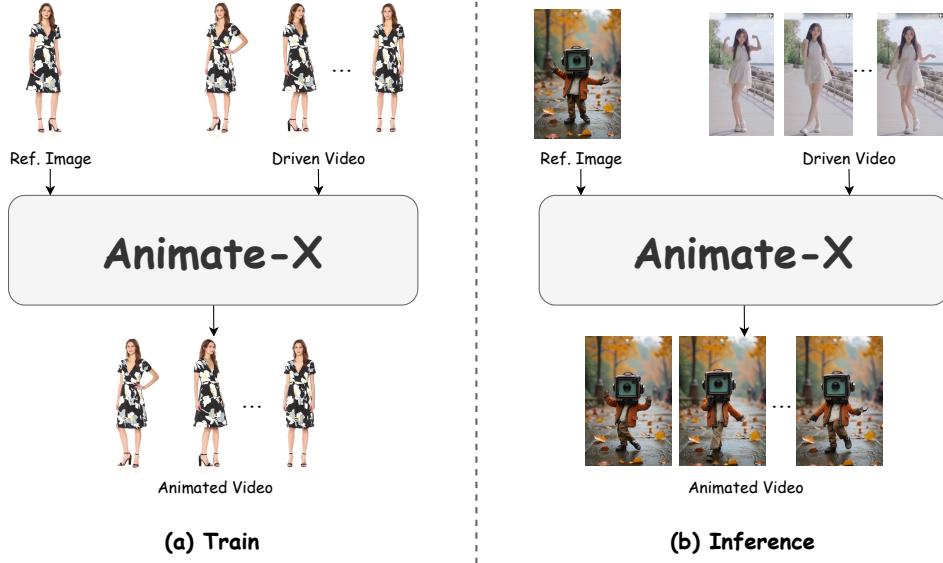


Figure 9: The difference of training and inference pipeline. During training, the reference image and the driven video come from the same video, while in the inference pipeline, the reference image and the driven video can be from any sources and appreciably different.

## B BENCHMARK DETAILS

### B.1 EVALUATION METRIC

We employ several evaluation metrics to quantitatively assess our results, including PSNR, SSIM, L1, LPIPS, FID, FID-VID and FVD. The detailed metrics are introduced as follows:

- PSNR is a measure used to evaluate the quality of reconstructed images compared to the original ones. It is expressed in decibels (dB) and higher values indicate better quality. PSNR is commonly used in image compression and restoration fields.
- SSIM assesses the similarity between two images based on their luminance, contrast, and structural information. It considers perceptual phenomena affecting human vision and thus provides a better correlation with perceived image quality than PSNR.
- The L1 metric refers to the mean absolute difference between the corresponding pixel values of two images. It quantifies the average magnitude of errors in predictions without considering their direction, making it useful for measuring the extent of differences.
- LPIPS is a perceptual distance metric based on deep learning. It evaluates the similarity between images by analyzing the feature representations of image patches and tends to align well with human visual perception, making it suitable for tasks like image generation.
- FID is used to assess the quality of images generated by generative models (like GANs) by comparing the distribution of generated images to that of real images in feature space (extracted by a pretrained CNN). Lower FID values suggest that the generated images are more similar to real images.
- FID-VID extends the FID metric to video data. It measures the quality of generated videos by comparing the distribution of generated video features to real video features, providing insights into the temporal aspects of video generation.
- FVD is another metric for evaluating video generation, similar to FID. It measures the distance between the feature distributions of real and generated videos, taking both spatial and temporal dimensions into account. Lower FVD indicates that generated videos are closer to real ones regarding visual quality and dynamics.

