InsViE-1M: Effective Instruction-based Video Editing with Elaborate Dataset Construction

Supplementary Material

In this supplementary file, we provide additional details of the construction pipeline of our InsViE-1M dataset in Sec. 1, additional settings of model training and testing in Sec. 2, more visual comparisons in Sec. 3, and more ablation studies in Sec. 4. In addition, we provide a demo video that includes more visual comparisons. Please view the video using software that can open MOV files.

1. Details of InsViE-1M Dataset

In this section, we first show the specific prompts used for generating instruction and filtering in Sec. 1.1 and Sec. 1.2, respectively. Then we illustrate the case study on CFG in Sec. 1.4. Finally, we present examples of the data construction process in Sec. 1.3.

1.1. Prompts for Recaptioning and Instruction Generation

The original video dataset provides initial video captions that outline the overall content of the videos. However, these captions are often either too long, containing excessive details, or too brief, consisting of only a few words. As a result, they are not suitable for generating effective instructions. Therefore, as shown in Fig. 1, we propose a systematic approach to generate video captions and the corresponding editing instructions by a large vision-language model [4]. The process begins with extracting three key frames from the source video to capture important moments. Based on the initial captions, the system generates supplementary descriptions for each key frame, capturing the actions and nuances within the frames. This ensures a coherent narrative that aligns with the initial caption while highlighting the key elements of the scene. Then we use the initial caption and the generated key frame captions to produce the final video caption. Finally, based on user-provided examples like [3], the system generates concise editing instructions from the final caption. These instructions guide the editing of video content, including specific objects, styles, colors, and weather conditions, enabling adaptive adjustments to the captions. Through this process, we effectively produce high-quality video captions and flexible editing instructions.

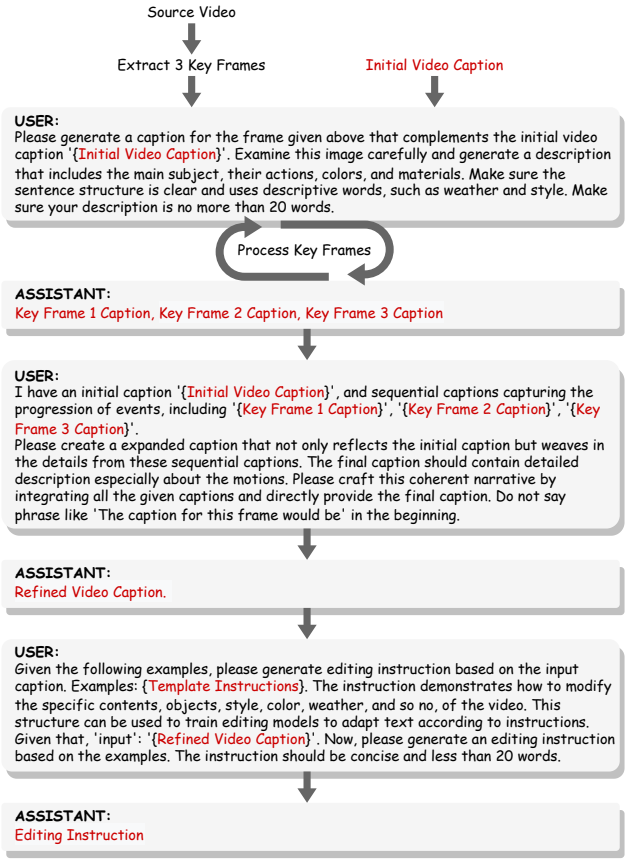

Figure 1. Pipeline and prompts of recaptioning and instruction generation.

In Tab. 1, we list the refined caption samples and the corresponding instructions with different editing types.

Table 1. Examples of refined captions and instructions.

Refined Caption	Instruction
The man in the blue shirt is eating a pizza on the boat.	Change the pizza to a sandwich.
The woman in the pink shirt is holding a green apple and smiling.	Replace the apple with an orange.
The red car is driving on a street with a yellow and green flag.	Make the flag blue and white.
A person holds a helmet, bright lighting highlighting its design.	Change to nighttime.
The man in the blue shirt and glasses is sitting in a room.	Add snow effect to the room.
The man in the gray jacket is driving a car.	Convert to watercolor portrait.

