OmniVCus: Feedforward Subject-driven Video Customization with Multimodal Control Conditions

Yuanhao Cai[1], **He Zhang**[2], **Xi Chen**[3,*], **Jinbo Xing**[4],
Yiwei Hu[2], **Yuqian Zhou**[2], **Kai Zhang**[2], **Zhifei Zhang**[2], **Soo Ye Kim**[2],
Tianyu Wang[2], **Yulun Zhang**[5,*], **Xiaokang Yang**[5], **Zhe Lin** [2], **Alan Yuille**[1]
[1] Johns Hopkins University, [2] Adobe Research, [3] The University of Hong Kong,
[4] The Chinese University of Hong Kong, [5] Shanghai Jiao Tong University

Abstract

Existing feedforward subject-driven video customization methods mainly study single-subject scenarios due to the difficulty of constructing multi-subject training data pairs. Another challenging problem that how to use the signals such as depth, mask, camera, and text prompts to control and edit the subject in the customized video is still less explored. In this paper, we first propose a data construction pipeline, VideoCus-Factory, to produce training data pairs for multi-subject customization from raw videos without labels and control signals such as depth-to-video and mask-to-video pairs. Based on our constructed data, we develop an Image-Video Transfer Mixed (IVTM) training with image editing data to enable instructive editing for the subject in the customized video. Then we propose a diffusion Transformer framework, OmniVCus, with two embedding mechanisms, Lottery Embedding (LE) and Temporally Aligned Embedding (TAE). LE enables inference with more subjects by using the training subjects to activate more frame embeddings. TAE encourages the generation process to extract guidance from temporally aligned control signals by assigning the same frame embeddings to the control and noise tokens. Experiments demonstrate that our method significantly surpasses state-of-the-art methods in both quantitative and qualitative evaluations. Project page is at https://caiyuanhao1998.github.io/project/OmniVCus/

1 Introduction

Text-to-video diffusion generation models [1–3] have achieved great success in creating high-quality videos from user-provided text prompts. These advancements spark increasing interest in subject-driven video customization that aims to create a video for specific identities in user-provided images.

Current subject-driven video customization methods are mainly divided into two categories: tuning-based and feedforward methods. Tuning-based solutions [4–6] are time-consuming. They fine-tune adapters [7, 8]/LoRAs [9] attached to a pre-trained video diffusion model each time for one inference. In contrast, feedforward methods [10–14] integrate the visual embeddings of subjects into diffusion models during training in a data-driven manner to enable video customization without test-time tuning. Despite progress on feedforward methods, there are still some challenges as follows:

(i) Existing methods mainly study single-subject customization due to the difficulty of constructing multi-subject data pairs. Some works have explored multi-subject data construction but still have limitations. For instance, ConceptMaster [11] constructs closed-set data pairs with limited subject categories, which degrades the model's generalization ability. **(ii)** As the subjects in each training video are always limited, how to enable inference with more subjects is important but under-explored. **(iii)** How to add control conditions of different modalities to subject-driven customization is also less studied. These conditions include textual instructions to edit the subject in the generated video, camera trajectory to move the viewpoint, segmentation mask sequence, depth map, and so on. The data-preparation pipelines of previous methods also neglect to produce control signals for the subjects.

*Corresponding Authors

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

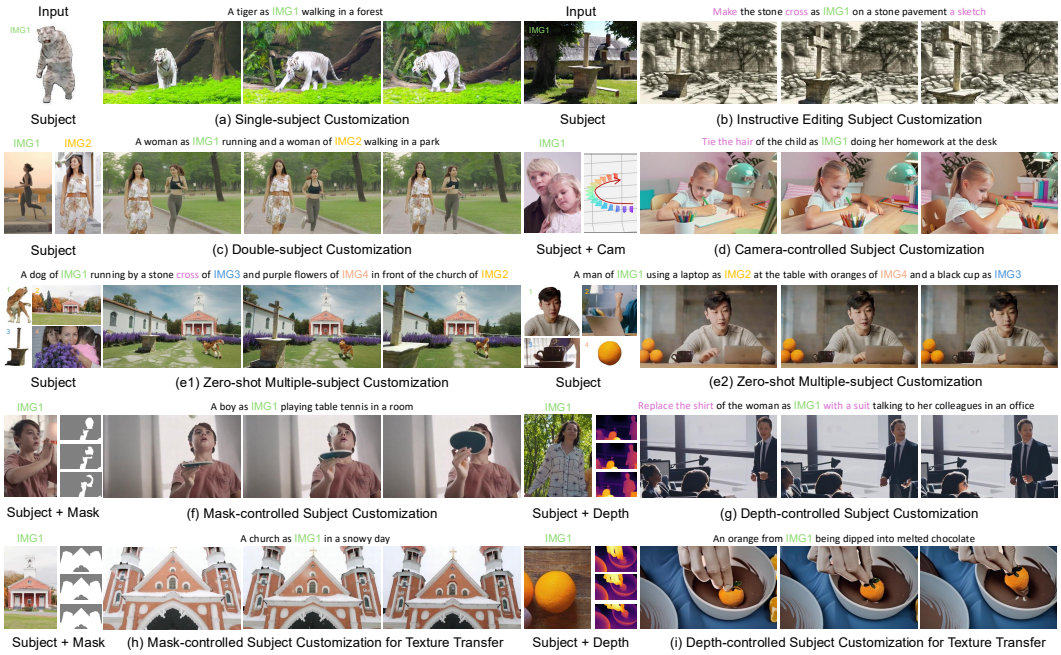

Figure 1: In (a) and (c), our method can change the pose and action of subjects. (b) The instructive editing texts are purple. In (e1) and (e2), our method trained with two subjects but can compose more subjects in inference. (d), (f), and (g) are the results under different controls. In the challenging cases of (g), (h), and (i), the subjects are unaligned with the mask or depth, but our method can adapt the shape or transfer the texture of the subjects.

To cope with these problems, we firstly propose a data construction pipeline, VideoCus-Factory, to produce training data pairs for multi-subject customization from raw videos without any labels. Our VideoCus-Factory first selects a frame from a video and uses a multimodal large language model to caption the frame and detect the subjects. Then we perform subject filtering, data augmentation, and random background placement to prevent the leakage of subject size, position, and background. This improves the variation and grounding ability of the training model and enables the inference of subject images with background. Based on our constructed data, we develop an Image-Video Transfer Mixed (IVTM) training with image instructive editing data to enable instructive editing effect for the subject in the customized video. Besides, our VideoCus-Factory can also generate signal data pairs such as depth-to-video and mask-to-video to control the subject-driven customization. Secondly, we propose a DiT-based framework, OmniVCus, to train on our constructed data. The input images, videos, and control signals are patchified, encoded, and concatenated into a long 1D token to input into OmniVCus. In particular, we design two embeddings in OmniVCus. As the number of subjects in a training video is limited, we propose a Lottery Embedding (LE) to enable customization with more subjects in inference than those used in training. The core idea of LE is to use a limited number of training subjects to activate the frame embeddings of more subjects. To enable more effective control effect of conditions, we propose a Temporally Aligned Embeddings (TAE). Our TAE assigns the same frame embeddings to the noise tokens and temporally aligned control tokens with dense semantic information, such as mask and depth. For the sparse viewpoint control signal without semantic information, TAE feeds them into a multi-layer perceptron (MLP) and then add them to the noise tokens to reduce the token length and computational complexity. Benefit from the constructed data and proposed techniques, our method can flexibly compose multimodal conditions to control the video customization, as shown in Fig. 7. In a nutshell, our contributions can be summarized as:

(i) We design a data construction pipeline, VideoCus-Factory, to produce training data pairs and control signals for subject-driven video customization from only raw videos. Based on our data, we develop an IVTM training strategy to enable instructive editing effect of the subject in the video.

