EasyV2V: A High-quality Instruction-based Video Editing Framework

Supplementary Material

A. Text-2-Video prior preservation

We perform LoRA tuning with the base T2V weights frozen to maximize prior preservation. Compared to full finetuning, LoRA finetuning consistently achieves higher scores across benchmarks, including T2V tasks (VBench [18]). Interestingly, LoRA finetuning on our V2V dataset also **improves T2V performance**, as reflected by a higher VBench score.

Method	Base Model	Full-tuning	EasyV2V
V2V (VLM Score ↑)	-	4.67	**7.73**
VBench Total ↑	79.99	77.91	**81.26**

B. Performance on image editing

During training, we occasionally sampled from image-editing datasets and treated each image–edit pair as a single-frame video, rather than exclusively applying affine transformations to synthesize pseudo-videos.

As shown in Table I, we evaluated EasyV2V on a recent image-editing benchmark [62], interpreting each image as a single-frame video with a resolution of $1 \times 832 \times 480$. *Impressively, although EasyV2V was not specifically designed for image editing, it surpassed all baselines on nearly all subtasks and achieved performance comparable to leading closed-source and commercial systems.*

Notably, EasyV2V outperformed EditVerse by a margin of **0.54**. The model also demonstrated strong results on both **action** and **hybrid** editing tasks, underscoring the effectiveness of our unified data pipeline that jointly leverages image-editing datasets and video–caption datasets featuring human actions. Figure I presents representative qualitative results, with additional examples provided in our supplementary materials.

C. Model configuration

C.1. Mask conditioning

We compare our mask-conditioned editing performance with WanVACE [21]. Note that as WanVACE is not an instruction-based editing model, we restrict our comparison to the edit types it supports via edit masks: *inpainting*, *object removal*, and *object replacement*. We evaluate two types of edit masks: pixel-wise spatial masks (indicating the edit region per frame) and frame-wise temporal masks (indicating which frames should be modified). While WanVACE utilizes an additional branch with complex context activation and injection for mask conditioning, we demonstrate that encoding masks using the video VAE is simple yet effective, particularly for temporal control.

We investigate different mask conditioning strategies for EasyV2V:

- Video VAE, $Z_{\text{msk}} + Z_{\text{src}}$: After encoding the mask video into mask latent Z_{msk} using the video VAE, we perform token addition to inject the mask condition into the source video latent. This is the default strategy for EasyV2V in the main paper.
- Video VAE, $Z_{\text{msk}} + Z_{\text{tgt}}$: After encoding the mask video into mask latent Z_{msk} using the video VAE, we perform token addition to inject the mask condition into the noisy target video latent.
- Downsample, $Z_{\text{msk}} + Z_{\text{src}}$: We apply spatial and temporal average pooling to downsample the mask video from input resolution to latent resolution, then perform token addition to inject the mask condition into the source video latent.
- Video VAE, $\texttt{Seq\_Cat}(Z_{\text{src}}, Z_{\text{tgt}}, Z_{\text{msk}})$: After encoding the mask video into mask latent Z_{msk} using the video VAE, we perform sequence concatenation for all condition signals, including Z_{src}, Z_{tgt}, Z_{msk}, and Z_{ref}.

To evaluate spatial mask editing, we adopt the VLM prompt from the EditVerse evaluation protocol. To evaluate temporal mask editing, where the edit occurs after a certain timestamp, we modify the VLM prompt to incorporate edit timestamp awareness. Please refer to the last page for our complete VLM prompts.

As shown in Table II, EasyV2V achieves the best performance when using token addition to inject the mask condition into the source video latent. Both sequence concatenation and our token addition strategy outperform WanVACE on mask-conditioned video editing.

From the qualitative comparison in Figure II, we observe that WanVACE [21] has limited ability to generalize to diverse edit prompts and fails to adhere to both temporal and spatial masks in many cases compared to EasyV2V.

C.2. LoRA Rank

We employ LoRA [16] to mitigate divergence and study the effect of LoRA rank. We train the model for 20K steps to ablate the impact of LoRA rank on the EditVerse benchmark. We observe that performance improves as we increase the rank. We adopt a LoRA rank of 256, as performance begins to saturate between ranks 128 and 256. Table III reports the results for ranks 64, 128, and 256. Although the model with rank 256 performs best on most metrics, the gap between ranks 128 and 256 is marginal, and rank 64 is only slightly inferior. This supports our hypothesis that pre-trained video models serve as strong priors for video editing, and that a low-rank update is sufficient for

Table I. Category-wise image-editing performance on ImgEdit Bench [62]. Scores are derived from a vision–language model (VLM) based on *Prompt Compliance*, *Visual Naturalness/Seamlessness*, and *Physical/Detail Coherence* (higher is better). **Impressively, EasyV2V surpasses all baselines across most categories and approaches the performance of leading commercial systems**, despite not being specifically designed for image editing.

Method	Add	Adjust	Extract	Replace	Remove	Background	Style	Hybrid	Action	Overall ↑
MagicBrush	2.84	1.58	1.51	1.97	1.58	1.75	2.38	1.62	1.22	1.83
Instruct-P2P	2.45	1.83	1.44	2.01	1.50	1.44	3.55	1.20	1.46	1.88
ICVE	2.50	2.11	1.66	2.81	1.89	2.32	3.98	1.55	2.44	2.36
AnyEdit	3.18	2.95	1.88	2.47	2.23	2.24	2.85	1.56	2.65	2.45
UltraEdit	3.44	2.81	2.13	2.96	1.45	2.83	3.76	1.91	2.98	2.70
ICEdit	3.58	3.39	1.73	3.15	2.93	3.08	3.84	2.04	3.68	3.05
Step1X-Edit	3.88	3.14	1.76	3.40	2.41	3.16	4.63	2.64	2.52	3.06
UniWorld-V1	3.82	3.64	2.27	3.47	3.24	2.99	4.21	2.96	2.74	3.26
BAGEL	3.81	3.59	1.58	3.85	3.16	3.39	4.51	2.67	4.25	3.42
OmniGen2	3.57	3.06	1.77	3.74	3.20	3.57	4.81	2.52	4.68	3.44
Kontext-dev	3.83	3.65	2.27	4.45	3.17	3.98	4.55	3.35	4.29	3.71
EasyV2V (Ours)	4.46	4.18	1.80	3.86	3.70	4.33	4.57	4.04	4.68	3.96
Closed-Source Commercial Models and Concurrent works										
EditVerse	3.81	3.62	1.44	3.95	3.14	3.58	4.71	2.72	3.80	3.42
Ovis-U1	3.99	3.73	2.66	4.38	4.15	4.05	4.86	3.43	4.68	3.97
GPT-4o-Image	4.61	4.33	2.90	4.35	3.66	4.57	4.93	3.96	4.89	4.20
QwenImageEdit	4.68	4.42	4.26	4.54	4.54	4.66	4.76	4.38	4.85	4.57