1.2. Prompts for Screening and Filtering

We illustrate the screening and filtering process mentioned in Sec. 3 of the main paper, and present the simplified prompts in Fig. 2 of the main paper. Below, we provide the complete prompts along with the input format for the GPT-4o API [1]. The prompts for screening are first presented, followed by the prompts employed for filtering.

Prompts of Screening:
System:
You are an advanced AI model specifically trained to assess the naturalness of edited images. Your task is to evaluate a set of edited images based on their adherence to the original image and the provided editing instructions. Here's how to perform the evaluation:
- Strict Adherence: Assess whether each edited image strictly follows the provided instructions. The modifications should directly reflect the requested changes without any deviations.
- Integration of Edits: Assess whether each edited image is seamlessly blended with the original image. The modifications should maintain a visual balance and consistency in color and tone.
- Absence of Artifacts: Evaluate whether the edits appear natural and free from any noticeable artifacts that would detract from the image.
- Subject Matter Consistency: Check for any distortions or elements that could have been introduced during the editing process. The edited images should be consistent in terms of lighting and shadows.
- Identify the Best Edit: Determine which edited image best reflects the requested changes and appears the most natural compared to the original.
User:
Please evaluate the following images based on their quality and natural appearance: The first image is the original image, and the next five images are the edited images. Editing Instructions: {instruction}. Based on your evaluation, identify which edited image best adheres to the original and editing instructions. Specify which image it is (0 through 5). Return the result as a Python dictionary string with the key 'best_image' indicating the number of the best image. DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. Only provide the Python dictionary string. For example, your response should look like this: {'best_image': 3}.
This is the first image: {'source_url'}.
Edited images are as follows: {'edited_url_0'}, {'edited_url_1'}, {'edited_url_2'}, {'edited_url_3'}, {'edited_url_4'}, {'edited_url_5'}.

Prompts of Filtering:
System:
You are an advanced AI tasked with evaluating the quality of video edits based on the adherence to specific editing instructions and the consistency of the edited frames. Your evaluation should focus on the following criteria:
- Strict Adherence: Assess whether each edited image strictly follows the provided instructions. The modifications should directly reflect the requested changes without any deviations.
- Integration of Edits: Assess whether each edited image is seamlessly blended with the original image. The modifications should maintain a visual balance and consistency in color and tone.
- Absence of Artifacts: Evaluate whether the edits appear natural and free from any noticeable artifacts that would detract from the image.
- Subject Matter Consistency: Check for any distortions or elements that could have been introduced during the editing process. The edited images should be consistent in terms of lighting and shadows.
- Composition Coherence: Examine the overall composition after the edits. The layout should maintain the visual balance across the frames.
- Content Consistency: Compare the edited frames with the original frames, ensuring that the contents are consistent across the frames.
Please conduct this evaluation by meticulously applying these criteria to determine the quality of the edits.
User:
Please evaluate the following video edit based on the provided instructions: The first three frames are from the original video, and the last three frames are from the edited video. Editing Instructions: {instruction} Based on your evaluation, answer the following questions: (1) Provide your evaluation solely as a quality score where the quality score is an integer value between 1 and 5, with 5 indicating the highest level of adherence to the instructions and overall quality. (2) Describe the aspects of the edit that were not executed well, including any artifacts or inconsistencies detected. Please generate the response in the form of a Python dictionary string with key 'score'. 'score' should be an integer indicating the quality score. DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. For example, your response should look like this: {'score': 3}.
Images from source video: {'source_url_0'}, {'source_url_1'}, {'source_url_2'}.
Images from edited video: {'edited_url_0'}, {'edited_url_1'}, {'edited_url_2'}.