(ii) We propose a new method, OmniVCus, with two embedding designs, LE and TAE, for subject-driven video customization. LE enables video customization with more subjects in inference than training. TAE enables more effective control of temporally aligned signals to the customized video.

(iii) Experiments show that our method significantly outperforms state-of-the-art (SOTA) methods in quantitative metrics while yielding more visually favorable, editable, and controllable results.

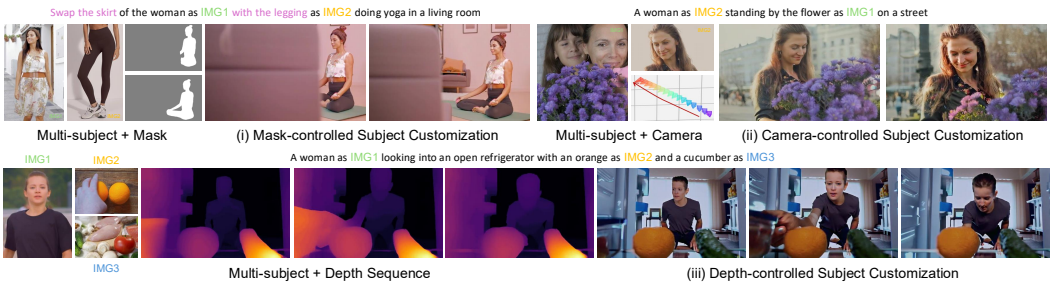

Figure 2: Our method can flexibly compose different conditions to control multi-subject video customization.

2 Related Work

Text-to-Video (T2V) Diffusion Models [1–3, 15–28] have witnessed significant progress in recent years. Preliminary T2V diffusion models are mainly based on stable diffusion [29], which formulates the diffusion process in latent space and uses a U-shaped convolutional neural network (CNN) [30] as the denoiser. Yet, the model capacity of CNN is limited for large-scale training. Thus, a later work DiT [31] employs Transformer [32] to replace U-Net in diffusion. These DiT-based methods [2,33–43] show very impressive performance in video generation and flexibility to add control conditions just by extending the input 1D tokens. This work exploits the advancement of the DiT framework to explore its potential in subject-driven video customization under different modalities of control signals.

Subject-driven Video Customization approaches are mainly divided into two categories: tuning-based [4,5,44–52] and feedforward methods [12,14,53–60]. Prior tuning-based methods are typically focused on single-subject scenarios. For instance, DreamVideo [4] fine-tunes an identity adapter and combines textual inversion to customize video for a subject. Videomage [52] employs subject and motion LoRAs to capture personalized content for multi-subject customization. These methods require a long time for inference. To avoid test-time tuning, later works [53] develop feedforward solutions. Some recent efforts [11, 12, 54] such as ConceptMaster [11] and Video Alchemist [12] try to construct data pairs for multiple subjects but their data pipelines still have limitations. Plus, how to add control to subject-driven video customization is still under-explored.

3 Method

3.1 VideoCus-Factory

Our data construction pipeline, VideoCus-Factory, is depicted in Fig. 3. VideoCus-Factory can produce training data pairs for multi-subject video customization from raw videos without any labels.

Video Captioning. For a video sequence, we first randomly select a frame and then use the multi-modal large language model, Kosmos-2 [61], to caption it and detect the subjects in the frame. Take the video in Fig. 3 as example, Kosmos-2 outputs the caption "An image of a bride and groom walking away from a car and looking back at it", a list of subjects ["a bride", "groom", "car"], the starting and ending positions of the subjects in the caption [[12, 19], [24, 29], [48, 53]], and bounding boxes of the detected subjects. We modify the caption by removing the prefix "An image of" and plug in image labels such as IMG1 and IMG2 corresponding to the subjects with their positions in the caption.

Subject Filtering. We feed the detected bboxes and raw video into SAM2 [62] to track and segment the subjects. Then we filter out the failure segmentation cases by thresholding the average segmentation values across frames. For example, in Fig. 3, "a car" is not segmented in some frames. Thus, we filter it out. We also filter out some word clouds and large background without identity.

Data Augmentation. If we directly use the images of segmented subjects to train a generation model, the scales, positions, and poses of subjects are leaked during training. As a result, the model tends to learn image animation with less variation. In addition, since the segmented subjects do not have background like ConceptMaster [11], directly training with this data degrades the model's grounding ability, limits its application as it needs to crop out the subject first, and easily leads to the copy-paste effect. To handle these problems, we randomly rotate the subjects, rescale them, move them to the center position, and augment their colors. Then we randomly select an image, which may also be pure white, and place it as the background for the subject to derive the input images of training data.

Control Signals. To add control conditions, VideoCus-Factory also constructs control signal training data pairs. As shown in Fig. 3, the mask sequence of the subject and the raw video with the modified

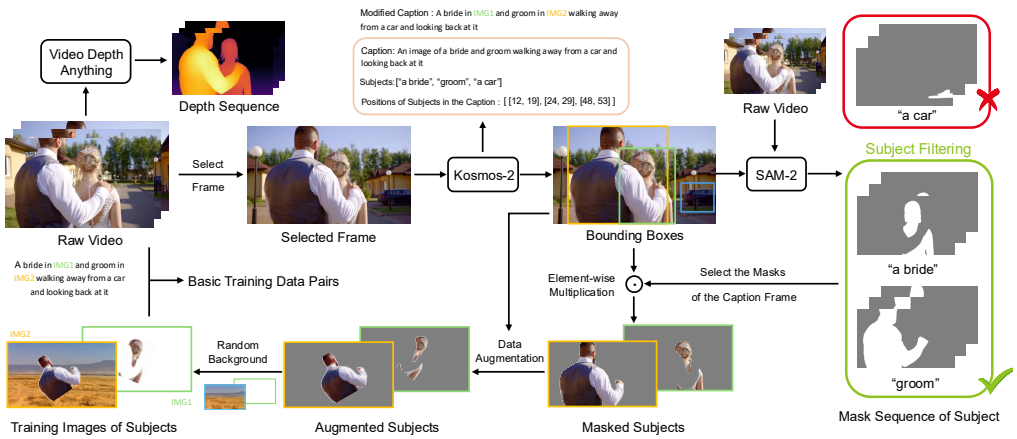

Figure 3: Our data construction pipeline VideoCus-Factory uses Kosmos-2 [61] to caption the raw video and detect the subjects. Then we use SAM-2 [62] to segment and filter the detected subjects to derive the training input images. VideoCus-Factory also constructs control data pairs such as mask-to-video and depth-to-video.

caption without image labels form the data pairs for mask-to-video control data. The depth sequence predicted by the video-depth-anything [63] and the raw video with the same caption form the depth-to-video control data pairs. Note that the mask-to-video and depth-to-video data are not paired with the subject-driven customization data in training. Our model can flexibly compose them in inference.

3.2 OmniVCus

As shown in Fig. 4, OmniVCus is a DiT-based framework. It can be mixed trained with different tasks, including single-/double-subject customization, depth-/mask-to-video, text-to-multiview, text-to-image/-video, and image instructive editing. Texts, images, and videos are patchified, encoded into the latent space, concatenated with the noise tokens, and fed into the full-attention DiT. OmniVCus can flexibly compose tokens of different signals to control and edit the subject in the customized video. We notice that the condition frame tokens are mainly divided into two parts: the image tokens containing the subjects, and the other temporally aligned tokens of control signals. To handle these two types of tokens, we design Lottery Embedding (LE) and Temporally Aligned Embedding (TAE).