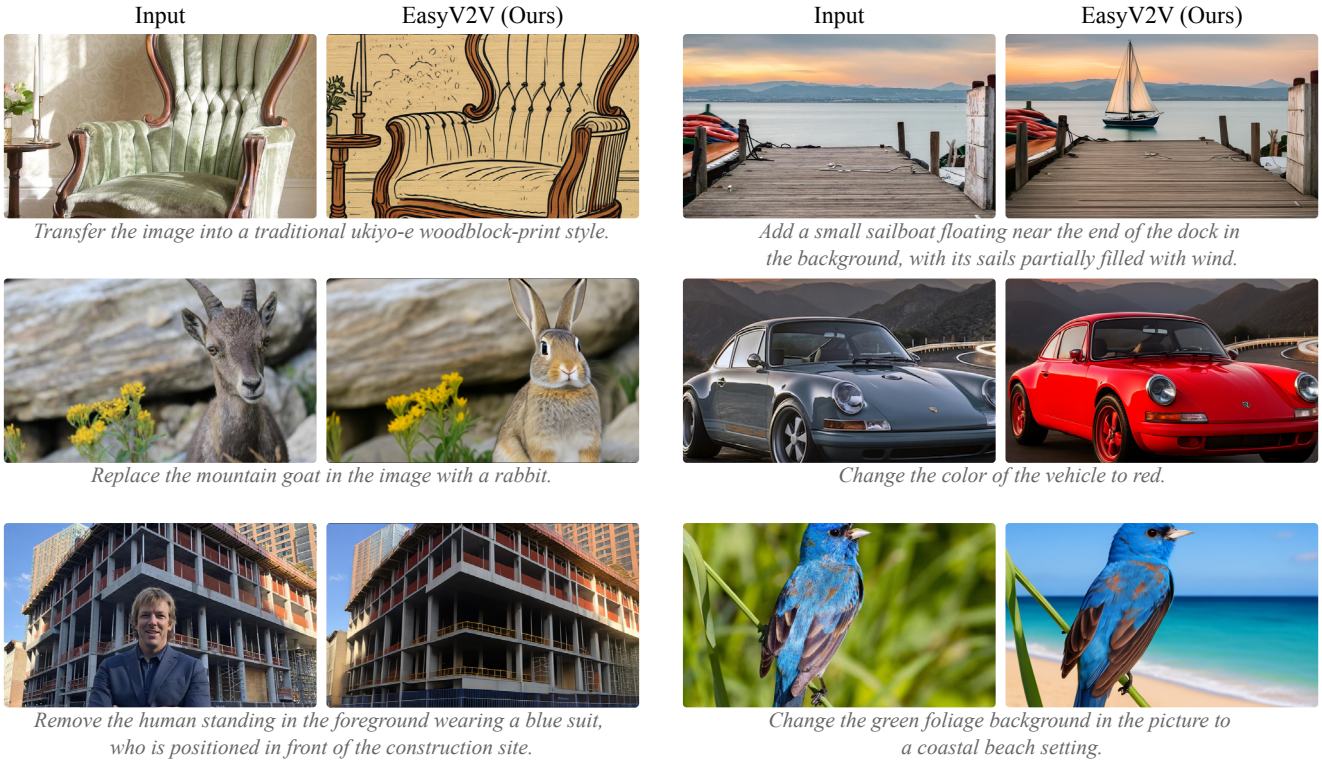

Figure I. Qualitative image-editing results from our video-editing model, which treats images as single-frame videos and achieves state-of-the-art performance, showing that video editing aids image editing.

robust performance.

(a) Edit with spatial mask. Edit prompt: *"remove the lamb."*

(b) Edit with temporal mask. Edit prompt: *"change the woman to a robot."*

Figure II. Comparison of mask-guided editing performance with WanVACE using samples from EditVerseBench [23].

Table II. Comparison of video editing mask strategies.

Method	**VLM score** (↑)		
	Pixel-wise spatial mask	**Frame-wise temporal mask**	**Average**
Wan VACE [21]	4.13	6.87	5.50
Video VAE, $Z_{\text{msk}} + Z_{\text{tgt}}$	5.50	7.40	6.45
Downsample, $Z_{\text{msk}} + Z_{\text{src}}$	5.80	5.40	5.60
Video VAE, $\texttt{Seq\_Cat}(Z_{\text{src}}, Z_{\text{tgt}}, Z_{\text{msk}})$	6.00	7.70	6.85
Video VAE, $Z_{\text{msk}} + Z_{\text{src}}$ (EasyV2V)	7.23	7.73	7.48

Table III. Ablation study on LoRA rank and the use of the reference image.

	Rank 64		Rank 128		Rank 256	
Metric	w/o Ref.	w/ Ref.	w/o Ref.	w/ Ref.	w/o Ref.	w/ Ref.
Frame-Text Alignment	24.80	27.27	24.99	**27.67**	25.32	**27.67**
Video-Text Alignment	21.31	24.36	21.45	**24.73**	21.90	24.60
PickScore Video Quality	19.42	20.20	19.52	20.36	19.58	**20.37**
VLM Quality Score	6.17	7.02	6.20	7.12	6.48	**7.22**

C.3. Time and memory profiling

We profile the efficiency of EasyV2V on a single NVIDIA H100 GPU for training and inference under different strategies: (1) full model fine-tuning with sequence concatenation of source and target latents, (2) full model fine-tuning with token addition of source and target latents, (3) LoRA fine-tuning with sequence concatenation of source and target latents, and (4) LoRA fine-tuning with token addition of source and target latents. FlashAttention is enabled during both training and inference. A complete comparison is provided in Table IV.

Table IV. Comparison of training and inference costs across different tuning strategies.

Metric	**Full Model w/ SeqCat.**	**Full Model w/ EmbedAdd.**	**LoRA w/ SeqCat.**	**LoRA w/ EmbedAdd.**
New Params	5B	5B	0.64B	0.64B
Train (s / batch)	5.63	4.60	5.70	4.54
Train VRAM	62 GB	62 GB	37 GB	32 GB
Inference (s / sample)	67.71	30.41	69.42	30.11

D. Additional details on data pipelines

Human animation. We use the 14B-parameter pretrained Wan Animate [8] model as the expert editor, following its preprocessing for face crops and poses. For first-frame editing, we apply Flux Kontext Dev [30]. Edit prompts are generated by ChatGPT [49] using 150 prompts created from the following instruction:

```
You are a helpful assistant to help with
↪ the generation of video editing
↪ prompts, to edit videos of people.
↪ Below are samples of the prompts we're
↪ interested it:

# Fantasy & Mythical Creatures
```

```
- "Make the person look like an elf"
- "Make the person look like a goblin"

# Professions & Archetypes
- "Make the person look like a knight"
- "Make the person look like a samurai"

# Horror & Dark Styles
- "Make the person look like a zombie"
- "Make the person look like a vampire"

# Animals & Hybrids
- "Make the person look like a lion"
- "Make the person look like a tiger"

# Sci-Fi & Futuristic
- "Make the person look like a robot"
- "Make the person look like a cyborg"

# Stylized & Surreal
- "Make the person look like a
↪  stained-glass figure"
- "Make the person look like a chalk
↪  drawing"

# Accessories
- "Give the person a pair of sunglasses"
- "Give the person a hat"

Now, generate several prompts per category
↪  mentioned above.
```

Actor transmutation.