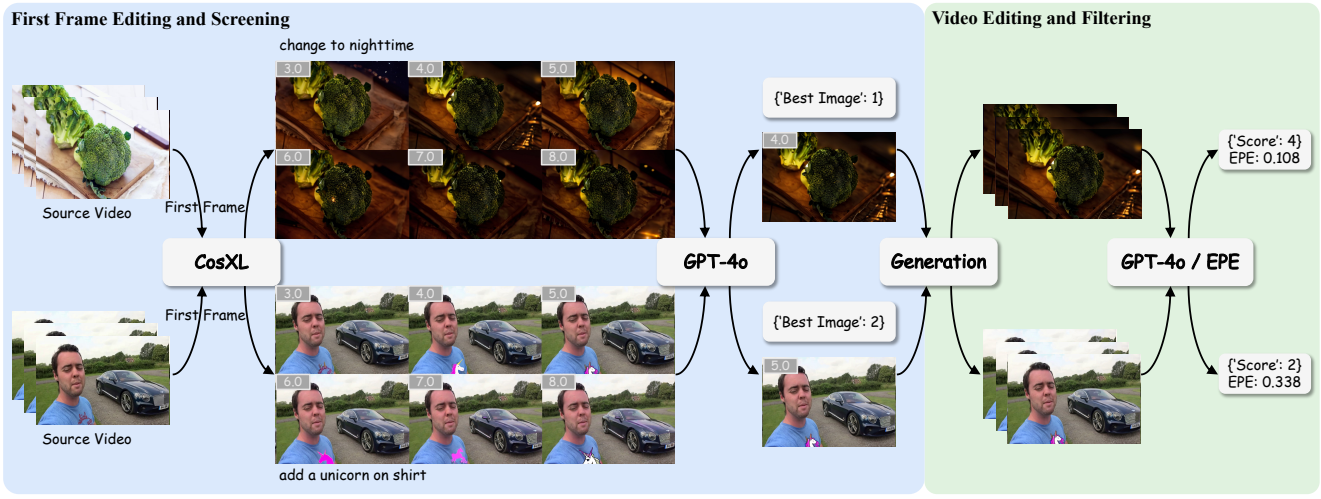

Figure 2. Examples of generating triplets from real-world videos.

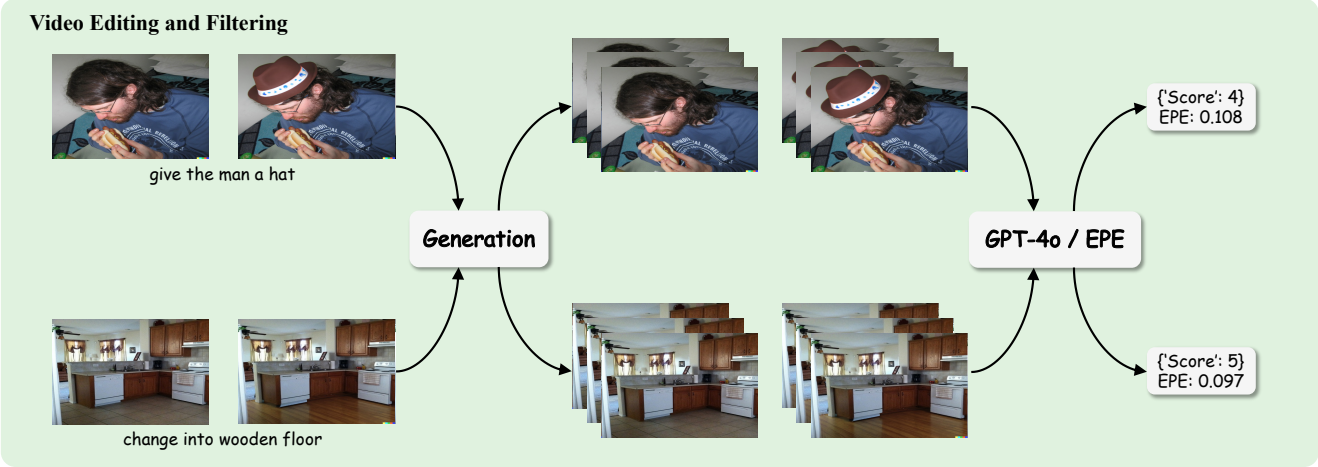

Figure 3. Examples of generating triplets from image editing pairs.

1.3. Examples of Triplets Construction Process

In this section, we provide examples of the construction process of the training triplets.

Triplet generation from real-world videos. In Fig. 2, we show two examples of the triplet generation from real-world videos by simplifying the intermediate process.

Triplet generation from image editing pairs. In Fig. 3, we show two examples of the triplet generation from image editing pairs by simplifying the intermediate process.