Lottery Embedding. As the subjects in each training sample are limited, it is important to enable customization with more subjects in inference than training. To this end, our Lottery Embedding (LE) uses the limited subjects in the training samples to activate more frame embeddings, as shown in Fig. 4 (a). Denote the max number of subjects in a training sample as K and the number of subjects we aim to compose as $M (M > K)$. Then LE randomly selects a set $\mathcal{S}$ of K numbers from $[1, M]$ as

$$\mathcal{S} \sim \text{Unif}\big(\{\, \mathcal{A} \subseteq \{1, \ldots, M\} \mid |\mathcal{A}| = K \,\}\big), \tag{1}$$

where Unif denotes the uniform distribution. Since the Transformer is unordered, we need to create an order for it on the frame embedding and match the order of the image index label. Thus, we sort $\mathcal{S}$ in ascending order to derive $\mathcal{S}_{\uparrow}$ and then assign the elements of $\mathcal{S}_{\uparrow}$ as frame embeddings to the input images of subjects. These frame embeddings are reshaped, undergo an MLP, and then added to the tokens of the corresponding subject images. By our LE, we can activate more frame position embeddings during training and enable zero-shot more-subject video customization in inference.

Temporally Aligned Embedding. We notice that the control signals, such as camera, depth, and mask are temporally aligned with the generated video. Thus, to better direct the Transformer model to extract guidance from these control signals, TAE assigns the same frame embeddings to the control tokens and noise tokens. In Fig. 4 (b), we denote the length of the generated video as N. Then the frame position embeddings for control tokens and noise tokens are $\{M + 1, M + 2, \cdots, M + N\}$. To distinguish the control signals and noise, we only add the timestep embedding to the noise tokens.

In particular, the depth and mask sequences are structural signals containing fine-grained spatial and semantic information. Hence, we use a 3D-VAE to encode them to preserve these information. In contrast, the camera signals do not contain such information. On the other hand, the computational complexity of Transformer is quadratic to the length of input tokens. Thus, we integrate the camera signals into noise instead of concatenating them. Specifically, to enhance the control of camera signals and spatially align them to the noise tokens, we adopt the pixel-aligned ray embeddings,

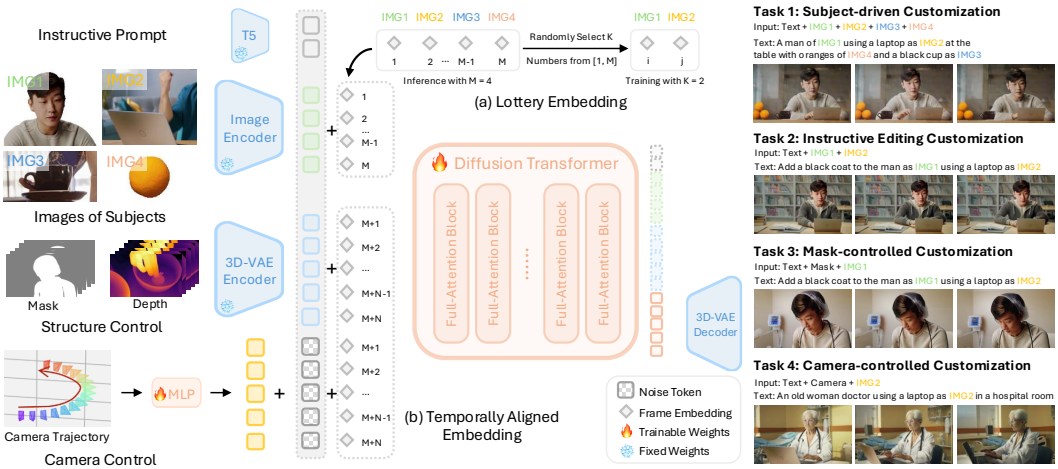

Figure 4: OmniVCus is DiT architecture that can compose different input signals to customize a video. (a) LE enables more-subject customization in inference by activating more frame embeddings with training subjects. (b) TAE extracts the guidance from control signals by aligning the frame embeddings of condition and noise tokens.

plücker coordinates, parameterized as $\boldsymbol{r} = (\boldsymbol{o} \times \boldsymbol{d}, \boldsymbol{d})$, where $\boldsymbol{o}$ and $\boldsymbol{d}$ are the position and direction of the ray landing on a pixel. Then the plücker coordinates are patchified and undergo an MLP to add with the noise. Denote the input structure control tokens as $\mathbf{T}_{sc}^{in}$, the noise tokens as $\mathbf{T}_{n}^{in}$, and the tokens of plücker coordinates as $\mathbf{T}_{c}^{in}$. Then the mapping function of TAE, f_{TAE}, is formulated as

$$\begin{aligned}(\mathbf{T}_{sc}^{out}, \mathbf{T}_{n}^{out}) &= f_{\text{TAE}}(\mathbf{T}_{sc}^{in}, \mathbf{T}_{n}^{in}, \mathbf{T}_{c}^{in}) \\ &= \Big(\mathbf{T}_{sc}^{in} + \text{MLP}_f(p_f), \mathbf{T}_{n}^{in} + \text{MLP}_f(p_f) + \text{MLP}_t(t) + \text{MLP}_c(\mathbf{T}_{c}^{in})\Big),\end{aligned} \tag{2}$$

where p_f is the frame position, t is the timestep, and MLP_f, MLP_t, MLP_c are three MLPs. The output tokens $\mathbf{T}_{sc}^{out}$ and $\mathbf{T}_{n}^{out}$ are then concatenated with other tokens to be fed into the DiT model.

Image-Video Transfer Mixed Training. Due to the lack of training data pairs for subject-driven customization with instructive editing. We develop an Image-Video Transfer Mixed (IVTM) training strategy to enable the editing effect for the subject of interest without constructing new data pairs. As the image instructive editing training data is sufficient, our goal is to transfer the editing effect from image to video. To this end, we need to construct a common task on image and video as the bridge for the knowledge transfer of instructive editing from image-to-image to image-to-video. In our IVTM training, this common task pair is single-subject image and video customization. For the image customization, we select the caption frame in Fig. 3 with the processed image of subject as the training pairs. In Fig. 4, we align the frame position embeddings of the input images of the image instructive editing and single-subject image/video customization in our IVTM training for better transferring. This frame embedding is also assigned by our LE in Eq. (1) with $K = 1$ to allow better composing instructive editing with multi-subject customization. In inference, we compose the prompts of image instructive editing and subject-driven customization to activate the editing effect.

Training Objective. Our model is mixed trained with different tasks using the flow-matching loss [64, 65]. Specifically, denoting the ground-truth video latents as $\mathbf{X}^1$ and noise as $\mathbf{X}^0 \sim \mathcal{N}(0, 1)$, the noisy input is produced by linear interpolation $\mathbf{X}^t = t\mathbf{X}^1 + (1-t)\mathbf{X}^0$ at timestep t. The model v_θ predicts the velocity $\mathbf{V}^t = \frac{\partial}{\partial t}\mathbf{X}^t = \mathbf{X}^1 - \mathbf{X}^0$. Then the overall training objective is formulated as

$$\min_{\theta} \mathbb{E}_{t,\mathbf{X}^0,\mathbf{X}^1}\Big[\big|\big|\mathbf{V}^t - v_\theta(\mathbf{X}^t, t \mid C_{txt}, C_{img}, C_{depth}, C_{mask}, C_{traj})\big|\big|_2^2\Big], \tag{3}$$

where $C_{txt}, C_{img}, C_{depth}, C_{mask}$, and C_{traj} denote the input texts, images, depths, masks, and cameras. As different training tasks are not paired with each other, some input conditions are missing in different training samples. But our model can flexibly compose different input signals in inference.

4 Experiment

4.1 Experimental Settings

Dataset. For subject-driven video customization, depth-to-video, and mask-to-video generation, we use our VideoCus-Factory to create ~1.2M, ~1.4M, and ~1.6M training data pairs. The data for these

Figure 5: Visual comparison of single-subject video customization with state-of-the-art algorithms. Our method can change the pose and viewpoint of the subject while keeping the identity such as the hair, jacket, and sweater.