We adapt FlowEdit [28], originally designed for image editing, on Wan 2.1 14B [50] for video editing. As described in the main paper, our prompt assets include three object categories: bipedals (e.g., clown, pirate, ninja, samurai, humanoid robot), quadrupeds (e.g., dog, cat, lion, cheetah, sheep), and avians (e.g., pigeon, duck, parrot, eagle, owl). We provide five examples per category to ChatGPT to generate extended lists. We also compile lists of scenes (e.g., jungle, mountain, beach, street, bedroom) and actions (e.g., walking, running, jumping, dancing).

Video stylization.

Similar to the prompts for our human animate dataset, we use ChatGPT to generate prompts for video stylization:

```
You are a helpful assistant to help with
↪  the generation of video stylization
↪  prompts, to edit and stylize in the
↪  wild videos. Below are samples of the
↪  prompts we're interested it:

# Distinct media / aesthetics
- "In the style of a watercolor painting"
- "In the style of a digital painting"

# Comic / cartoon & animation houses
- "In the style of a manga"
- "In the style of a cartoon"

# Art movements
- "In the style of Brutalism"
- "In the style of Impressionism"

# Lighting
- "Captured in the golden hour"
- "Captured bathed in neon lights"

# Cinematic & photographic framing styles
- "Shot on 35 mm film grain"
- "Shot in black-and-white film noir style"

# Global traditional arts & patterns
- "In the style of a Roman floor mosaic"
- "In the style of Persian miniature
↪  painting"

# Color palette & tonal approaches
- "Rendered in soft pastel hues"
- "Rendered in muted earth tones"

Now, generate several prompts per category
↪  mentioned above.
```

We end up generating 350 different stylization prompts.

Controllable video generation.

We curate a 15K-sample dataset from in-the-wild videos through manual filtering, then apply a range of model-free and model-based video-to-video transformations to build a paired dataset for training our video editing model. These transformations include human pose estimation (DWPose), Canny and HED edge detection, RAFT large optical flow, random black rectangle masks (inpainting), random black borders (outpainting), depth prediction with Depth Anything V2, grayscale conversion, Gaussian blur, color negation, saturation/contrast/brightness adjustments, pixelation, wave warping, posterization, Gaussian noise, and color overlays.

Dense-captioned text-to-video data

We apply strict filtering criteria to ensure high-quality training pairs from dense-captioned datasets. First, we require videos to have at least 162 frames after downsampling to 15 fps (enabling 81 frames for both source and target clips). Second, we filter temporal segments based on: (i) start time must allow sufficient preceding frames ($t_i \geq 81/\text{fps}_{\text{downsample}}$), (ii) segment duration must exceed 2 seconds to ensure meaningful actions, and (iii) no scene cuts within the segment interval. For instruction generation, we flatten all temporal segments from multiple videos into batches and process them simultaneously with Qwen-3-4B [58], discarding segments where the LLM returns empty strings (indicating unsuitable actions for video editing conversion). The LLM prompt instructs the model to convert

action descriptions into imperative instructions starting with verbs like "make," "let," or "have," while preserving all key details.

Image-to-image data

We employ a multi-stage pipeline to generate high-quality instructional I2I pairs from image captions at scale. Given an image caption, an LLM (Qwen3-4B-Instruct [58]) generates up to five diverse edit instructions spanning canonical edit types: `add`, `remove`, `replace`, `change_global`, `change_local`, `change_color`, `transform_global`, `transform_local`, `text`, and `other`. Each instruction is then executed using instruction-following image editors (Qwen-Image-Edit [38] , or Flux-Kontext [30]), producing candidate edit pairs. To ensure quality, we apply VLM-based filter with Gemma-3-27B [48].

For I2I-to-V2V conversion, we generate smooth affine camera trajectories by sampling random target poses with bounded parameters: rotation angles in $[-15°, 15°]$, zoom factors in $[0.66, 1.0]$ (avoiding excessive zoom-out), and translation offsets within $\pm 33\%$ of frame dimensions. These parameters are linearly interpolated across frames and constrained via linear programming to ensure the transformed bounding box remains fully within the frame boundaries throughout the trajectory. The same trajectory is applied to both source and target images using perspective transforms, creating temporally coherent pseudo-videos with motion cues (zoom, pan, rotation) while preserving the edit signal. With 50% probability, trajectories are reversed to balance zoom-in and zoom-out motions, providing diverse camera movement patterns that enhance the model's robustness to dynamic viewpoints during training.

E. Impact of dataset size and generalization to unseen edits

We split a subset of our V2V training data to include three edit types only and ablate on training data size with $10K$, $100K$, and $1M$ samples. As training data size increases, Figure III shows performance improves accordingly for both seen and unseen edits. We observe that training with only 10K examples already yields fair performance. Moreover, editing capability on seen tasks consistently enhances performance on unseen edit categories, validating that the inherent edit ability of a pretrained T2V model can be unlocked with our efficient tuning.

F. Impact of classifier free guidance

We perform an ablation study on the impact of classifier-free guidance (CFG) [15] during inference.

CFG Implementation. Following the standard CFG formulation, we guide the denoising process by interpolating between conditional and unconditional predictions. Given the predicted noise ϵ_θ from our diffusion model, the CFG-guided prediction is computed as:

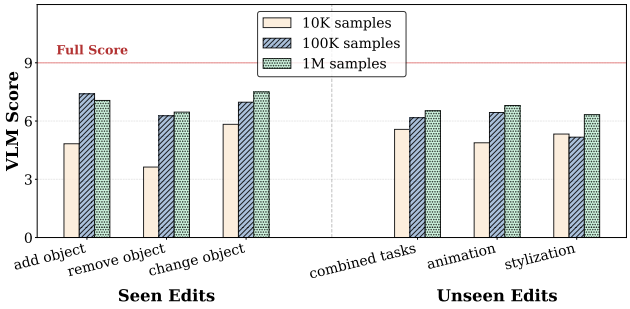

Figure III. Ablation study on training data size and generalization.

$$\hat{\epsilon}_\theta = \epsilon_\theta(\emptyset) + s \cdot (\epsilon_\theta(c) - \epsilon_\theta(\emptyset)) \tag{1}$$

where c represents the conditioning signal, $\emptyset$ denotes the unconditional (null) condition, and s is the guidance scale. This can be reformulated as:

$$\hat{\epsilon}_\theta = (1 - s) \cdot \epsilon_\theta(\emptyset) + s \cdot \epsilon_\theta(c) \tag{2}$$

When $s = 1.0$, the model performs purely conditional generation, while larger values of s amplify the influence of the conditioning signal.