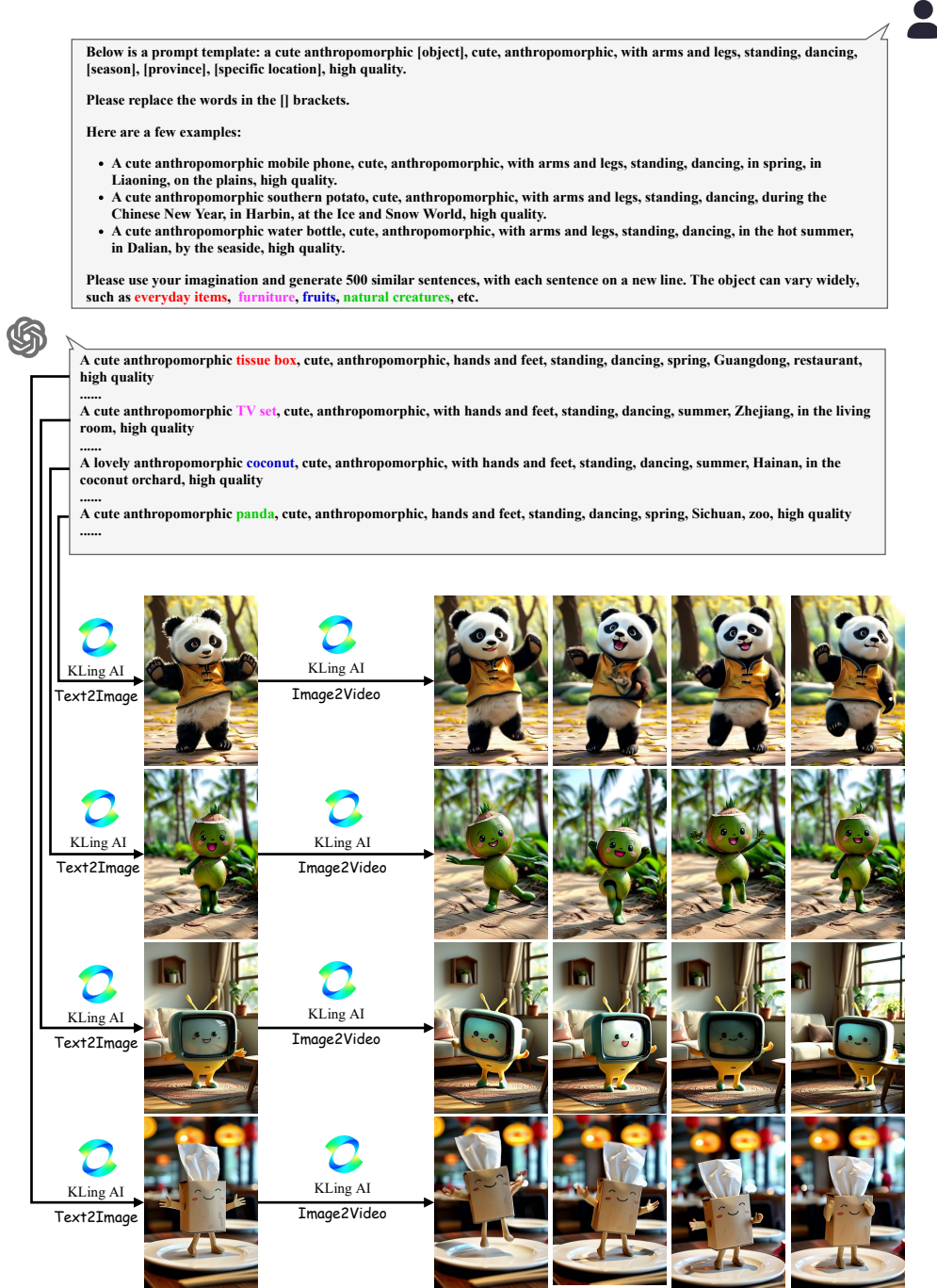
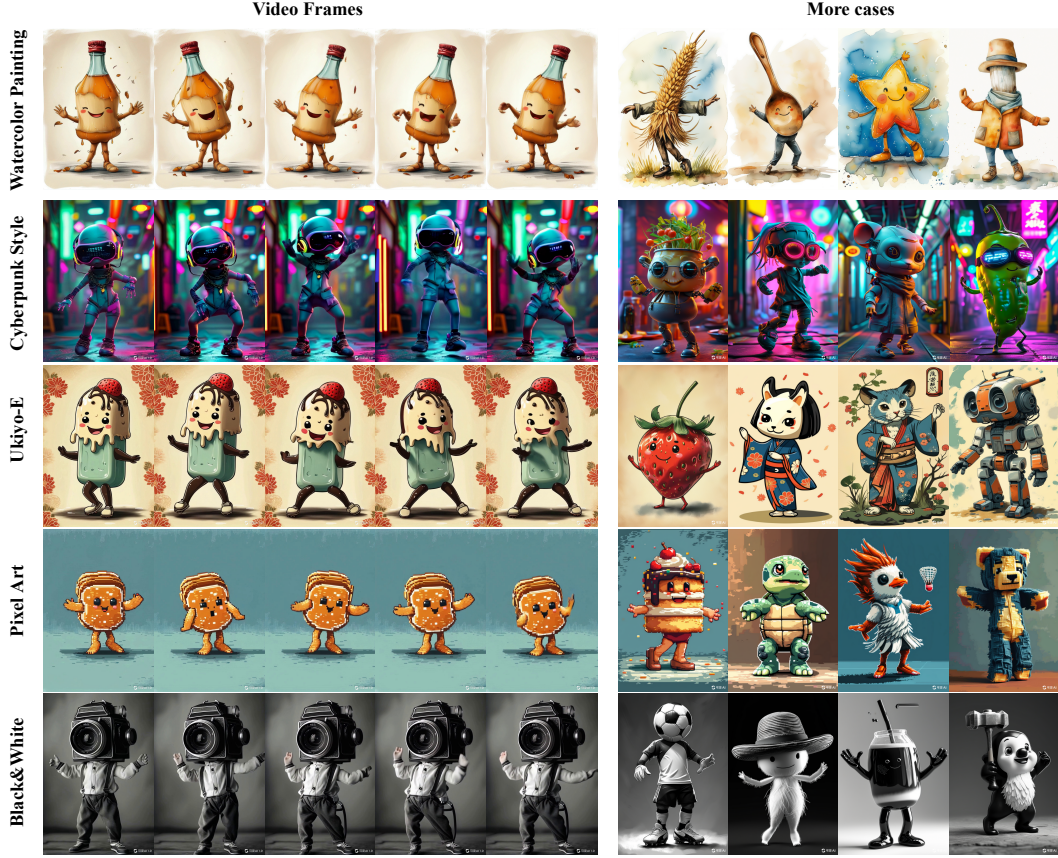


Figure 10: Detailed pipeline for building  $A^2Bench$  based on large-scale pretrained models, including Open-ChatGPT 4o and K Ling AI.

## B.2 DATA DETAILS

The detailed process for constructing  $A^2Bench$  is outlined in Fig. 10. We initially provide GPT-4o with a template that clearly specifies the demand to generate ‘anthropomorphized’ images. The images were required to be cute, with arms and legs, standing, dancing, and of high quality. To allow for a variety of image outputs, we left the fields for ‘object’, ‘season’, ‘province’, and ‘specific

Figure 11: More styles in  $A^2Bench$ .

location’ empty. For the key factor influencing diversity and relevance, i.e., ‘object’, we provide a selectable range, such as everyday items, furniture, fruits, and natural creatures. To help GPT-4o better understand our intent, we additionally provide two examples, where the prompts had already been proven to generate satisfactory images by text-to-image module of Keling AI. Thanks to the text understanding and generation capabilities of GPT-4o, we collect 500 prompts for image generation. We then fed these 500 prompts into the text-to-image module of Keling AI, obtaining corresponding anthropomorphic characters images. Based on these images, we further generate videos of them dancing using the image-to-video module of Keling AI. In this way, we collect 500 pairs of images and videos of anthropomorphic characters, forming our  $A^2Bench$ .

Moreover, we add style trigger words such as “*Watercolor Painting*”, “*Cyberpunk Style*”, “*Van Gogh*”, “*Ukiyo-E*”, “*Pixel Art*” and so on. The results are presented in Figure 11, which further enhances the diversity and complexity of  $A^2Bench$ .