Methods	Subject	CLIP-T	CLIP-I	DINO-I	Consistency	Dynamic
VideoBooth [53]	Single	0.2541	0.5891	0.3033	0.9593	0.4287
DreamVideo [4]	Single	0.2799	0.6214	0.3792	0.9609	0.4696
Wan2.1-I2V† [1]	Single	0.2785	0.6319	0.4203	0.9754	0.5310
SkyReels [54]	Single	0.2820	0.6609	0.4612	0.9797	0.5238
Ours	Single	0.3293	0.7154	0.5215	0.9928	0.5541
SkyReels [54]	Multiple	0.2785	0.6429	0.4107	0.9710	0.5892
Ours	Multiple	0.3264	0.6672	0.4965	0.9908	0.6878

(a) Comparison of subject-driven video customization.

Methods	Alignment	Identity	Quality
VideoBooth [53]	91.9	97.3	94.6
DreamVideo [4]	89.2	89.2	97.3
Wan2.1-I2V† [1]	83.8	86.4	73.0
SkyReels [54]	75.7	81.1	70.3
Ours	–	–	–
SkyReels [54]	73.0	78.4	67.6
Ours	–	–	–

(b) User preference (%) of our method.

Methods	CLIP-T	CLIP-I	DINO-I	Consistency	Dynamic
VideoBooth [53]	0.2453	0.5935	0.2989	0.9570	0.5825
DreamVideo [4]	0.2642	0.6116	0.3511	0.9604	0.5760
Wan2.1-I2V [1]	0.2690	0.6208	0.3722	0.9734	0.6157
SkyReels [54]	0.2761	0.6368	0.4259	0.9635	0.5904
Ours	0.3126	0.7061	0.4942	0.9915	0.6226

(c) Comparison of instructive editing for subject customization.

Methods	Alignment	Identity	Quality
VideoBooth [53]	94.6	94.6	91.9
DreamVideo [4]	91.9	97.3	94.6
Wan2.1-I2V [1]	78.4	81.1	67.6
SkyReels [54]	73.0	75.7	64.9
Ours	–	–	–

(d) User preference (%) of our method.

Methods	CLIP-T	CLIP-I	DINO-I	Consistency	Dynamic
Motionctrl [66]	0.2984	0.5215	0.2066	0.9857	0.4272
Cameractrl [67]	0.2909	0.5163	0.1982	0.9711	0.5845
CamI2V [68]	0.2871	0.5365	0.2248	0.9660	0.5623
Ours	0.3104	0.6751	0.5233	0.9911	0.6204

(e) Comparison of cameral-controlled subject customization.

Methods	Alignment	Identity	Quality
Motionctrl [66]	91.9	86.8	78.4
Cameractrl [67]	70.3	89.2	91.9
CamI2V [68]	73.0	83.8	81.1
Ours	–	–	–

(f) User preference (%) of our method.

Table 1: Quantitative results and user study of state-of-the-art subject-driven video customization methods.

three tasks are not paired with each other, and every input video sequence to our VideoCus-Factory is randomly selected from our internal video data pool. For subject-driven video customization, the scale factor is randomly selected from 0.7 to 1.3 and the color augmentation includes brightness scaling (0.9∼1.1), linear contrast adjustment (0.9∼1.1), saturation scaling (0.9∼1.1), and hue shift (-10°∼10°). For text-to-multiview, we select 320K samples from Objaverse [69] labeled with long and short text prompts as the training samples. We adopt the OmniEdit [70] as the image instructive editing dataset containing 1.2M data pairs. Besides, we also fine-tune the model with text-to-image (∼300M) and text-to-video (∼1M) data. In evaluation, we collect 112 samples for single-subject customization and instructive editing customization, 76/74/56 samples for double-/triple-/quadruple-subject customization, and 112 samples for camera-controlled subject-driven video customization.

Implementation Details. Our model is fine-tuned from a T2V model with 5B parameters for 100K steps in total at a batch size of 356 on 64 A100 GPUs for 5 days. We adopt the AdamW optimizer [71] ($\beta_1 = 0.9$, $\beta_2 = 0.95$) with a weight decay of 0.1. The learning rate is linearly warmed up to $1e^{-5}$ with 2K iterations and decays to $1e^{-6}$ using cosine annealing [72]. The spatial resolution of training images and videos is set to 512×512 for text-to-multiview and image instructive editing and 384×640 for other tasks. The frame number and fps of the training video are set to 64 and 24. We use five metrics for evaluation. (1) CLIP-T computes the average cosine similarity between CLIP [73] image embeddings of all generated frames and their text embedding. We remove the image index label and adapt the text prompts after instructive editing when computing CLIP-T. (2) CLIP-I calculates the average cosine similarity between the CLIP image embeddings of all generated images and the target images. (3) DINO-I [74] also measures the visual similarity between the generated and target subjects using ViTS/16 DINO [75]. (4) Temporal Consistency computes the CLIP image embeddings

Figure 6: Visual comparison of instructive editing for subject-driven video customization with SOTA methods. VideoBooth, DreamVideo, and SkyReels-A2 first use OmniGen to edit the subject and then customize the video. Wan2.1-I2V [1] uses OmniGen to edit and customize the subject and then animate the image to derive the video.

Figure 7: Visual comparison of multi-subject video customization with the SOTA method SkyReels-A2 [54].

and averages the cosine similarity between every pair of consecutive frames. (5) Dynamic Degree is computed as the optical flow predicted by RAFT [76] magnitude between consecutive frames.

4.2 Main Results

Composing Different Control Conditions. As shown in Fig. 1 and 7, our model can flexibly compose different control signals. **(i)** In Fig. 1 (b), benefit from IVTM training, our model can modify the subject and transfer its style to sketch. **(ii)** Benefit from LE, our model trained with 2 subjects but can compose 4 subjects in inference, as shown in Fig. 1 (e1) and (e2). **(iii)** In Fig. 1 (f) and (g), our model can change the pose and action of subjects following the mask or depth. (g) is a hard case where the depth is from a man but our model can fill the depth with the woman while keeping the face identity and swapping her shirt with a suit following the instruction. In harder cases where the subjects are severely unaligned with the mask or depth, our model can still transfer the texture of subjects. In Fig. 1 (h), the model transfers the appearance of church to the mask of house. In Fig. 1 (i), the texture of orange is transferred to the depth of strawberry. **(iv)** Even for more challenging multi-subject cases under different control signals in Fig. 7, our model can still robustly handle them.

Comparison with SOTA Methods. (i) We compare OmniVCus with 4 SOTA methods including an I2V method (Wan2.1-I2V [1]), two single-subject video customization methods (DreamVideo [4] and VideoBooth [53]), and a multi-subject video customization method (SkyReels-A2 [54]) on subject-driven video customization without and with instructive editing in Tab. 1a and 1c. In Tab. 1a, Wan2.1-I2V [1] uses the SOTA image customization model OmniGen [77] to first customize the subject and then animate it. The results for multi-subject customization are averaged on double- and triple-subject customization, as SkyReels-A2 can support at most three subjects. In 1c, OmniGen is first used to instructively edit the subject for all compared baselines as they show limitations in editing the subject itself. Our OmniVCus significantly outperforms previous methods in all tracks in both Tab. 1a and 1c, suggesting its advantages in identity preserving and high-quality video generation. Fig. 5, 6, and 7 show the visual comparisons of single-subject customization without and with instructive editing and multi-subject customization. In Fig. 5, our method can better change the pose of the woman while keeping the identity such as the hair, jacket, and sweater. In Fig. 6, our method can better follow the text to remove the blazer while keeping the man's identity and customize the background. In Fig. 7, SkyReels-A2 misses some subjects and tends to animate the image. In contrast, our method can better compose all the subjects and follow the texts to customize the video.