Generate static video triplets from real-world images. In Fig. 4, we show two examples of the triplet generation from real-world images by simplifying the intermediate process. Most of the construction pipeline is the same with triplet generation from real-world videos, while the generation step in "Video Editing and Filtering" is replaced by the addition of the camera motion. Specifically, we illustrate the detailed process of adding camera motion in the bottom example of Fig. 4, which is mentioned in Sec. 3.3 of the main paper. For "zoom in" and "zoom out", we set the minimum cropping size to 90% of the original image size and produce image sequences by gradually decreasing or increasing the cropping size. For "move right", "move left", "move down" and "move up", we set the cropping size to 90% of the original image size and produce image sequences by gradually adjusting the cropping location.

1.4. The Selection of CFGs

In Sec. 3.1 of the main paper, we choose a range CFGs to edit the video first frames. We randomly select 10K first frames and images from our initial dataset and utilize CosXL [2] to produce the edited outputs using various CFGs (from 1.0 to 10.0) for each image. Then we use GPT-4o [1] to screen the edited images and count the numbers

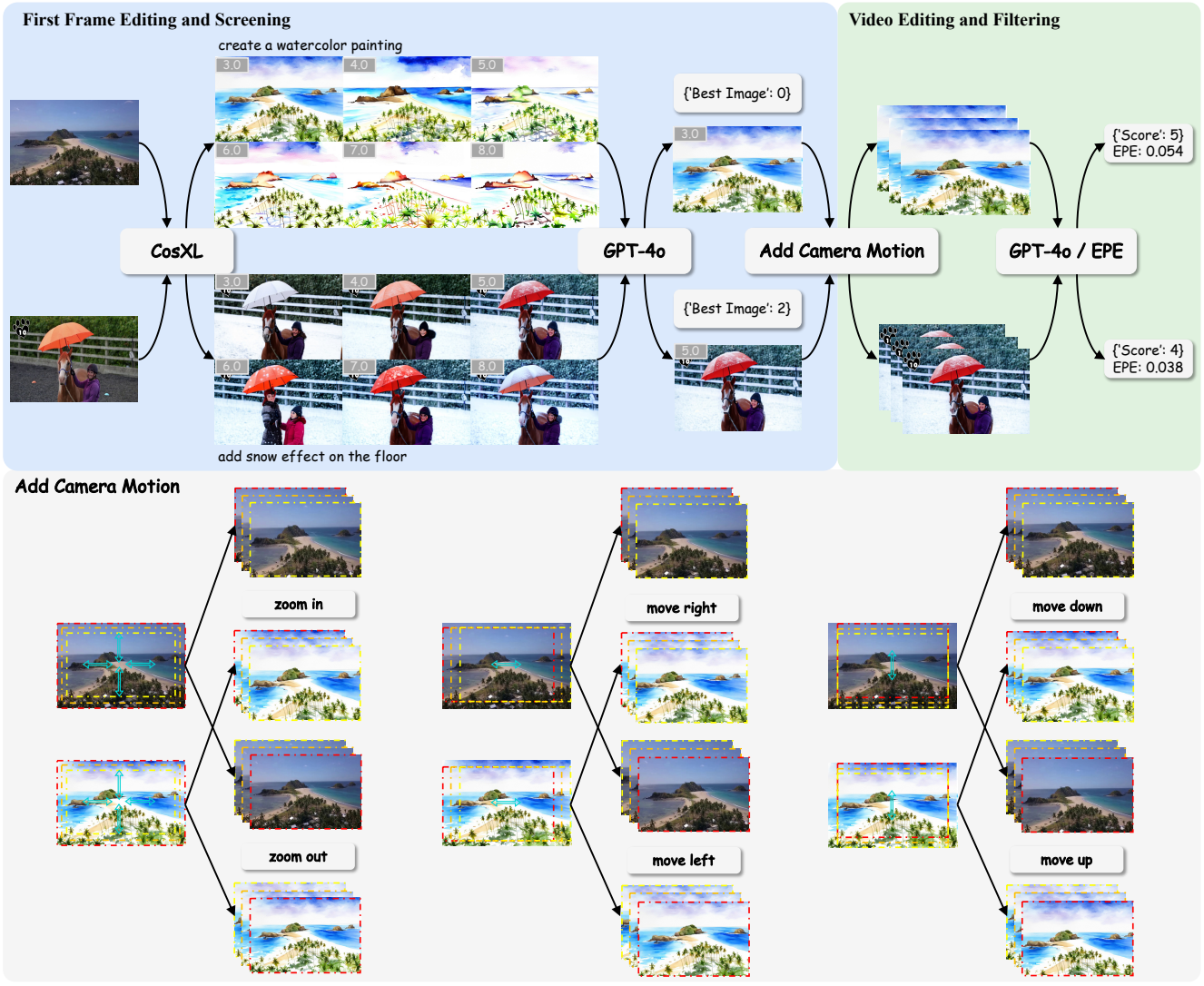

Figure 4. Examples of generating triplets from real-world images.

of best edited images produced by each CFG. As shown in Fig. 5, most of the best samples can be generated with CFGs from 3.0 to 8.0. Therefore, we set CFG within [3, 8] to generate 6 edited samples, which is also acceptable in terms of resource consumption.

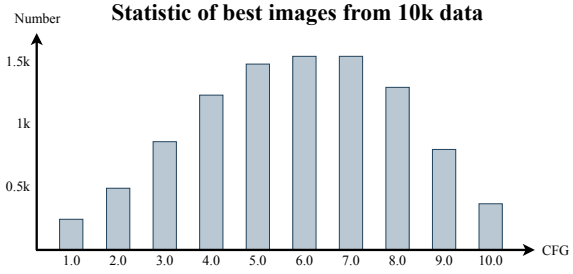

Figure 5. Statistic of the best edited images with different CFGs on 10K images.

2. Training and Testing

Implementation details. We train the InsViE model using similar settings to the default settings of CogVideoX [5]. The training is conducted on 8 nodes, each equipped with 8 Nvidia A100 GPUs, utilizing a batch size of 128 for a total of 40k steps. The Adam optimizer is employed with exponential moving average (EMA), setting the learning rate to 1e-3, betas to 0.9 and 0.95, weight decay to 1e-5, and EMA decay to 0.9999. The training data comprises a diverse set of video samples, with 720×480 pixels and 25 frames per video, ensuring consistency across inputs. At the last stage, both the weight of LPIPS loss and L_2 loss are set as 0.5.