Since most current animation methods Wang et al. (2024b); Hu et al. (2023); Zhang et al. (2024) take a pose image sequence as motion source, we also provide our  $A^2Bench$  with additional pose images. To achieve this, we employ DWPose Yang et al. (2023) to extract pose sequences from the videos. However, since DWPose is trained on human data, it does not accurately extract every pose in the dancing video of the anthropomorphic character, so after extraction, we manually screen 100 videos with accurate poses, and view them as test videos for calculating quantitative metrics. Fig. 3 displays several examples, which include anthropomorphic characters of plants, animals, food, furniture, etc. For images and videos where pose extraction is not feasible, we take them as key sources of reference images in our qualitative demonstrations. This will inspire the community to animate a wider range of interesting cases. We also anticipate that these data could serve as an important resource for future pose extraction algorithms tailored to anthropomorphic datasets, making them accessible for broader use.



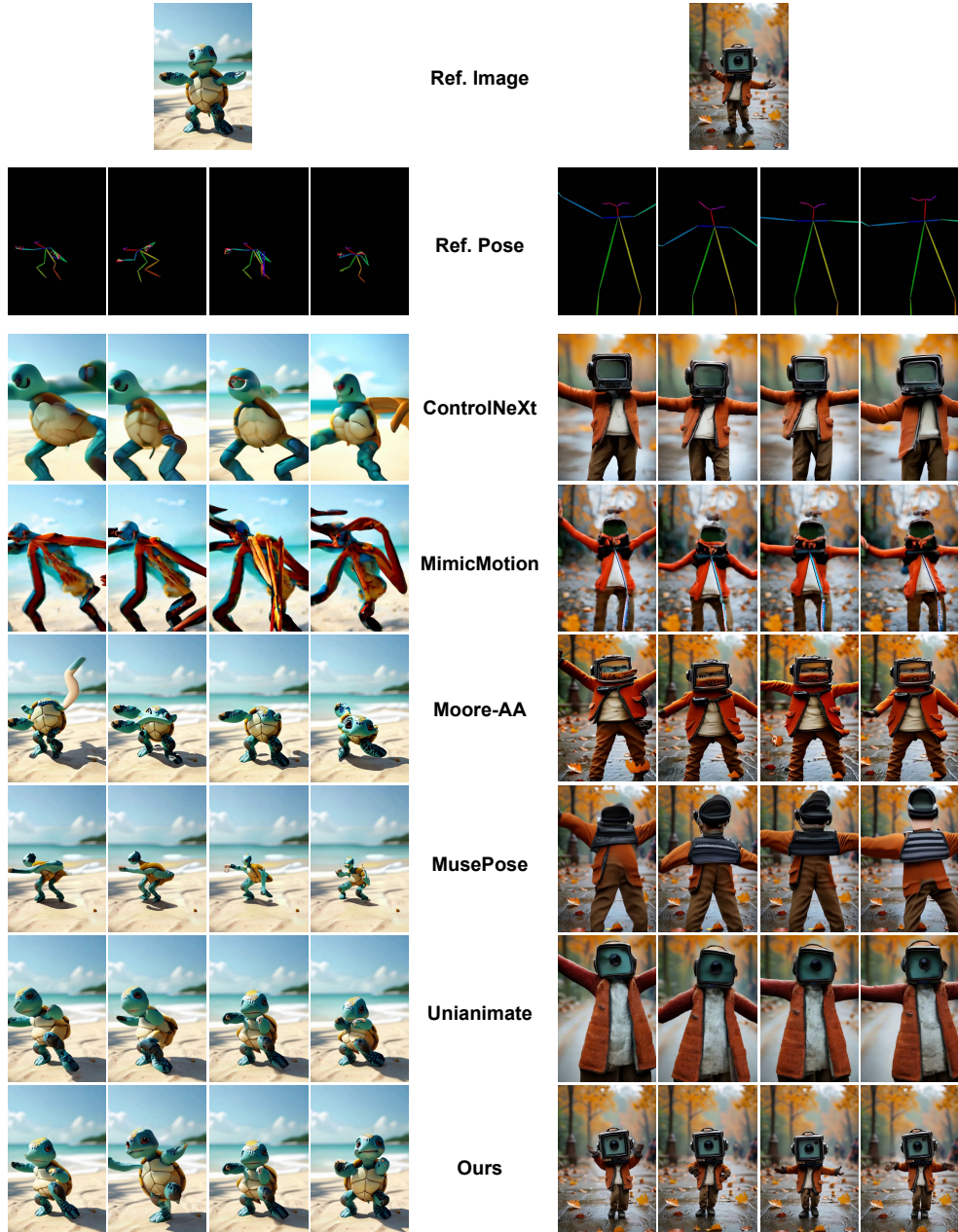


Figure 12: Visualization of cases in the user study

## C USER STUDY

In Fig. 12, we present examples shown to participants for evaluation in our user study. To obtain genuine feedback reflective of practical applications, the ten participants in our user study experiment come from diverse academic backgrounds. Since many of them do not major in computer vision, we provide detailed explanations for each question to assist their judgments.

- **Identity Preservation:** By comparing the reference image with the two generated videos by different methods, determine which video’s character more closely resembles the character in the image.

- **Temporal Consistency:** Evaluate the motion changes of the character within the video and compare which video exhibits more coherent movement.
- **Visual Quality:** Compared to the previous two questions, this one involves more subjective judgment. Participants should assess the videos comprehensively based on visual content (e.g., flashes, distortions, afterimages), motion effects (e.g., smoothness, physical logic), and overall plausibility.

## D ADDITIONAL EXPERIMENTAL RESULTS

### D.1 MORE QUALITATIVE RESULTS

In the main paper, we present qualitative comparison results between our method and the state-of-the-art (SOTA) methods under a cross-driven setting on a human-like character, where our approach demonstrates outstanding performance. Considering that the other methods are primarily self-driven and trained on human characters, making them more suitable for inference in such settings, we additionally provide comparison results under a self-reconstruction setting on Tiktok and Abench. As shown in Fig. 17, when there is a appreciably difference between the reference pose and the reference image, the GAN-based LIA Wang et al. (2022) produces noticeable artifacts. Thanks to the powerful generative capabilities of diffusion models, diffusion-based models generate higher-quality results. However, MusePose Tong et al. (2024) and MimicMotion Zhang et al. (2024) generate awkward arms and blurry hands, respectively, while ControlNeXt Peng et al. (2024) synthesizes incorrect movements. Only Unianimate Wang et al. (2024b) can obtain results comparable to ours. Yet, when the reference image is a non-human character, even in a self-driven setting with the same training strategy as Unianimate, their results still show distorted heads. Fig. 18 provides results of more comparison results, including MRAA Siarohin et al. (2021a), MagicAnimate Xu et al. (2023a) and Moore-AnimateAnyone Corporation (2024). In contrast, our method consistently generates satisfactory results for both human and anthropomorphic characters, demonstrating its ability to drive X character and highlighting its strong generalization and robustness.