Figure 8: Comparison of cameral-controlled subject-driven video customization. Motionctrl, Cameractrl, and CamI2V first employ OmniGen to customize the image of subject and then animate it following the camera.

Method	CLIP-T	DINO-I	Consistency	Dynamic
Baseline	0.2175	0.2405	0.9588	0.3759
+ Subject Filtering	0.2431	0.5053	0.9617	0.3826
+ Data Augementation	0.3293	0.5215	0.9928	0.5541

(a) Ablation of VideoCus-Factory data construction

Embedding	CLIP-T	DINO-I	Consistency	Dynamic
Naive	0.2618	0.2947	0.9751	0.4948
Add-to-Noise	0.1722	0.1680	0.9319	0.5437
Our TAE	0.3054	0.3794	0.9909	0.4965

(b) Ablation of our Temporally Alignment Embedding

Emebedding	CLIP-T	DINO-I	Consistency	Dynamic
w/o LE	0.2105	0.3364	0.9702	0.6943
with LE	0.2728	0.4163	0.9810	0.6806

(c) Ablation study of our Lottery Embedding mechanism

Training Method	No Mixed	Direct Mixed	Our IVTM
CLIP-T	0.2137	0.2585	0.3126
User Pref. (%)	91.9	75.7	–

(d) Ablation of our mixed training for editing effect

Table 2: Ablation study. (a) is conducted on single-subject customization. (b) is done on depth-controlled customization. (c) is done on multi-subject customization. (d) is done on instructive editing subject customization.

(ii) Tab. 1e compares OmniVCus with three SOTA camera-controlled I2V methods (Motionctrl [66], Cameractrl [67], and CamI2V [68]). They use OmniGen to customize the subject first and then animate it. Our method surpasses the recent best method CamI2V by 0.2985 in DINO-I. Fig. 8 shows the visual results. Our method can better keep the identity of woman and follow the camera trajectory.

(iii) We also conduct a user study with 37 participants on single-subject video customization without and with instructive editing and camera-controlled subject-driven video customization in Tab. 1b, 1d, and 1f. Each participant views the videos generated by OmniVCus and a random competing method, along with the images of the subject, text prompts, and control signals. The participants are asked three questions: 1) Which video aligns better with the customization prompt/editing instruction/camera trajectory? 2) Which video better keeps the identity of the subject? 3) Which video has better quality? As reported in Tab. 1b, 1d, and 1f, reports the user preference (%) of our method over competing entries. Our method outperforms all SOTA methods by a large margin.

4.3 Ablation Study

VideoCus-Factory. We conduct experiments on single-subject video customization to study the steps of our VideoCus-Factory in Tab. 2a. We remove the subject filtering and data augmentation including random background placement from VideoCus-Factory as the baseline. When we apply the subject filtering, the DINO-I score significantly improves by 0.2648 because the left training video data after filtering can keep the subject in all frames. Subsequently, when using the data augmentation, the CLIP-T and dynamic degree gain by 0.0862 and 0.1715. This is because the segmented subject without data augmentation leaks the position, scale, and background in the training process. As a result, the customized videos have less variation and suffer from the copy-paste issue. As shown in Fig. 9 (a), with our data augmentation, the dog can change its pose in the customized video.

Temporally Aligned Embedding. We conduct experiments to study the effect on depth-controlled subject video customization in Tab. 2b. We compare our TAE with two options: 1) Naive method that assigns different frame embeddings to the tokens of depth (length N) and noise (length N) from $M+1$ to $M+2N$ and the timestep embedding is added to all tokens. 2) Adding the depth to the noise after undergoing an MLP. As listed in Tab. 2b, the add-to-noise method causes model collapse because the fine-grained spatial information in depth tokens is corrupted by the noise. Our TAE surpasses the naive embedding by a large margin in CLIP-T and DINO-I scores. As shown in Fig. 9 (c1) and (c2), our TAE can help the model better follow the guidance of depth to generate higher-quality video with fewer artifacts. Besides, TAE can help the control signals to be better composed with other tokens. The naive embedding fails to customize the bottle of the beer from text tokens and keep the identity of the boy from image tokens. In contrast, TAE can preserve the identity and follow the prompts to customize the scenario. Fig. 9 (d1) and (d2) compare the naive embedding

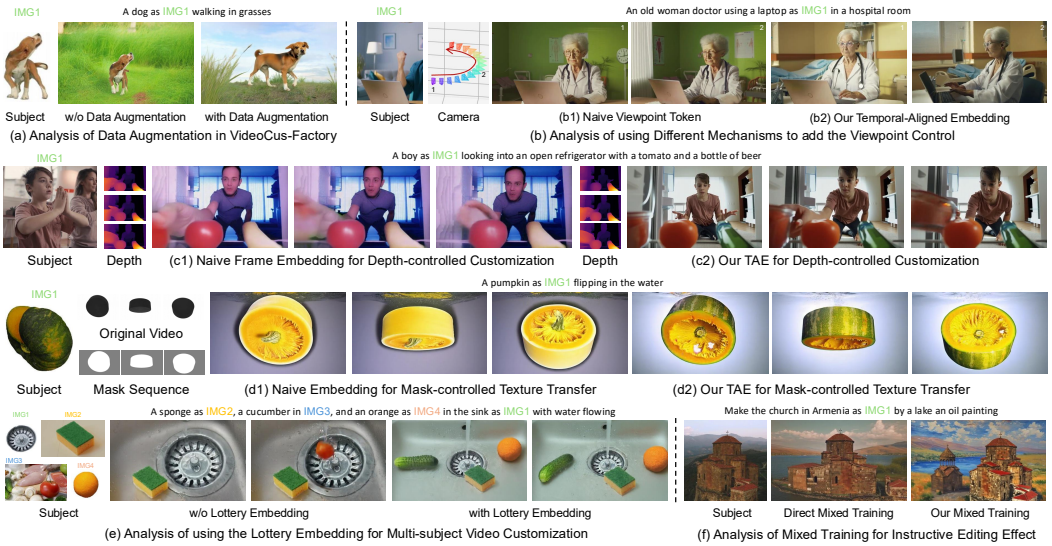

Figure 9: Visual analysis. (a) Using the data augmentation in our VideoCus-Factory can vary the scale, pose, and action of subject in the customized video. (b) Our TAE can better control the viewpoint. (c1) and (c2) show that our TAE can improve the consistency, video quality, and guidance of depth. (d1) and (d2) show that our TAE can better handle the case that the subject is not aligned with the mask by transferring the texture. (e) Using our LE better preserves the identity in zero-shot multi-subject customization. (f) studies the effect of IVTM training.

and TAE on mask-controlled customization. When the subject is unaligned with the mask of the black cylinder, the naive embedding customizes low-quality video without the pumpkin texture while generating undesired black edge. In contrast, TAE can better transfer the texture with less artifacts.

We also compare the camera embedding in TAE with the naive method that directly concatenates the viewpoint tokens into the overall long 1D tokens. This naive method leads to an increase in training time by 10% as the computational complexity of self-attention is quadratic to the length of input tokens. As shown in Fig. 9 (b1) and (b2), our TAE can control the viewpoint rotation more effectively.

Lottery Embedding. We conduct experiments on triple- and quadruple-subject customization that do not appear in the training data to study the effect of our LE on zero-shot more-subject customization. The averaged results are reported in Tab. 2c. When using our LE, the CLIP-T and DINO-I scores are significantly improved by 0.0623 and 0.0799. We also conduct a visual analysis in Fig. 9 (e), without using LE, the customized video misses the subject of orange and mistakenly extracts the tomato from IMG3 instead of the desired subject, the cucumber. In contrast, using our LE can accurately compose the four subjects with proper scales and physically consistent motion in the customized video.