In our framework, we support two CFG strategies depending on which conditions are used for guidance: *Prompt-only CFG* and *Prompt + Reference CFG*.

Prompt-only CFG. By default, we apply CFG only to the text prompt while keeping all visual conditions (source video and reference image) shared between conditional and unconditional branches:

$$\hat{\epsilon}_\theta = \epsilon_\theta(c_{\text{vis}}) + s \cdot (\epsilon_\theta(c_{\text{vis}}, c_{\text{prompt}}) - \epsilon_\theta(c_{\text{vis}})) \tag{3}$$

where c_{vis} denotes visual conditions (source video and optionally reference image), and c_{prompt} is the edit instruction. This approach maintains consistent visual context while allowing the text prompt to guide the editing direction.

Prompt + Reference CFG. Alternatively, we can apply CFG to both the text prompt and reference image:

$$\hat{\epsilon}_\theta = \epsilon_\theta(c_{\text{src}}) + s \cdot (\epsilon_\theta(c_{\text{src}}, c_{\text{ref}}, c_{\text{prompt}}) - \epsilon_\theta(c_{\text{src}})) \tag{4}$$

where c_{src} is the source video (always present), c_{ref} is the reference image, and c_{prompt} is the text prompt. This strategy applies guidance to both the reference appearance and text instruction simultaneously. Our method is uniquely suitable for this type of guidance because the reference image is an optional input to our model.

Experimental results. In Table V, we present results under different CFG scales when using only the edit prompt

Table V. Effect of the CFG scale for the edit prompt.

	Inference w/o Ref.				Inference w/ Ref.			
CFG scale	**1.0**	**3.0**	**5.0**	**7.0**	**1.0**	**3.0**	**5.0**	**7.0**
Frame-Text Alignment	26.65	27.59	27.49	27.11	27.26	27.28	27.33	27.29
Video-Text Alignment	24.52	24.46	24.16	23.40	24.01	24.07	24.27	24.33
PickScore Video Quality	20.05	20.36	20.23	20.05	20.57	20.60	20.58	20.53
VLM Quality Score	6.79	7.73	7.48	6.98	7.14	7.30	7.28	7.21

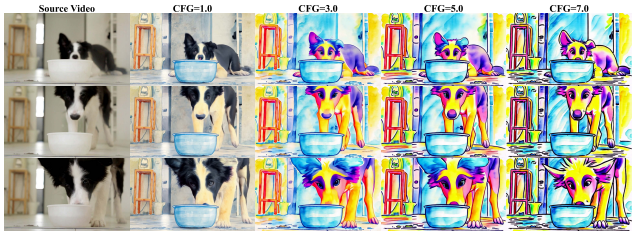

Figure IV. Effect of the CFG for the edit prompt. Edit prompt: *"Transform the entire visual style of the video using a hand-drawn watercolor animation effect."*

Table VI. Effect of the CFG scale for the edit prompt and reference image.

	Inference w/ Ref.			
CFG scale	**1.0**	**3.0**	**5.0**	**7.0**
Frame-Text Alignment	26.65	27.77	27.49	27.11
Video-Text Alignment	24.52	24.54	24.16	23.40
PickScore Video Quality	20.05	20.34	20.23	20.05
VLM Quality Score	6.79	7.69	7.47	6.98

for CFG. We also provide a visualized example in Figure IV. As shown, higher CFG scales generally improve text alignment but may reduce temporal consistency and video quality when the scale becomes too large.

In Table VI, we show results under different CFG scales when using both the edit prompt and reference image for CFG. For fair comparison, we report inference performance only when a reference image is provided. A visualized example is shown in Figure V. The dual-condition CFG provides stronger control over both appearance and semantic alignment but requires careful tuning of the guidance scale to balance faithfulness to conditions and generation quality.

We observe that the best performance is achieved when CFG scales are between 3.0 and 5.0. We adopt a moderate CFG scale of $s = 3.0$ by default with prompt-only guidance. We believe that further fine-grained CFG hyperparameter tuning could yield even better performance.

G. User study

We construct a custom benchmark comprising 160 horizontal and vertical videos spanning 18 distinct edit types, including *actor transmutation, add effect, add object, animation, change action, change background, change color, change material, change object, change weather, complex edit, control video, edit with mask, freeform, local stylization, global stylization, remove object, and sim2real*. We conduct a user study in which participants select the superior sample between outputs generated by two different methods, evaluating them along three dimensions: Instruction Alignment (adherence to the text prompt), Preservation of Unedited Regions (temporal consistency in unchanged areas), and Video Quality (overall visual fidelity). As shown in Figure VI, EasyV2V outperforms all other methods across all three dimensions.

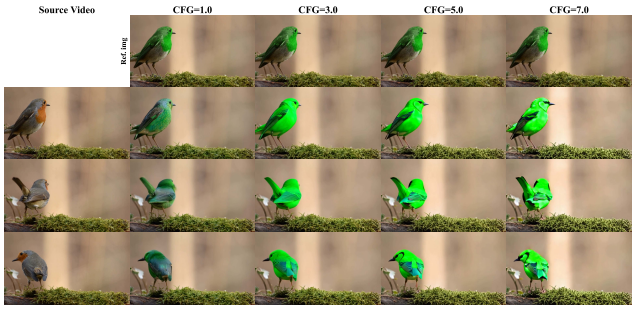

Figure V. Visual illustration of the effect of the CFG scale for the edit prompt and reference image. Edit prompt: *"Change the bird's color to emerald green."* The top row shows the reference image generated by Qwen-Image-Edit.

H. Additional visualization

H.1. More comparisons

We nonetheless bring more comparisons with Ins-ViE [55], ICVE [31], Runway Aleph [42] and Editto [2]. We also found that EasyV2V (**7.73**) qualitatively outperforms Editto (**5.00**) on EditVerseBench.