### D.2 MORE QUANTITATIVE RESULTS

Tab. 10 and Tab. 11 presents the quantitative results on TikTok Jafarian & Park (2021) and Fashion Zablotskaia et al. (2019a) dataset, which suggests the superiority of methods over the comparison SOTA methods. Only Unianimate achieves comparable performance; however, our method is applicable to a wider range of characters and various unaligned pose inputs, as demonstrated in Tab. 1. This addresses the main issue that this paper aims to solve: developing a universal character image animation model.

### D.3 ROBUSTNESS

Our method demonstrates robustness to both input X character and pose variations. On the one hand, as shown in Fig. 1, our approach successfully handles inputs from diverse subjects, including characters vastly different from humans, such as those without limbs, as well as game characters or those generated by other models. Despite these variations, our method consistently produces satisfactory results without crashing, showcasing its robustness to the input reference images. On the other hand, as illustrated in Fig. 13, even when the pose images exhibit body part omissions (highlighted by the red circles), our method correctly interprets the intended motion and generates coherent results for the reference images. This highlights the robustness of our approach to different pose images.

### D.4 $A^2$ BENCH

**Difficulty Level.** We add the difficulty level split for Animate-X. As shown in Figure 14, we categorize the videos in  $A^2$ Bench into three difficulty levels: Level 1, Level 2, and Level 3. The classification is based on their appearance characteristics. **First**, we classify characters that have body shapes and other appearance features similar to humans, as shown in the first row of Figure 14, into the easiest, Level 1 category. These characters are generally simpler to drive, produce fewer

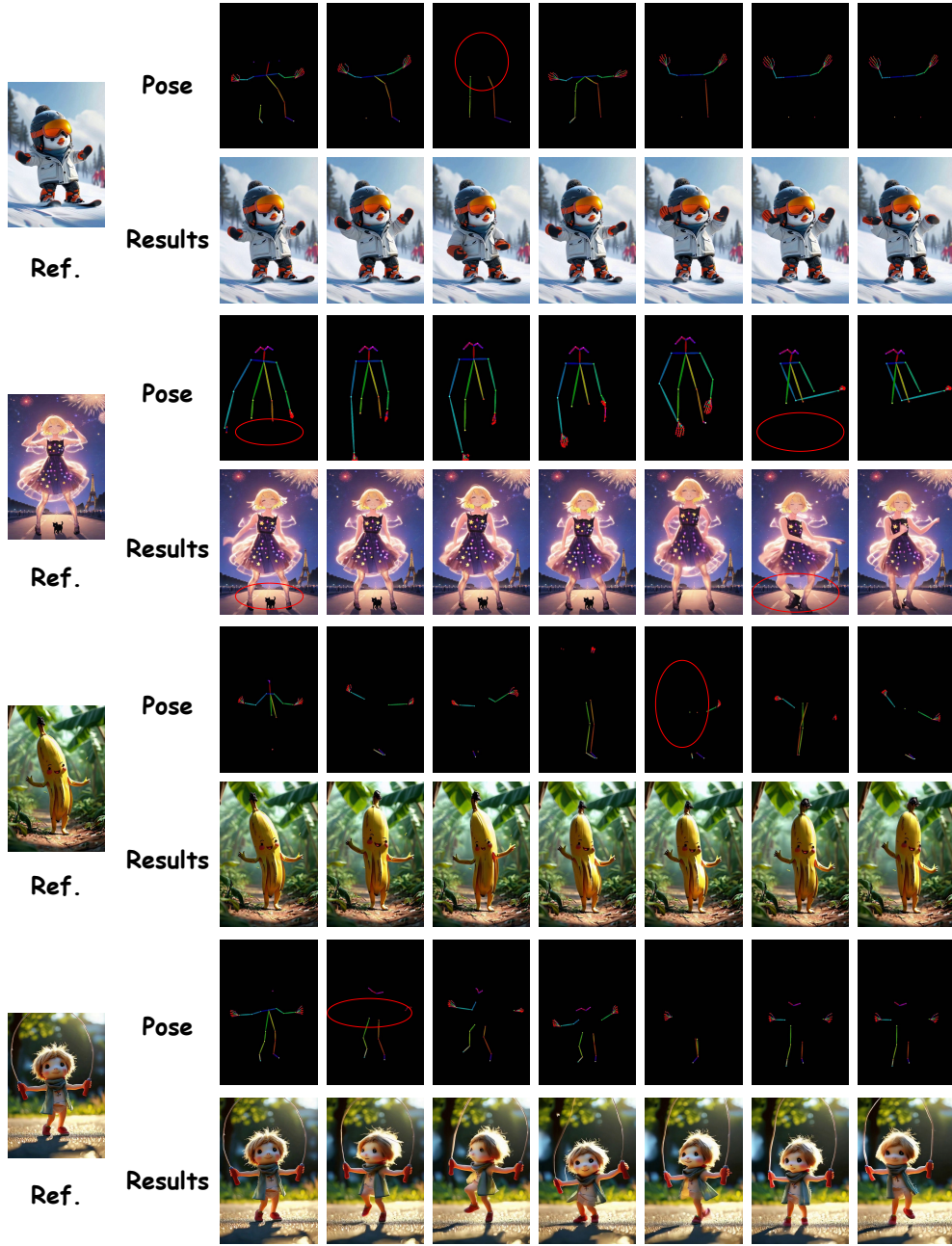


Figure 13: Visualization of the robustness of Animate-X.

artifacts, and have better motion consistency. **In contrast**, characters that maintain more distinct structural features from humans, such as dragons and ducks in the third row of Figure 14, are classified into the most difficult Level 3 category. These characters often preserve their original structures (*e.g.*, a duck’s webbed feet and wings), which makes balancing identity preservation and motion consistency more challenging. To ensure identity preservation, the consistency of motion may be compromised, and vice versa. Additionally, images involving interactions between characters, objects, environments, and backgrounds are also placed in Level 3, as they increase the difficulty for the model to distinguish the parts that need to be driven from those that do not. **Videos in between these two categories**, like those in the second row of Figure 14, are classified as Level 2. These characters often strike a good balance between anthropomorphism and their original form, making



Figure 14: Difficulty levels in  $A^2$ Bench.