Image-Video Transfer Mixed Training. We conduct experiments on the instructive editing single-subject customization in Tab. 2d to study the effect of our IVTM training strategy. CLIP-T score and the user preference percentage of IVTM training are reported. Our IVTM performs much better than no mixed training and direct mixed training with the OmniEdit dataset. We observe that the direct mixed training still can not enable some hard instructive editing categories such as removal, color change, style transfer, *etc.* Fig. 9 (f) shows an example of style transfer. The direct mixed training can not follow the editing instruction while our IVTM can customize the church in an oil painting style.

5 Conclusion

In this paper, we focus on studying the subject-driven video customization with different control conditions in a feedforward manner. We first propose a data construction pipeline, VideoCus-Factory, to produce training data pairs. Our VideoCus-Factory can also create depth-to-video and mask-to-video control signal data. Subsequently, we present a DiT-based framework, OmniVCus, with two embedding mechanisms, LE and TAE. LE enables more-subject video customization in inference by using limited training subjects to activate more frame position embeddings. TAE enhances the control effect of temporally aligned signals by assigning the same frame embeddings to the control tokens and noise tokens. Experiments show that our method outperforms SOTA algorithms in both quantitative and qualitative evaluations while achieving more flexible control and editing effects.

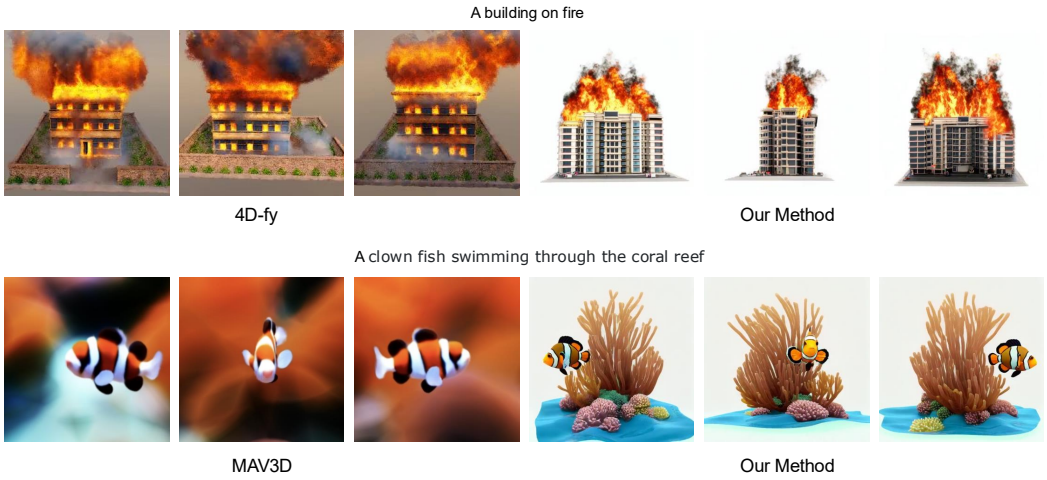

Figure 10: Text-to-4D generation comparison with 4D-fy [78] (upper) and MAV3D [79] (lower).

6 Limitations

The main limitation of our method is that it takes a long time (about three months) to create data. Meanwhile, the labeling quality is also constrained by the capability of the used foundational models. Specifically, the quality of the caption label and detection bounding boxes is constrained by the model capability of Kosmos-2 [61]. The quality of mask-to-video and depth-to-video control data pairs is constrained by SAM2 [62] and Video-Depth-Anything [63]. However, we believe that as the capability of these foundational models improves, the quality of data labeling will be better.

7 Broader Impact

Subject-driven video customization is an important and challenging topic in computer vision. It enables creators to quickly generate or edit video content based on specific subjects (people, objects or scenes), giving rise to a variety of applications: from short video clips of a few seconds on social media, to special effects synthesis at the film industry level, to personalized narratives and immersive experiences in advertising, education and digital cultural heritage. This technology not only lowers the threshold for high-quality content production, unleashes the creative potential of small and medium-sized studios and independent bloggers, but also brings higher efficiency and more flexible iteration space to the traditional film and television production process.

Until now, subject-driven video customization techniques have no negative social impact yet. Our proposed method does not present any negative foreseeable societal consequences, either.

8 Model Emergence Capability: Text-to-4D

As aforementioned, our model mixed trained with customization data (dynamic) and text-to-3D data (static) can perform text-to-4D generation. Please note that the 4D here refers to dynamic multi-view. Our model does not directly contain 4D representation.

We compare our method with two SOTA text-to-4D generation methods: MAV3D [79] and 4D-fy [78] in Fig. 10. Our method can generate more 3D-consistent novel views with higher dynamic degree.

Please refer to our project page for more video dynamic generation results.

Acknowledgement

This work was supported by the office of Naval Research with award N000142412696 and in part by Adobe Inc.

References

[1] A. Wang, B. Ai, B. Wen, C. Mao, C.-W. Xie, D. Chen, F. Yu, H. Zhao, J. Yang, J. Zeng, *et al.*, "Wan: Open and advanced large-scale video generative models," *arXiv preprint arXiv:2503.20314*, 2025.

[2] Z. Yang, J. Teng, W. Zheng, M. Ding, S. Huang, J. Xu, Y. Yang, W. Hong, X. Zhang, G. Feng, *et al.*, "Cogvideox: Text-to-video diffusion models with an expert transformer," in *ICLR*, 2025.

[3] T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing, D. Schnurr, J. Taylor, T. Luhman, E. Luhman, C. Ng, R. Wang, and A. Ramesh, "Video generation models as world simulators." *Sora*. Accessed Mar.1, 2024 [Online].

[4] Y. Wei, S. Zhang, Z. Qing, H. Yuan, Z. Liu, Y. Liu, Y. Zhang, J. Zhou, and H. Shan, "Dreamvideo: Composing your dream videos with customized subject and motion," in *CVPR*, 2024.

[5] E. Molad, E. Horwitz, D. Valevski, A. R. Acha, Y. Matias, Y. Pritch, Y. Leviathan, and Y. Hoshen, "Dreamix: Video diffusion models are general video editors," *arXiv preprint arXiv:2302.01329*, 2023.

[6] Y. Wei, S. Zhang, H. Yuan, B. Gong, L. Tang, X. Wang, H. Qiu, H. Li, S. Tan, Y. Zhang, *et al.*, "Dreamrelation: Relation-centric video customization," *arXiv preprint arXiv:2503.07602*, 2025.

[7] L. Zhang, A. Rao, and M. Agrawala, "Adding conditional control to text-to-image diffusion models," in *ICCV*, 2023.

[8] C. Mou, X. Wang, L. Xie, Y. Wu, J. Zhang, Z. Qi, and Y. Shan, "T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models," in *AAAI*, 2024.

[9] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, *et al.*, "Lora: Low-rank adaptation of large language models.," in *ICLR*, 2022.

[10] Y. Wei, S. Zhang, H. Yuan, X. Wang, H. Qiu, R. Zhao, Y. Feng, F. Liu, Z. Huang, J. Ye, *et al.*, "Dreamvideo-2: Zero-shot subject-driven video customization with precise motion control," *arXiv preprint arXiv:2410.13830*, 2024.

[11] Y. Huang, Z. Yuan, Q. Liu, Q. Wang, X. Wang, R. Zhang, P. Wan, D. Zhang, and K. Gai, "Conceptmaster: Multi-concept video customization on diffusion transformer models without test-time tuning," *arXiv preprint arXiv:2501.04698*, 2025.