Prompt for GPT score of testing. In terms of using GPT-4o to evaluate the edited videos, the selection of frames and prompts differs from the screening and filtering process out-

lined in data construction pipeline. Firstly, instead of sampling three key frames from video pairs as described in Sec. 3.1 of the main paper, we input all the frames to GPT-4o in testing stage, since the resource consumption associated with the scale of the test set is acceptable. Secondly, according to the "Evaluation Metrics" in Sec. 5.1 of the main paper, we provide the scores of GPT-4o across three aspects as a new metric. The original prompts used in the screening and filtering process are slightly adjusted. To be specific, the prompts for evaluating temporal consistency and textual alignment are modified to concentrate on each respective aspect, while the prompts for evaluating the video quality remains the same as the prompts of filtering. The revised prompts are shown below.

Temporal Consistency:
System:
You are an advanced AI tasked with evaluating the quality of video edits based on the adherence to specific editing instructions and the consistency of the edited frames. Your evaluation should focus on the following criteria:
- Composition Coherence: Examine the overall composition after the edits. The layout should maintain the visual balance across the frames.
- Content Consistency: Compare the edited frames with the original frames, ensuring that the contents are consistent across the frames.
Please conduct this evaluation by meticulously applying these criteria to determine the quality of the edits.
User:
Please evaluate the following video edit based on the provided instructions: The first half of the frames are from the original video, and the second half of the frames are from the edited video. Editing Instructions: {instruction} Based on your evaluation, answer the following question: Provide your evaluation solely as a score that is an integer value between 1 and 5, with 5 indicating the highest level of temporal consistency between videos and across the frames. Please generate the response in the form of a Python dictionary string with key 'score'. 'score' should be an integer indicating the temporal consistency score. DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. For example, your response should look like this: {'score': 3}.
Images from source video: {'source_url_0'}, ..., {'source_url_n'}.
Images from edited video: {'edited_url_0'}, ..., {'edited_url_n'}.