Model-Level	PSNR* $\uparrow$	SSIM $\uparrow$	L1 $\downarrow$	LPIPS $\downarrow$	FID $\downarrow$	FID-VID $\downarrow$	FVD $\downarrow$
Animate-X-level1	13.96	0.461	9.67E-05	0.418	24.24	31.37	681.53
Animate-X-level2	13.74	0.457	9.82E-05	0.429	26.12	32.19	693.63
Animate-X-level3	13.17	0.442	1.11E-04	0.437	27.34	35.64	721.41
UniAnimate-level1	11.93	0.413	1.14E-04	0.521	42.39	52.14	1120.45
UniAnimate-level2	11.89	0.408	1.20E-04	0.526	46.27	58.53	1147.34
UniAnimate-level3	10.91	0.379	1.35E-04	0.549	56.58	65.39	1204.53

Table 5: User study results.

Method	PSNR* $\uparrow$	SSIM $\uparrow$	L1 $\uparrow$	LPIPS $\downarrow$	FID $\downarrow$	FID-VID $\downarrow$	FVD $\downarrow$
w/o IPI	13.30	0.433	1.35E-04	0.454	32.56	64.31	893.31
w/o LQ	<u>13.48</u>	0.445	1.76E-04	0.454	28.24	42.74	754.37
w/o DQ	13.39	0.445	<b>1.01E-04</b>	0.456	30.33	62.34	913.33
PA	13.25	0.436	1.11E-04	0.464	27.63	46.54	785.36
KV_Q	13.34	0.443	1.17E-04	0.459	26.75	42.14	785.69
w/o EPI	12.63	0.403	1.80E-04	0.509	42.17	58.17	948.25
w/o Add	13.28	0.442	1.56E-04	0.459	34.24	52.94	804.37
w/o Drop	13.36	0.441	1.94E-04	0.458	<u>26.65</u>	44.55	764.52
w/o BS	13.27	0.443	1.08E-04	0.461	29.60	56.56	850.17
w/o NF	13.41	<u>0.446</u>	1.82E-04	0.455	29.21	56.48	878.11
w/o AL	13.04	0.429	1.04E-04	0.474	27.17	<u>33.97</u>	765.69
w/o Rescalings	13.23	0.438	1.21E-04	0.464	27.64	35.95	<u>721.11</u>
w/o Realign	12.27	0.433	1.17E-04	0.434	34.60	49.33	860.25
<b>Animate-X</b>	<b>13.60</b>	<b>0.452</b>	<u>1.02E-04</u>	<b>0.430</b>	<b>26.11</b>	<b>32.23</b>	<b>703.87</b>

Table 6: Quantitative results of ablation study.

them easier to animate with better motion consistency than Level 3 characters and more interesting results than Level 1 characters. We evaluate the results of Animate-X and UniAnimate for each subset. As shown in Tab. 5, as the difficulty increases, each evaluation result shows a decline.

**Motivation of T2I+I2V for  $A^2$ Bench.** The choice to use T2I models stems from a clear need: current T2V models often struggle with imaginative and logically complex inputs, such as “*personified refrigerators*” or “*human-like bees*”. T2I models offer strict logic and imagination in these scenarios, allowing to generate reasonable cartoon characters as the ground-truth. To prove this point, as



shown in Table 7, we assess the semantic accuracy of  $A^2\text{Bench}$  using CLIP scores, which are commonly used to evaluate whether the semantic logic of images and text is strictly aligned (*i.e.*, Does the generated “human-like bee” maintain the visual essence of a bee while seamlessly incorporating human-like features, such as hands and feet?). We also add other metrics from VBench Huang et al. (2024), such as *Background Consistency*, *Motion Smoothness*, *Aesthetic Quality* and *Image Quality*, to assess the spatial and temporal consistency of the videos in  $A^2\text{Bench}$ . For comparison, we also evaluate the publicly available TikTok and Fashion datasets using the same metric. As shown in Table 7,  $A^2\text{Bench}$  achieves the highest level of strict logical alignment.  $A^2\text{Bench}$  outperforms TikTok dataset in all aspects and achieve comparable scores to Fashion dataset, where both TikTok and Fashion are collected from real-world scenarios. It demonstrates that the video generated by our method has the same level of spatial and temporal consistency as the real videos.

**Furthermore**, we input the images from  $A^2\text{Bench}$  into a multimodal large language model (MLLM) with logical reasoning, such as QWen Bai et al. (2023), to conduct a logical analysis of the visual outputs generated by the T2I model. The results, shown in Figure 15, reveal that the image descriptions answered by the MLLM closely aligns with our input prompts, which verifies again that the data in  $A^2\text{Bench}$  maintains strict logic.

Table 7: Quantitative results of different benchmarks. The best and second results for each column are **bold** and underlined, respectively.