[12] T.-S. Chen, A. Siarohin, W. Menapace, Y. Fang, K. S. Lee, I. Skorokhodov, K. Aberman, J.-Y. Zhu, M.-H. Yang, and S. Tulyakov, "Multi-subject open-set personalization in video generation," in *CVPR*, 2025.

[13] T. Hu, Z. Yu, Z. Zhou, S. Liang, Y. Zhou, Q. Lin, and Q. Lu, "Hunyuancustom: A multimodal-driven architecture for customized video generation," *arXiv preprint arXiv:2505.04512*, 2025.

[14] X. Ju, W. Ye, Q. Liu, Q. Wang, X. Wang, P. Wan, D. Zhang, K. Gai, and Q. Xu, "Fulldit: Multi-task video generative foundation model with full attention," *arXiv preprint arXiv:2503.19907*, 2025.

[15] X. Chen, Z. Liu, M. Chen, Y. Feng, Y. Liu, Y. Shen, and H. Zhao, "Livephoto: Real image animation with text-guided motion control," in *ECCV*, 2024.

[16] J. Xing, M. Xia, Y. Zhang, H. Chen, W. Yu, H. Liu, G. Liu, X. Wang, Y. Shan, and T.-T. Wong, "Dynamicrafter: Animating open-domain images with video diffusion priors," in *ECCV*, 2024.

[17] J. Wang, H. Yuan, D. Chen, Y. Zhang, X. Wang, and S. Zhang, "Modelscope text-to-video technical report," *arXiv preprint arXiv:2308.06571*, 2023.

[18] A. Blattmann, R. Rombach, H. Ling, T. Dockhorn, S. W. Kim, S. Fidler, and K. Kreis, "Align your latents: High-resolution video synthesis with latent diffusion models," in *CVPR*, 2023.

[19] R. Girdhar, M. Singh, A. Brown, Q. Duval, S. Azadi, S. Rambhatla, A. Shah, X. Yin, D. Parikh, and I. Misra, "Emu video: Factorizing text-to-video generation by explicit image conditioning," *arXiv preprint arXiv:2311.10709*, 2023.

[20] Z. Luo, D. Chen, Y. Zhang, Y. Huang, L. Wang, Y. Shen, D. Zhao, J. Zhou, and T. Tan, "Videofusion: Decomposed diffusion models for high-quality video generation," in *CVPR*, 2023.

[21] A. Mahapatra, A. Siarohin, H.-Y. Lee, S. Tulyakov, and J.-Y. Zhu, "Text-guided synthesis of eulerian cinemagraphs," *ACM TOG*, 2023.

[22] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni, *et al.*, "Make-a-video: Text-to-video generation without text-video data," *arXiv preprint arXiv:2209.14792*, 2022.

[23] D. Zhou, W. Wang, H. Yan, W. Lv, Y. Zhu, and J. Feng, "Magicvideo: Efficient video generation with latent diffusion models," *arXiv preprint arXiv:2211.11018*, 2022.

[24] Y. He, T. Yang, Y. Zhang, Y. Shan, and Q. Chen, "Latent video diffusion models for high-fidelity long video generation," *arXiv preprint arXiv:2211.13221*, 2022.

[25] A. Blattmann, T. Dockhorn, S. Kulal, D. Mendelevitch, M. Kilian, D. Lorenz, Y. Levi, Z. English, V. Voleti, A. Letts, *et al.*, "Stable video diffusion: Scaling latent video diffusion models to large datasets," *arXiv preprint arXiv:2311.15127*, 2023.

[26] J. Ho, T. Salimans, A. Gritsenko, W. Chan, M. Norouzi, and D. J. Fleet, "Video diffusion models," *NeurIPS*, 2022.

[27] S. Zhang, J. Wang, Y. Zhang, K. Zhao, H. Yuan, Z. Qin, X. Wang, D. Zhao, and J. Zhou, "I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models," *arXiv preprint arXiv:2311.04145*, 2023.

[28] Y. Wang, X. Chen, X. Ma, S. Zhou, Z. Huang, Y. Wang, C. Yang, Y. He, J. Yu, P. Yang, *et al.*, "Lavie: High-quality video generation with cascaded latent diffusion models," *IJCV*, 2025.

[29] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *CVPR*, 2022.

[30] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI*, 2015.

[31] W. Peebles and S. Xie, "Scalable diffusion models with transformers," in *ICCV*, 2023.

[32] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017.

[33] T. Brooks, B. Peebles, C. Holmes, W. DePue, Y. Guo, L. Jing, D. Schnurr, J. Taylor, T. Luhman, E. Luhman, *et al.*, "Video generation models as world simulators," *OpenAI Blog*, 2024.

[34] W. Hong, M. Ding, W. Zheng, X. Liu, and J. Tang, "Cogvideo: Large-scale pretraining for text-to-video generation via transformers," in *ICLR*, 2023.

[35] W. Menapace, A. Siarohin, I. Skorokhodov, E. Deyneka, T.-S. Chen, A. Kag, Y. Fang, A. Stoliar, E. Ricci, J. Ren, *et al.*, "Snap video: Scaled spatiotemporal transformers for text-to-video synthesis," in *CVPR*, 2024.

[36] H. Lu, G. Yang, N. Fei, Y. Huo, Z. Lu, P. Luo, and M. Ding, "Vdt: General-purpose video diffusion transformers via mask modeling," in *ICLR*, 2024.

[37] X. Ma, Y. Wang, G. Jia, X. Chen, Z. Liu, Y.-F. Li, C. Chen, and Y. Qiao, "Latte: Latent diffusion transformer for video generation," *TMLR*, 2024.

[38] P. Gao, L. Zhuo, D. Liu, R. Du, X. Luo, L. Qiu, Y. Zhang, C. Lin, R. Huang, S. Geng, *et al.*, "Lumina-t2x: Transforming text into any modality, resolution, and duration via flow-based large diffusion transformers," *arXiv preprint arXiv:2405.05945*, 2024.

[39] A. Polyak, A. Zohar, A. Brown, A. Tjandra, A. Sinha, A. Lee, A. Vyas, B. Shi, C.-Y. Ma, C.-Y. Chuang, *et al.*, "Movie gen: A cast of media foundation models," *arXiv preprint arXiv:2410.13720*, 2024.

[40] A. Gupta, L. Yu, K. Sohn, X. Gu, M. Hahn, F.-F. Li, I. Essa, L. Jiang, and J. Lezama, "Photorealistic video generation with diffusion models," in *ECCV*, 2024.

[41] D. Kondratyuk, L. Yu, X. Gu, J. Lezama, J. Huang, G. Schindler, R. Hornung, V. Birodkar, J. Yan, M.-C. Chiu, *et al.*, "Videopoet: A large language model for zero-shot video generation," in *ICML*, 2024.

[42] J. Xing, L. Mai, C. Ham, J. Huang, A. Mahapatra, C.-W. Fu, T.-T. Wong, and F. Liu, "Motioncanvas: Cinematic shot design with controllable image-to-video generation," *arXiv preprint arXiv:2502.04299*, 2025.

[43] S. Liu, T. Wang, J.-H. Wang, Q. Liu, Z. Zhang, J.-Y. Lee, Y. Li, B. Yu, Z. Lin, S. Y. Kim, *et al.*, "Generative video propagation," *arXiv preprint arXiv:2412.19761*, 2024.

[44] X. He, Q. Liu, S. Qian, X. Wang, T. Hu, K. Cao, K. Yan, and J. Zhang, "Id-animator: Zero-shot identity-preserving human video generation," *arXiv preprint arXiv:2404.15275*, 2024.

[45] H. Ye, J. Zhang, S. Liu, X. Han, and W. Yang, "Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models," *arXiv preprint arXiv:2308.06721*, 2023.