H.2. Robustness to Reference Image

We provide visualization results of EasyV2V based on the choice of reference image. By default, we derive the reference image by applying the image editing model to the first frame of the source video. We also present results using the first, middle, or last frame of the source video as the basis for the reference. As shown in Figure VIII, our model is robust to the choice of reference image, indicating that EasyV2V effectively captures the identity of the reference image for video editing. Moreover, EasyV2V can rectify inconsistent zoom-in effects and human pose misalignments introduced by the image editing model. Notably, without an external reference image from the image editing model, EasyV2V achieves even better consistency with the source video; for instance, the background remains well preserved.

H.3. Comparison on Human Animate and Flow Edit Datasets

We curate the Human Animate dataset, which contains human-centric video edits, and the Flow Edit dataset, which

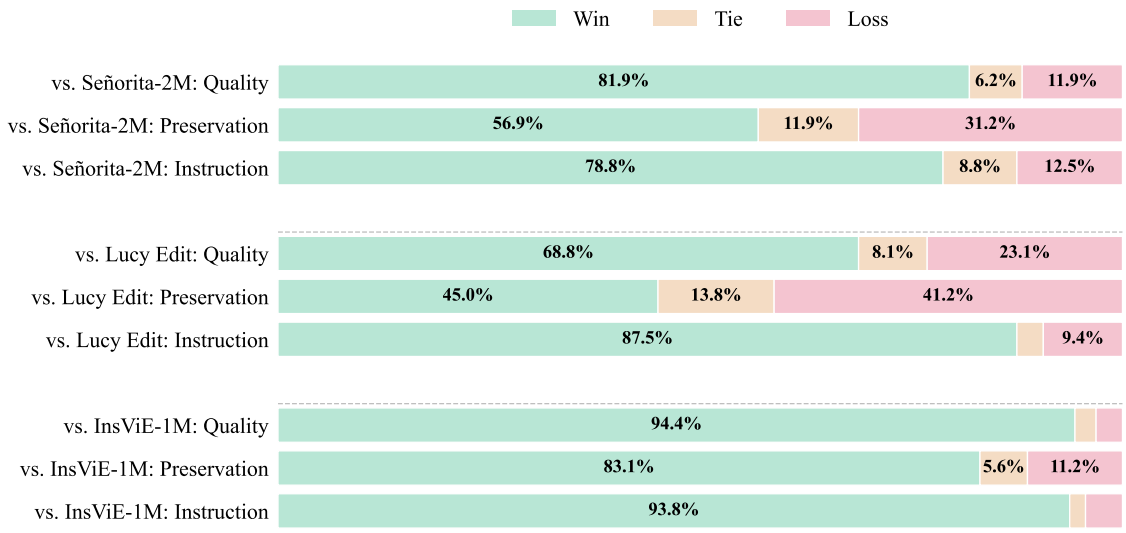

Figure VI. Results of the user study. EasyV2V is the most preferred method across all evaluation criteria.

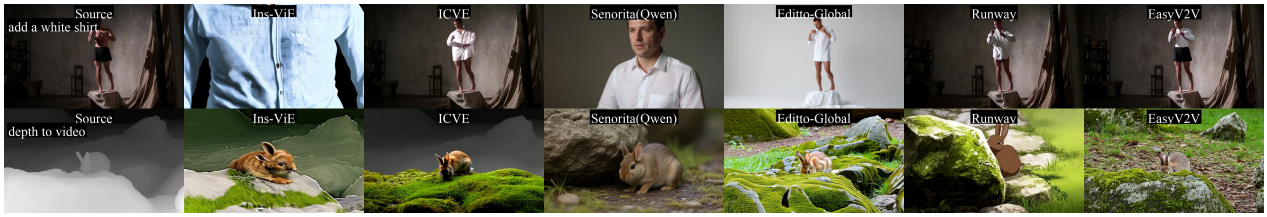

Figure VII. Comparison with more methods

focuses on actor transmutation edits. We provide comparisons in Figure IX between a model trained exclusively on the Human Animate dataset and one trained on the Flow Edit dataset. The model trained on the Human Animate dataset often achieves superior visual details, preserves poses more effectively, and generalizes better to unseen human-specific pose-to-video tasks. Facial expressions are also better preserved when training on the Human Animate dataset compared to the Flow Edit (actor transmutation) dataset.

H.4. Exhibition Gallery and Visualization

We present additional video editing examples on our visualization website included in the supplementary material. We strongly encourage readers to view the attached HTML file for comprehensive and diverse video results.

Capability for Human Action Editing. Leveraging a densely captioned video dataset during training, our model demonstrates a strong capability to follow text instructions for modifying human actions. Compared to concurrent works, which often struggle to alter human actions effectively, EasyV2V exhibits a **unique** proficiency in such edits, highlighting the effectiveness of our curated human action data.

Natural Edit Transitions. Although we employ a simple blending transition strategy during training, EasyV2V exhibits an emergent ability to produce natural transitions. We evaluate this using frame-wise temporal masks where the edit is restricted to the second half of the video. We observe smooth and realistic transitions at the timestamp where the source-to-target edit initiates.

High-Resolution Results. We train our model at a higher resolution of $81 \times 1280 \times 704$ to further validate our method and data pipeline. The total training data is subsampled to approximately 6 million samples due to the computational cost of high-resolution training. Note that $81 \times 1280 \times 704$ is the maximum supported resolution of Wan-2.2-TI2V-5B [50]. We observe rapid convergence within a few training steps, confirming that our architecture design with low-rank tuning efficiently adapts a T2V model to V2V tasks. To the best of our knowledge, EasyV2V is the first instruction-based video editing model capable of editing $\sim$720P videos with a duration of 81 frames.

Figure VIII. Robustness of EasyV2V to the choice of reference image. Edit prompt: *"Change the apron and blouse to a classic clown costume."* The top row shows reference images generated by Qwen-Image-Edit based on different frames from the source video, followed by videos generated by EasyV2V conditioned on these reference images. The source video is taken from Lucy Edit's website [46].