Benchmark	CLIP Score	Background Consistency	Motion Smoothness	Aesthetic Quality	Image Quality
TikTok	<u>26.92</u>	94.10 %	99.05 %	<u>55.14 %</u>	<u>62.54 %</u>
Fashion	20.18	<b>98.25 %</b>	<b>99.45 %</b>	49.62 %	49.96 %
$A^2\text{Bench}$	<b>33.24</b>	<u>96.66 %</u>	<u>99.39 %</u>	<b>69.86 %</b>	<b>69.32 %</b>

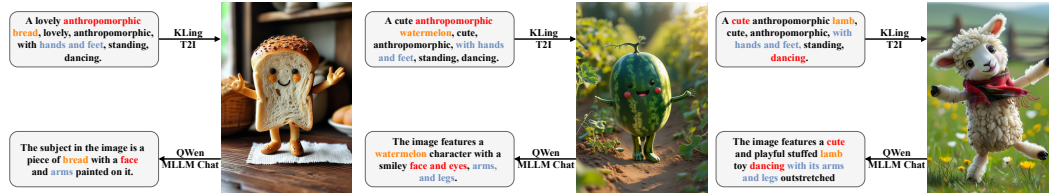


Figure 15: Prompts, generated images by T2I in  $A^2\text{Bench}$ , and logical answers from QWen.

## D.5 MORE ABLATION STUDY

In the main paper, we present the results of the primary ablation experiments for IPI and EPI. In this section, we supplement those results with additional ablation experiments to further demonstrate the contribution of each individual module.

**Ablation on Implicit Pose Indicator.** For more detailed analysis about the structure of IPI, we set up several variants: (1) remove IPI: w/o IPI. (2) remove learnable query: w/o LQ. (3) remove DWPose query: w/o DQ. (4) set IPI and spatial Attention to Parallel: PA. (5) set CLIP features as Q and DWPose as K, V in IPI: KV\_Q. The quantitative results are shown in Tab. 6. It can be seen that removing the entire IPI presents the worst performance. By modifying the IPI module, although it improves on the w/o IPI, it still falls short of the final result of Animate-X, which suggests that our current IPI structure is the most reasonable and achieves the best performance.

Since IPI is embedded in Animate-X in the form of residual connection, *i.e.*,  $x = x + \alpha IPI(x)$ , we also explore the impact of the weight  $\alpha$  of IPI on performance as illustrated in Fig. 16, as  $\alpha$  increases from 0 to 1, all metrics show a stable improvement despite some fluctuations. The best performance is achieved when  $\alpha$  is set to 1, so we empirically set  $\alpha$  to 1 in the final configuration.

**Ablation on Explicit Pose Indicator.** We conduct more detailed ablation experiments for different pairs of pose transformations by (1) removing the entire EPI: w/o EPI; (2)&(3) removing adding and dropping parts; canceling the change of the length of (4) body and should: w/o BS; (5) neck and face: w/o NF; (6) arm and leg: w/o AL; (7) removing all rescaling process: w/o Rescalings; (8)

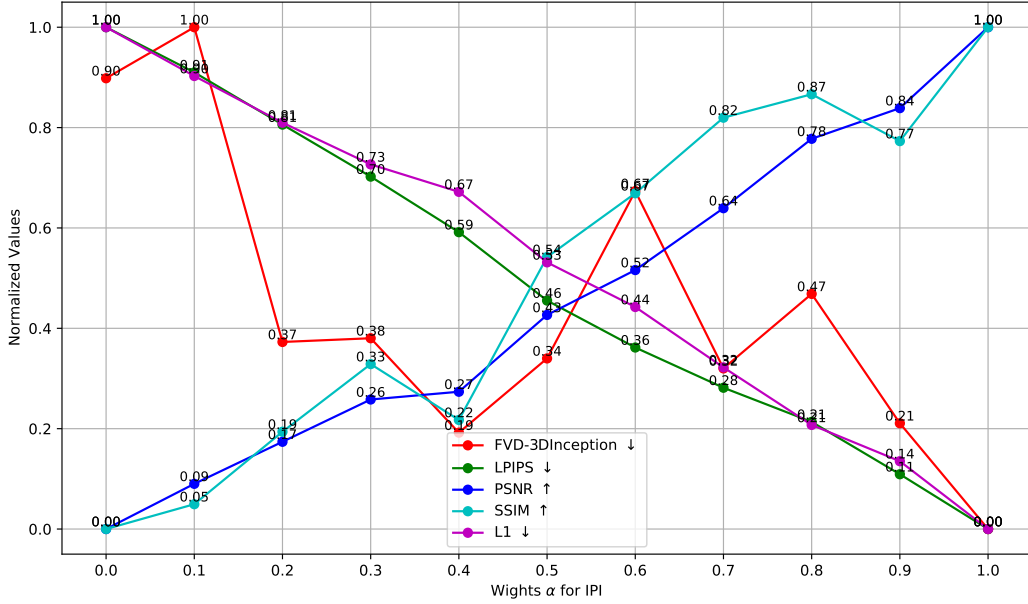


Figure 16: Ablation study on the weight  $\alpha$  of Implicit Pose Indicator. To better visualize the impact of  $\alpha$  on performance, we normalize all the values to the range of 0 to 1.

remove another person pose alignment: w/o Realign. From the results displayed in Tab. 6, we found that each pose transformation contributes compared to w/o EPI, with aligned transformations with another person’s pose contributing the most. It suggests that maintaining the overall integrity of the pose while allowing for some variations is the most important factor, and EPI also learns the overall integrity of the pose. The final result indicates that all the transformations together achieve the best performance.

To explore the effect of different probabilities  $\lambda$  of using pose transformation for EPI on the model performance, we set  $\lambda$  as 100%, 98%, 95%, 90% and 80% for the ablation experiments on two datasets. The results presented in Tab. 9 suggest that a high  $\lambda$  performs better on  $A^2$ Bench, i.e., it performs better when the reference image and pose image are not aligned, but harms performance on the TikTok dataset, i.e., when the reference image and pose image are strictly aligned. In contrast, a relatively low  $\lambda$ , e.g., 90%, would be in this case perform better. It is reasonable that in the case of strict alignment, we expect the pose to provide a strictly accurate motion source, and thus need to reduce the percentage  $\lambda$  of pose transformation. However, in the non-strictly aligned case, we expect the pose image to provide an approximate motion trend, so we need to increase  $\lambda$ .