[46] Y. Huang, Y. Qin, S. Lu, X. Wang, R. Huang, Y. Shan, and R. Zhang, "Story3d-agent: Exploring 3d storytelling visualization with large language models," *arXiv preprint arXiv:2408.11801*, 2024.

[47] T. Wu, Y. Zhang, X. Wang, X. Zhou, G. Zheng, Z. Qi, Y. Shan, and X. Li, "Customcrafter: Customized video generation with preserving motion and concept composition abilities," in *AAAI*, 2025.

[48] J. Materzyńska, J. Sivic, E. Shechtman, A. Torralba, R. Zhang, and B. Russell, "Newmove: Customizing text-to-video models with novel motions," in *ACCV*, 2024.

[49] Z. Ma, D. Zhou, C.-H. Yeh, X.-S. Wang, X. Li, H. Yang, Z. Dong, K. Keutzer, and J. Feng, "Magic-me: Identity-specific video customized diffusion," *arXiv preprint arXiv:2402.09368*, 2024.

[50] F. Long, Z. Qiu, T. Yao, and T. Mei, "Videodrafter: Content-consistent multi-scene video generation with llm," *CoRR*, 2024.

[51] Y. Fang, W. Menapace, A. Siarohin, T.-S. Chen, K.-C. Wang, I. Skorokhodov, G. Neubig, and S. Tulyakov, "Vimi: Grounding video generation through multi-modal instruction," in *EMNLP*, 2024.

[52] C.-P. Huang, Y.-S. Wu, H.-K. Chung, K.-P. Chang, F.-E. Yang, and Y.-C. F. Wang, "Videomage: Multi-subject and motion customization of text-to-video diffusion models," in *CVPR*, 2025.

[53] Y. Jiang, T. Wu, S. Yang, C. Si, D. Lin, Y. Qiao, C. C. Loy, and Z. Liu, "Videobooth: Diffusion-based video generation with image prompts," in *CVPR*, 2024.

[54] Z. Fei, D. Li, D. Qiu, J. Wang, Y. Dou, R. Wang, J. Xu, M. Fan, G. Chen, Y. Li, *et al.*, "Skyreels-a2: Compose anything in video diffusion transformers," *arXiv preprint arXiv:2504.02436*, 2025.

[55] Y. Zhou, D. Zhou, M.-M. Cheng, J. Feng, and Q. Hou, "Storydiffusion: Consistent self-attention for long-range image and video generation," in *NeurIPS*, 2024.

[56] D. J. Zhang, D. Li, H. Le, M. Z. Shou, C. Xiong, and D. Sahoo, "Moonshot: Towards controllable video generation and editing with multimodal conditions," *IJCV*, 2025.

[57] Z. Wang, A. Li, L. Zhu, Y. Guo, Q. Dou, and Z. Li, "Customvideo: Customizing text-to-video generation with multiple subjects," *arXiv preprint arXiv:2401.09962*, 2024.

[58] T.-S. Chen, C. H. Lin, H.-Y. Tseng, T.-Y. Lin, and M.-H. Yang, "Motion-conditioned diffusion model for controllable video synthesis," *arXiv preprint arXiv:2304.14404*, 2023.

[59] D. Kim, J. Zhang, W. Jin, S. Cho, Q. Dai, J. Park, and C. Luo, "Subject-driven video generation via disentangled identity and motion," *arXiv preprint arXiv:2504.17816*, 2025.

[60] Y. Zhou, R. Zhang, J. Gu, N. Zhao, J. Shi, and T. Sun, "Sugar: Subject-driven video customization in a zero-shot manner," *arXiv preprint arXiv:2412.10533*, 2024.

[61] Z. Peng, W. Wang, L. Dong, Y. Hao, S. Huang, S. Ma, and F. Wei, "Kosmos-2: Grounding multimodal large language models to the world," in *ICLR*, 2023.

[62] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, *et al.*, "Sam 2: Segment anything in images and videos," in *ICLR*, 2025.

[63] S. Chen, H. Guo, S. Zhu, F. Zhang, Z. Huang, J. Feng, and B. Kang, "Video depth anything: Consistent depth estimation for super-long videos," in *CVPR*, 2025.

[64] Y. Lipman, R. T. Chen, H. Ben-Hamu, M. Nickel, and M. Le, "Flow matching for generative modeling," in *ICLR*, 2023.

[65] X. Liu, X. Zhang, J. Ma, J. Peng, *et al.*, "Instaflow: One step is enough for high-quality diffusion-based text-to-image generation," in *ICLR*, 2024.

[66] Z. Wang, Z. Yuan, X. Wang, Y. Li, T. Chen, M. Xia, P. Luo, and Y. Shan, "Motionctrl: A unified and flexible motion controller for video generation," in *ACM SIGGRAPH*, 2024.

[67] H. He, Y. Xu, Y. Guo, G. Wetzstein, B. Dai, H. Li, and C. Yang, "Cameractrl: Enabling camera control for text-to-video generation," in *ICLR*, 2025.

[68] G. Zheng, T. Li, R. Jiang, Y. Lu, T. Wu, and X. Li, "Cami2v: Camera-controlled image-to-video diffusion model," *arXiv preprint arXiv:2410.15957*, 2024.

[69] M. Deitke, D. Schwenk, J. Salvador, L. Weihs, O. Michel, E. VanderBilt, L. Schmidt, K. Ehsani, A. Kembhavi, and A. Farhadi, "Objaverse: A universe of annotated 3d objects," in *CVPR*, 2023.

[70] C. Wei, Z. Xiong, W. Ren, X. Du, G. Zhang, and W. Chen, "Omniedit: Building image editing generalist models through specialist supervision," in *ICLR*, 2025.

[71] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *ICLR*, 2019.

[72] I. Loshchilov and F. Hutter, "Sgdr: Stochastic gradient descent with warm restarts," in *ICLR*, 2017.

[73] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *ICML*, 2022.

[74] N. Ruiz, Y. Li, V. Jampani, Y. Pritch, M. Rubinstein, and K. Aberman, "Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation," in *CVPR*, 2023.

[75] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *ICCV*, 2021.

[76] Z. Teed and J. Deng, "Raft: Recurrent all-pairs field transforms for optical flow," in *ECCV*, Springer, 2020.

[77] S. Xiao, Y. Wang, J. Zhou, H. Yuan, X. Xing, R. Yan, C. Li, S. Wang, T. Huang, and Z. Liu, "Omnigen: Unified image generation," *arXiv preprint arXiv:2409.11340*, 2024.

[78] S. Bahmani, I. Skorokhodov, V. Rong, G. Wetzstein, L. Guibas, P. Wonka, S. Tulyakov, J. J. Park, A. Tagliasacchi, and D. B. Lindell, "4d-fy: Text-to-4d generation using hybrid score distillation sampling," in *CVPR*, 2024.

[79] U. Singer, S. Sheynin, A. Polyak, O. Ashual, I. Makarov, F. Kokkinos, N. Goyal, A. Vedaldi, D. Parikh, J. Johnson, *et al.*, "Text-to-4d dynamic scene generation," in *ICML*, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Please refer to the abstract and introduction.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Please refer to Sec. 6 for details.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: his paper does not include any theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Please refer to Sec. 3 and Sec. 4.1 for the details of our method and experimental settings.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: Code and data are the assets of the company. We will apply for approval to release the code.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Please refer to Sec. 4.1 for details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Please refer to Sec. 4 for details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Please refer to Sec. 4.1 for details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics https://neurips.cc/public/EthicsGuidelines?

   Answer: [Yes]

   Justification: The research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Please refer to Sec. 7 for more details.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: This paper poses no such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We have carefully credited all previous works we used in the paper. The license and terms are properly respected.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: the paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLM is not an important component of our method. The declaration is not required.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.