I. VLM prompts for evaluation

VLM prompt we used for spatial mask editing evaluation:

```
'You are a meticulous video editing quality
 ↪  evaluator. Your task is to provide a
 ↪  detailed assessment of a video edit by
 ↪  comparing the original image with the
 ↪  edited image based on a given text
 ↪  prompt.\n\
Editing Prompt:\n{editing_prompt}\n\
Instructions:\n\
Analyze the provided image (the edited
 ↪  video frame) and evaluate how well the
 ↪  "Editing Prompt" has been executed. You
 ↪  will evaluate the edit across three
 ↪  distinct criteria. For each criterion,
 ↪  provide a score from 0 (worst) to 3
 ↪  (best) and a brief justification.
 ↪  Finally, provide the total score.\n\
Your evaluation should focus on three key
 ↪  aspects:\n\
1. Prompt Following (Score: 0-3) \n\
Question: Does the edit accurately and
 ↪  completely fulfill the instructions in
 ↪  the "Editing Prompt"? \n\
Scoring Guide:\n\
- 3: The prompt is perfectly and completely
 ↪  followed.\n\
- 2: The prompt is mostly followed but with
 ↪  minor inaccuracies or omissions.\n\
- 1: The prompt is poorly followed or only
 ↪  partially executed.\n\
- 0: The prompt is completely ignored or
 ↪  the opposite was done. \n\
2. Edit Quality (Score: 0-3) \n\
Question: How is the visual quality of the
 ↪  edited area itself? Is it realistic,
 ↪  seamless, and free of artifacts (e.g.,
 ↪  blurriness, distortion, unnatural
 ↪  textures)?\n\
Scoring Guide:\n\
- 3: The edit is of high visual quality,
 ↪  seamless, and artifact-free.\n\
- 2: The edit is good but has minor,
 ↪  noticeable artifacts.\n\
- 1: The edit is of low quality with
 ↪  significant, distracting artifacts.\n\
- 0: The edited area is extremely poor,
 ↪  garbled, or has completely failed.\n\
3. Background Consistency (Score: 0-3) \n\
Question: Have the areas that should not
 ↪  have been edited remained unchanged
 ↪  between the "Before" and "After"
 ↪  images? \n\
Scoring Guide:\n\
- 3: The areas that should not have been
 ↪  edited are perfectly preserved and
 ↪  stable. \n\
- 2: There are minor, subtle, but
 ↪  noticeable changes or flickers in the
 ↪  areas that should not have been
 ↪  edited.\n\
- 1: There are significant and distracting
 ↪  changes in the areas that should not
 ↪  have been edited. \n\
- 0: The areas that should not have been
 ↪  edited is completely or
 ↪  catastrophically altered. \n\
```

VLM prompt we used for temporal mask editing evaluation:

Figure IX. Comparison between models trained on the Flow Edit dataset and the human animate dataset.

```
You are a meticulous video editing quality
↪  evaluator. Your task is to provide a
↪  detailed assessment of a temporal video
↪  edit by comparing the original image
↪  with the edited image based on a given
↪  text prompt.

Editing Prompt:
{editing_prompt}

IMPORTANT TEMPORAL CONTEXT:
This frame is from the FIRST HALF of the
↪  video, where the edit SHOULD NOT have
↪  occurred yet. The frame should remain
↪  unchanged from the original.

....
Your evaluation should focus on three key
↪  aspects:

1. Temporal Consistency (Score: 0-3)
Question: Does the frame correctly show NO
↪  editing in the first half of the video?
Scoring Guide:
- 3: The frame is perfectly unchanged,
↪  showing no signs of the edit that
↪  should only appear in the second half.
- 2: The frame is mostly unchanged but
↪  shows very minor, subtle hints of the
↪  edit.
- 1: The frame shows partial editing when
↪  it should be unchanged.
- 0: The frame is fully edited when it
↪  should be completely unchanged.

#############
IMPORTANT TEMPORAL CONTEXT:
This frame is from the TRANSITION PERIOD
↪  (around the middle of the video), where
↪  the edit IS HAPPENING. The frame should
↪  show the edit in progress or just
↪  completed.
```

```
Instructions:
Analyze the provided images and evaluate how well the temporal editing is progressing. You will evaluate the edit across three distinct criteria. For each criterion, provide a score from 0 (worst) to 3 (best) and a brief justification. Finally, provide the total score.

Your evaluation should focus on three key aspects:

1. Edit Progress (Score: 0-3)
Question: Does the frame show appropriate edit progression during this transition period?
Scoring Guide:
- 3: The frame shows natural, smooth editing transition that aligns with the temporal position.
- 2: The frame shows editing but the transition is slightly abrupt or unnatural.
- 1: The frame shows poor editing progression or timing.
- 0: The frame shows no editing or completely wrong timing.

....
IMPORTANT TEMPORAL CONTEXT:
This frame is from the SECOND HALF of the video, where the edit SHOULD HAVE been completed. The frame should show the fully edited result.

Instructions:
Analyze the provided images and evaluate how well the "Editing Prompt" has been executed. You will evaluate the edit across three distinct criteria. For each criterion, provide a score from 0 (worst) to 3 (best) and a brief justification. Finally, provide the total score.

Your evaluation should focus on three key aspects:

1. Prompt Following (Score: 0-3)
Question: Does the edit accurately and completely fulfill the instructions in the "Editing Prompt"?
Scoring Guide:
- 3: The prompt is perfectly and completely followed.
- 2: The prompt is mostly followed but with minor inaccuracies or omissions.
- 1: The prompt is poorly followed or only partially executed.
- 0: The prompt is completely ignored or the opposite was done.
#############
```

References

[1] VideoX-Fun: A more flexible framework that can generate videos at any resolution and creates videos from images. https://github.com/aigc-apps/VideoX-Fun, 2025. 1, 5

[2] Qingyan Bai, Qiuyu Wang, Hao Ouyang, Yue Yu, Hanlin Wang, Wen Wang, Ka Leong Cheng, Shuailei Ma, Yanhong Zeng, Zichen Liu, Yinghao Xu, Yujun Shen, and Qifeng Chen. Scaling instruction-based video editing with a high-quality synthetic dataset. *arXiv*, 2025. 3, 4, 8, 15

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report, 2025. 5

[4] Tim Brooks, Aleksander Holynski, and Alexei A Efros. InstructPix2Pix: Learning to follow image editing instructions. *CVPR*, 2023. 3

[5] Tim Brooks, Bill Peebles, Connor Holmes, Will DePue, Yufei Guo, Li Jing, David Schnurr, Joe Taylor, Troy Luhman, Eric Luhman, Clarence Ng, Ricky Wang, and Aditya Ramesh. Video generation models as world simulators. 2024. 3

[6] Qi Cai, Jingwen Chen, Yang Chen, Yehao Li, Fuchen Long, Yingwei Pan, Zhaofan Qiu, Yiheng Zhang, Fengbin Gao, Peihan Xu, et al. HiDream-I1: A high-efficient image generative foundation model with sparse diffusion transformer. *arXiv*, 2025. 3

[7] Mingdeng Cao, Xintao Wang, Zhongang Qi, Ying Shan, Xiaohu Qie, and Yinqiang Zheng. MasaCtrl: Tuning-free mutual self-attention control for consistent image synthesis and editing. In *ICCV*, 2023. 3

[8] Gang Cheng, Xin Gao, Li Hu, Siqi Hu, Mingyang Huang, Chaonan Ji, Ju Li, Dechao Meng, Jinwei Qi, Penchong Qiao, Zhen Shen, Yafei Song, Ke Sun, Linrui Tian, Feng Wang, Guangyuan Wang, Qi Wang, Zhongjian Wang, Jiayu Xiao, Sheng Xu, Bang Zhang, Peng Zhang, Xindi Zhang, Zhe Zhang, Jingren Zhou, and Lian Zhuo. Wan-Animate: Unified character animation and replacement with holistic replication, 2025. 1, 3, 5, 12