Since the anchor poses are chosen from the entire training set, we further conduct the statistical analysis for rescaling ratio. First, we randomly sample a driven pose  $I^p$  and then traverse the entire pose pool, treating each pose in the pool as an anchor pose to calculate the rescaling ratio. We repeat this process 10 times. Finally, we divide the range from 0.001 to 10 into 10 intervals, counting the proportion of rescaling ratios that fell within each interval. We analyze the proportions of other important parts like shoulder length, body length, upper arm length, lower arm length, upper leg length, lower leg length. As shown in Tab. 8, the overall distribution covers a wide range (from 0.001 to 10.0), which allows the model to learn poses of various characters, encompassing non-human subjects.

## E DISCUSSION

### E.1 LIMITATION AND FUTURE WORK

Although our method has made remarkable progress, it still has certain limitations. Firstly, its ability to model hands and faces remains insufficient, a limitation commonly faced by most current

Interval	Shoulder Length	Body Length	Upper Arm Length	Lower Arm Length	Upper Leg Length	Lower Leg Length
[0.001, 0.1)	0.19%	0.14%	0.05%	0.08%	0.05%	0.81%
[0.1, 0.3)	1.52%	5.73%	4.04%	3.22%	0.59%	4.60%
[0.3, 0.5)	12.21%	18.57%	15.28%	7.63%	4.26%	5.65%
[0.5, 0.7)	15.33%	16.93%	12.97%	7.54%	12.02%	9.61%
[0.7, 1.0)	20.07%	18.48%	17.15%	11.35%	24.86%	19.53%
[1.0, 1.5)	22.09%	18.63%	17.56%	15.38%	27.90%	24.89%
[1.5, 2)	10.07%	8.34%	7.93%	11.73%	14.31%	14.47%
[2.0, 3.0)	9.75%	6.52%	7.73%	16.19%	11.83%	15.28%
[3.0, 6.0)	6.33%	6.28%	10.93%	18.40%	2.73%	4.30%
[6.0, 10.0)	2.43%	0.37%	6.37%	8.47%	1.45%	0.85%

Table 8: Statistical analysis for rescaling ratio.

Method	A <sup>2</sup> Bench				TikTok Jafarian & Park (2021)			
	SSIM↑	FID↓	FID-VID↓	FVD↓	SSIM↑	FID↓	FID-VID↓	FVD↓
<b>100%</b>	<b>0.452</b>	<b>26.11</b>	<b>32.23</b>	<b>703.87</b>	0.802	55.26	17.47	138.36
<b>98%</b>	<u>0.448</u>	26.93	<u>37.67</u>	<u>775.24</u>	0.797	55.81	16.28	<u>129.48</u>
<b>95%</b>	0.447	27.46	39.21	785.55	<u>0.804</u>	<b>52.72</b>	<u>14.61</u>	<b>124.92</b>
<b>90%</b>	0.444	27.15	38.03	775.38	<b>0.806</b>	<u>52.81</u>	14.82	139.01
<b>80%</b>	0.442	29.13	47.93	803.97	0.802	<u>54.51</u>	<b>14.42</b>	133.78

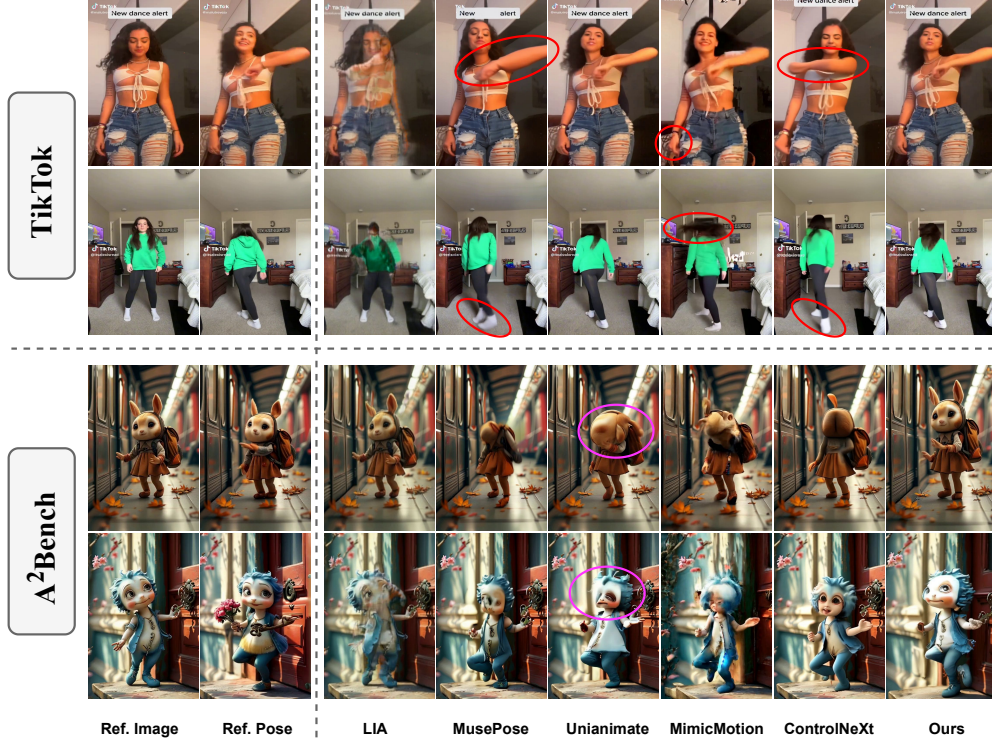
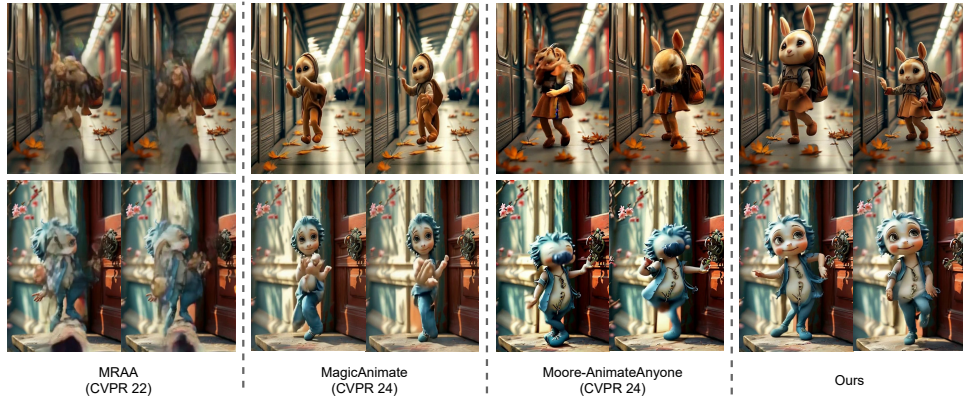
Table 9: Quantitative results for different probabilities of using pose transformation.

Method	L1 ↓	PSNR ↑	PSNR* ↑	SSIM ↑	LPIPS ↓	FVD ↓
FOMM Siarohin et al. (2019a) <small>(NeurIPS19)</small>	3.61E-04	-	17.26	0.648	0.335	405.22
MRAA Siarohin et al. (2021a) <small>(CVPR21)</small>	3.21E-04	-	18.14	0.672	0.296	284.82
TPS Zhao & Zhang (2022a) <small>(CVPR22)</small>	3.23E-04	-	18.32	0.673	0.299	306.17
DreamPose Karras et al. (2023) <small>(ICCV23)</small>	6.88E-04	28.11	12.82	0.511	0.442	551.02
DisCo Wang et al. (2024a) <small>(CVPR24)</small>	3.78E-04	29.03	16.55	0.668	0.292	292.80
MagicAnimate Xu et al. (2023a) <small>(CVPR24)</small>	3.13E-04	29.16	-	0.714	0.239	179.07
Animate Anyone Hu et al. (2023) <small>(CVPR24)</small>	-	29.56	-	0.718	0.285	171.90
Champ Zhu et al. (2024) <small>(ECCV24)</small>	2.94E-04	29.91	-	0.802	0.234	160.82
Unianimate Wang et al. (2024b) <small>(ArXiv24)</small>	<b>2.66E-04</b>	<u>30.77</u>	<u>20.58</u>	<b>0.811</b>	<b>0.231</b>	<u>148.06</u>
MusePose Tong et al. (2024) <small>(ArXiv24)</small>	3.86E-04	-	17.67	0.744	0.297	215.72
MimicMotion Zhang et al. (2024) <small>(ArXiv24)</small>	5.85E-04	-	14.44	0.601	0.414	232.95
ControlNeXt Peng et al. (2024) <small>(ArXiv24)</small>	6.20E-04	-	13.83	0.615	0.416	326.57
<b>Animate-X</b>	<u>2.70E-04</u>	<b>30.78</b>	<b>20.77</b>	<u>0.806</u>	<u>0.232</u>	<b>139.01</b>

Table 10: Quantitative comparisons with existing methods on TikTok dataset.

Method	PSNR ↑	PSNR* ↑	SSIM ↑	LPIPS ↓	FVD ↓
MRAA Siarohin et al. (2021a) <small>(CVPR21)</small>	-	-	0.749	0.212	253.6
TPS Zhao & Zhang (2022a) <small>(CVPR22)</small>	-	-	0.746	0.213	247.5
DPTN Zhang et al. (2022a) <small>(CVPR22)</small>	-	24.00	0.907	0.060	215.1
NTED Ren et al. (2022) <small>(CVPR22)</small>	-	22.03	0.890	0.073	278.9
PIDM Bhunia et al. (2023) <small>(CVPR23)</small>	-	-	0.713	0.288	1197.4
DBMM Yu et al. (2023) <small>(ICCV23)</small>	-	24.07	0.918	0.048	168.3
DreamPose Karras et al. (2023) <small>(ICCV23)</small>	-	-	0.885	0.068	238.7
DreamPose w/o Finetune Karras et al. (2023) <small>(ICCV23)</small>	34.75	-	0.879	0.111	279.6
Animate Anyone Hu et al. (2023) <small>(CVPR24)</small>	<b>38.49</b>	-	0.931	0.044	81.6
Unianimate Wang et al. (2024b) <small>(ArXiv24)</small>	<u>37.92</u>	<u>27.56</u>	<b>0.940</b>	<b>0.031</b>	<b>68.1</b>
MimicMotion Zhang et al. (2024) <small>(ArXiv24)</small>	-	27.06	0.928	0.036	118.48
<b>Animate-X</b>	36.73	<b>27.78</b>	<b>0.940</b>	<b>0.030</b>	<u>79.4</u>

Table 11: Quantitative comparisons with existing methods on the Fashion dataset. “w/o Finetune” represents the method without additional finetuning on the fashion dataset.

Figure 17: Visualization comparison on TikTok dataset and  $A^2$ Bench.Figure 18: Comparison with more SOTAs on  $A^2$ Bench.

generative models. While our **IPI** leverages CLIP features to extract implicit information such as motion patterns from the driving video, mitigating the reliance on potentially inaccurate hand and face detection by DWPose, there is still a gap between our results and the desired realism. Secondly, due to the multiple denoising steps in the diffusion process, even though we replace the transformer with a more efficient Mamba model for temporal modeling, **Animate-X** still cannot achieve real-time animation. In future work, we aim to address these two limitations. Additionally, we will focus on studying interactions between the character and the surrounding environment, such as the background, as a key task to resolve. As for  $A^2$ Bench, creating 3D models and rendering them with predefined actions using tools like Blender and Maya is a superior approach for developing a character benchmark, which is also part of our future work.



## E.2 ETHICAL CONSIDERATIONS

Our approach focuses on generating high-quality character animation videos, which can be applied in diverse fields such as gaming, virtual reality, and cinematic production. By providing body movement, our method enables animators to create more lifelike and dynamic characters. However, the potential misuse of this technology, particularly in creating misleading or harmful content on digital platforms, is a concern. While greatly progress has been made in detecting manipulated animations [Boulkenafet et al. \(2015\)](#); [Wang et al. \(2020\)](#); [Yu et al. \(2020\)](#), challenges remain in accurately identifying increasingly sophisticated forgeries. We believe that our animation results can contribute to the development of better detection techniques, ensuring the responsible use of animation technology across different domains.