[9] Jiaxin Cheng, Tianjun Xiao, and Tong He. Consistent video-to-video transfer using synthetic dataset. *ICLR*, 2024. 1, 2, 3, 7

[10] Paul Couairon, Clément Rambour, Jean-Emmanuel HAUGEARD, and Nicolas THOME. VidEdit: Zero-shot and spatially aware text-driven video editing. *TMLR*, 2024. 1, 3

[11] Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, and Feng Li. Emerging properties in unified multimodal pretraining: The BAGEL model. *arXiv*, 2025. 3

[12] Yu Gao, Lixue Gong, Qiushan Guo, Xiaoxia Hou, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, Wei Liu, Yichun Shi, Shiqi Sun, Yu Tian, Zhi Tian, Peng Wang, Rui Wang, Xuanda Wang, Xun Wang, Ye Wang, Guofeng Wu, Jie Wu, Xin Xia, Xuefeng Xiao, Zhonghua Zhai, Xinyu Zhang, Qi Zhang, Yuwei Zhang, Shijia Zhao, Jianchao Yang, and Weilin Huang. Seedream 3.0 technical report, 2025. 3

[13] Yuying Ge, Sijie Zhao, Chen Li, Yixiao Ge, and Ying Shan. SEED-Data-Edit technical report: A hybrid dataset for instructional image editing. *arXiv*, 2024. 3

[14] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv*, 2022. 3

[15] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 14

[16] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *ICLR*, 2022. 2, 10

[17] Jiahao Hu, Tianxiong Zhong, Xuebo Wang, Boyuan Jiang, Xingye Tian, Fei Yang, Pengfei Wan, and Di Zhang. VIVID-10M: A dataset and baseline for versatile and interactive video local editing, 2025. 3

[18] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024. 10

[19] Mude Hui, Siwei Yang, Bingchen Zhao, Yichun Shi, Heng Wang, Peng Wang, Yuyin Zhou, and Cihang Xie. HQ-Edit: A high-quality dataset for instruction-based image editing. *arXiv*, 2024. 3, 6

[20] Imagen-Team-Google. Imagen 3, 2024. 3

[21] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. VACE: All-in-one video creation and editing. In *ICCV*, 2025. 1, 3, 5, 10, 12

[22] Xuan Ju, Ailing Zeng, Yuxuan Bian, Shaoteng Liu, and Qiang Xu. PnP inversion: Boosting diffusion-based editing with 3 lines of code. *ICLR*, 2024. 3

[23] Xuan Ju, Tianyu Wang, Yuqian Zhou, He Zhang, Qing Liu, Nanxuan Zhao, Zhifei Zhang, Yijun Li, Yuanhao Cai, Shaoteng Liu, Daniil Pakhomov, Zhe Lin, Soo Ye Kim, and Qiang Xu. EditVerse: Unifying image and video editing and generation with In-Context learning, 2025. 3, 6, 7, 12

[24] Xuan Ju, Weicai Ye, Quande Liu, Qiulin Wang, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, and Qiang Xu. FullDiT: Multi-task video generative foundation model with full attention, 2025. 3

[25] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *Advances in neural information processing systems*, 36:36652–36663, 2023. 6

[26] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv*, 2024. 3

[27] Max Ku, Cong Wei, Weiming Ren, Harry Yang, and Wenhu Chen. AnyV2V: A tuning-free framework for any video-to-video editing tasks. *TMLR*, 2024. 1, 3

[28] Vladimir Kulikov, Matan Kleiner, Inbar Huberman-Spiegelglas, and Tomer Michaeli. FlowEdit: Inversion-free text-based editing using pre-trained flow models. *ICCV*, 2025. 3, 5, 13

[29] Black Forest Labs. FLUX. https://github.com/black-forest-labs/flux, 2024. 3

[30] Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, and Luke Smith. FLUX.1 Kontext: Flow matching for in-context image generation and editing in latent space, 2025. 3, 5, 6, 12, 14

[31] Xinyao Liao, Xianfang Zeng, Ziye Song, Zhoujie Fu, Gang Yu, and Guosheng Lin. In-context learning with unpaired clips for instruction-based video editing. *arXiv preprint arXiv:2510.14648*, 2025. 15

[32] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-P2P: Video editing with cross-attention control. *CVPR*, 2024. 3

[33] Shaoteng Liu, Tianyu Wang, Jui-Hsien Wang, Qing Liu, Zhifei Zhang, Joon-Young Lee, Yijun Li, Bei Yu, Zhe Lin, Soo Ye Kim, and Jiaya Jia. Generative video propagation. *CVPR*, 2025. 3

[34] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[35] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. FateZero: Fusing attentions for zero-shot text-based video editing. *ICCV*, 2023. 3

[36] Bosheng Qin, Juncheng Li, Siliang Tang, Tat-Seng Chua, and Yueting Zhuang. InstructVid2Vid: Controllable video editing with natural language instructions. In *ICME*, 2024. 3

[37] Liao Qu, Huichao Zhang, Yiheng Liu, Xu Wang, Yi Jiang, Yiming Gao, Hu Ye, Daniel K. Du, Zehuan Yuan, and Xinglong Wu. Tokenflow: Unified image tokenizer for multimodal understanding and generation. In *CVPR*, 2025. 7

[38] Qwen Team. Qwen-Image: A unified foundation model for image generation and editing. *arXiv*, 2025. 4, 6, 14

[39] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 6

[40] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-

Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment anything in images and videos. *arXiv*, 2024. 5

[41] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 3

[42] Runway. Introducing runway aleph. https://runwayml.com/research/introducing-runway-aleph. 7, 15

[43] Team Seedream, :, Yunpeng Chen, Yu Gao, Lixue Gong, Meng Guo, Qiushan Guo, Zhiyao Guo, Xiaoxia Hou, Weilin Huang, Yixuan Huang, Xiaowen Jian, Huafeng Kuang, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, Wei Liu, Yanzuo Lu, Zhengxiong Luo, Tongtong Ou, Guang Shi, Yichun Shi, Shiqi Sun, Yu Tian, Zhi Tian, Peng Wang, Rui Wang, Xun Wang, Ye Wang, Guofeng Wu, Jie Wu, Wenxu Wu, Yonghui Wu, Xin Xia, Xuefeng Xiao, Shuang Xu, Xin Yan, Ceyuan Yang, Jianchao Yang, Zhonghua Zhai, Chenlin Zhang, Heng Zhang, Qi Zhang, Xinyu Zhang, Yuwei Zhang, Shijia Zhao, Wenliang Zhao, and Wenjia Zhu. Seedream 4.0: Toward next-generation multimodal image generation, 2025. 3

[44] Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. Emu edit: Precise image editing via recognition and generation tasks. In *CVPR*, 2023. 3

[45] Uriel Singer, Amit Zohar, Yuval Kirstain, Shelly Sheynin, Adam Polyak, Devi Parikh, and Yaniv Taigman. EVE: Video editing via factorized diffusion distillation. *ECCV*, 2024. 3

[46] Decart Team. Lucy edit: Open-weight text-guided video editing. https://d2drjpuinn46lb.cloudfront.net/Lucy_Edit__High_Fidelity_Text_Guided_Video_Editing.pdf, 2025. 3, 7, 17

[47] Genmo Team. Mochi 1. https://github.com/genmoai/models, 2024. 3

[48] Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*, 2025. 14

[49] OpenAI Team. GPT-4o system card, 2024. 3, 6, 12

[50] Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models, 2025. 3, 5, 6, 13, 16

[51] Wen Wang, Yan Jiang, Kangyang Xie, Zide Liu, Hao Chen, Yue Cao, Xinlong Wang, and Chunhua Shen. Zero-shot video editing using off-the-shelf image diffusion models. *arXiv*, 2023. 3

[52] Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. Internvid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023. 6

[53] Yuhan Wang, Siwei Yang, Bingchen Zhao, Letian Zhang, Qing Liu, Yuyin Zhou, and Cihang Xie. GPT-IMAGE-EDIT-1.5M: A million-scale, GPT-generated image dataset, 2025. 3, 4, 6

[54] Cong Wei, Zheyang Xiong, Weiming Ren, Xinrun Du, Ge Zhang, and Wenhu Chen. OmniEdit: Building image editing generalist models through specialist supervision. In *ICLR*, 2025. 3, 6

[55] Yuhui Wu, Liyi Chen, Ruibin Li, Shihao Wang, Chenxi Xie, and Lei Zhang. InsViE-1M: Effective instruction-based video editing with elaborate dataset construction. *ICCV*, 2025. 1, 2, 3, 6, 7, 15

[56] Ziyi Wu, Aliaksandr Siarohin, Willi Menapace, Ivan Skorokhodov, Yuwei Fang, Varnith Chordia, Igor Gilitschenski, and Sergey Tulyakov. Mind the time: Temporally-controlled multi-event video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 23989–24000, 2025. 6

[57] Shitao Xiao, Yueze Wang, Junjie Zhou, Huaying Yuan, Xingrun Xing, Ruiran Yan, Chaofan Li, Shuting Wang, Tiejun Huang, and Zheng Liu. OmniGen: Unified image generation. In *CVPR*, 2025. 3

[58] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv*, 2025. 6, 13, 14

[59] Ling Yang, Bohan Zeng, Jiaming Liu, Hong Li, Minghao Xu, Wentao Zhang, and Shuicheng Yan. EditWorld: Simulating world dynamics for instruction-following image editing. *arXiv*, 2024. 3

[60] Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. CogVideoX: Text-to-video diffusion models with an expert transformer. *ICLR*, 2025. 3

[61] Danah Yatim, Rafail Fridman, Omer Bar-Tal, Yoni Kasten, and Tali Dekel. Space-time diffusion features for zero-shot text-driven motion transfer. In *CVPR*, 2024. 7

[62] Yang Ye, Xianyi He, Zongjian Li, Bin Lin, Shenghai Yuan, Zhiyuan Yan, Bohan Hou, and Li Yuan. Imgedit: A unified image editing dataset and benchmark. *arXiv preprint arXiv:2505.20275*, 2025. 7, 8, 10, 11

[63] Zixuan Ye, Xuanhua He, Quande Liu, Qiulin Wang, Xintao Wang, Pengfei Wan, Di Zhang, Kun Gai, Qifeng Chen, and Wenhan Luo. UNIC: Unified in-context video editing, 2025. 3

[64] Jaehong Yoon, Shoubin Yu, and Mohit Bansal. RACCooN: A versatile instructional video editing framework with auto-generated narratives. *EMNLP*, 2025. 3

[65] Qifan Yu, Wei Chow, Zhongqi Yue, Kaihang Pan, Yang Wu, Xiaoyang Wan, Juncheng Li, Siliang Tang, Hanwang Zhang, and Yueting Zhuang. AnyEdit: Mastering unified high-quality image editing for any idea. In *CVPR*, 2025. 3

[66] Shoubin Yu, Difan Liu, Ziqiao Ma, Yicong Hong, Yang Zhou, Hao Tan, Joyce Chai, and Mohit Bansal. VEGGIE: Instructional editing and reasoning video concepts with grounded generation. *ICCV*, 2025. 1, 2, 3

[67] Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. MagicBrush: A manually annotated dataset for instruction-guided image editing. In *NeurIPS*, 2023. 3

[68] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023. 3

[69] Youcai Zhang, Xinyu Huang, Jinyu Ma, Zhaoyang Li, Zhaochuan Luo, Yanchun Xie, Yuzhuo Qin, Tong Luo, Yaqian Li, Shilong Liu, et al. Recognize anything: A strong image tagging model. *arXiv*, 2023. 5

[70] Zhenghao Zhang, Zuozhuo Dai, Long Qin, and Weizhi Wang. EffiVED: Efficient video editing via text-instruction diffusion models, 2024. 3

[71] Zechuan Zhang, Ji Xie, Yu Lu, Zongxin Yang, and Yi Yang. In-Context Edit: Enabling instructional image editing with in-context generation in large-scale diffusion transformers. *NeurIPS*, 2025. 3

[72] Haozhe Zhao, Xiaojian Ma, Liang Chen, Shuzheng Si, Rujie Wu, Kaikai An, Peiyu Yu, Minjia Zhang, Qing Li, and Baobao Chang. UltraEdit: Instruction-based fine-grained image editing at scale. In *NeurIPS*, 2024. 3, 6

[73] Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model, 2024. 3

[74] Bojia Zi, Weixuan Peng, Xianbiao Qi, Jianan Wang, Shihao Zhao, Rong Xiao, and Kam-Fai Wong. MiniMax-Remover: Taming bad noise helps video object removal. *arXiv*, 2025. 1, 5

[75] Bojia Zi, Penghui Ruan, Marco Chen, Xianbiao Qi, Shaozhe Hao, Shihao Zhao, Youze Huang, Bin Liang, Rong Xiao, and Kam-Fai Wong. Señorita-2M: A high-quality instruction-based dataset for general video editing by video specialists. In *NeurIPS*, 2025. 1, 2, 3, 4, 6, 7, 8