Textual Alignment:
System:
You are an advanced AI tasked with evaluating the quality of video edits based on the adherence to specific editing instructions and the consistency of the edited frames. Your evaluation should focus on the following criteria:
- Strict Adherence: Assess whether each edited image strictly follows the provided instructions. The modifications should directly reflect the requested changes without any deviations.
- Integration of Edits: Assess whether each edited image is seamlessly blended with the original image. The modifications should maintain a visual balance and consistency in color and tone.
User:
Please evaluate the following video edit based on the provided instructions: The first half of the frames are from the original video, and the second half of the frames are from the edited video. Editing Instructions: {instruction} Based on your evaluation, answer the following question: Provide your evaluation solely as a score that is an integer value between 1 and 5, with 5 indicating the highest level of textual alignment of the edited video frames. Please generate the response in the form of a Python dictionary string with key 'score'. 'score' should be an integer indicating the textual alignment score. DO NOT PROVIDE ANY OTHER OUTPUT TEXT OR EXPLANATION. For example, your response should look like this: {'score': 3}.
Images from source video: {'source_url_0'}, ..., {'source_url_n'}.
Images from edited video: {'edited_url_0'}, ..., {'edited_url_n'}.

3. More Visual Results

In this section, we provide more samples of InsViE-1M dataset and more qualitative comparisons between InsViE and previous methods. As shown in Figs. 6 and 7, we present more triplet samples of our InsViE-1M dataset, including removal, substitution, addition, stylization, *et al*. Additional comparisons with previous methods are shown in Figs. 8 to 12. From the visual comparisons, one can see that our InsViE model achieves better editing performance among various editing instructions, producing more visually more pleasing videos.

4. More Ablation Studies

Ablation on the LPIPS loss in Stage 3. As described in Sec. 4.2 in the main paper, we use L_2 loss in the first two training stages. LPIPS loss is added in the final stage to

Table 2. Ablation study on static-real ratio in the final training stage.

Training Settings	Temporal Consistency			Textual Alignment			Video Quality	
	CLIP ↑	OF EPE ↓	GPT Score ↑	CLIP ↑	Pick Score ↑	GPT Score ↑	DOVER ↑	GPT Score ↑
Static:Real=0:1	0.951	4.88	3.82	19.15	18.70	3.79	0.519	3.65
Static:Real=0.5:1	0.954	4.89	3.86	19.21	18.73	3.80	0.540	3.71
Static:Real=1:1	0.956	4.85	3.87	19.18	18.69	3.79	0.547	3.72
Static:Real=5:1	0.956	4.84	3.87	19.37	18.91	3.84	0.567	3.79

Table 3. Ablation study on the LPIPS loss in Stage 3.

Stage 1&2&3	TC GPT↑	TA GPT↑	DOVER↑	VQ GPT↑
w/ LPIPS	**3.88**	**3.84**	**0.567**	**3.79**
w/o LPIPS	3.86	3.83	0.543	3.73

enhance detail generation. As shown in Tab. 3, it contributes more to video quality metrics.

Ablation on static-real ratio. We further investigate the impact of different ratios of static to real videos in Set-S3. In Tab. 2, "Static:Real=0:1" exhibits similar results to "Stage 1&2", indicating the limitation of using real videos only. Increasing the ratio to "0.5:1" leads to better results than "Stage 1&2" on all the metrics. By setting "Static:Real=1:1", the model's performance stabilizes with better DOVER and GPT quality scores, demonstrating the benefits of static videos for visual quality. The most notable gain can be observed at "Static:Real=5:1", especially on the textual alignment and video quality.

References

[1] *https://cdn.openai.com/gpt-4o-system-card.pdf (2024)*. 2, 3

[2] Stability AI. Cosxl: A text-to-image model. Hugging Face Model Hub, 2024. 3

[3] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 1

[4] Wenyi Hong, Weihan Wang, Qingsong Lv, Jiazheng Xu, Wenmeng Yu, Junhui Ji, Yan Wang, Zihan Wang, Yuxiao Dong, Ming Ding, et al. Cogagent: A visual language model for gui agents. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14281–14290, 2024. 1

[5] 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. *arXiv preprint arXiv:2408.06072*, 2024. 4

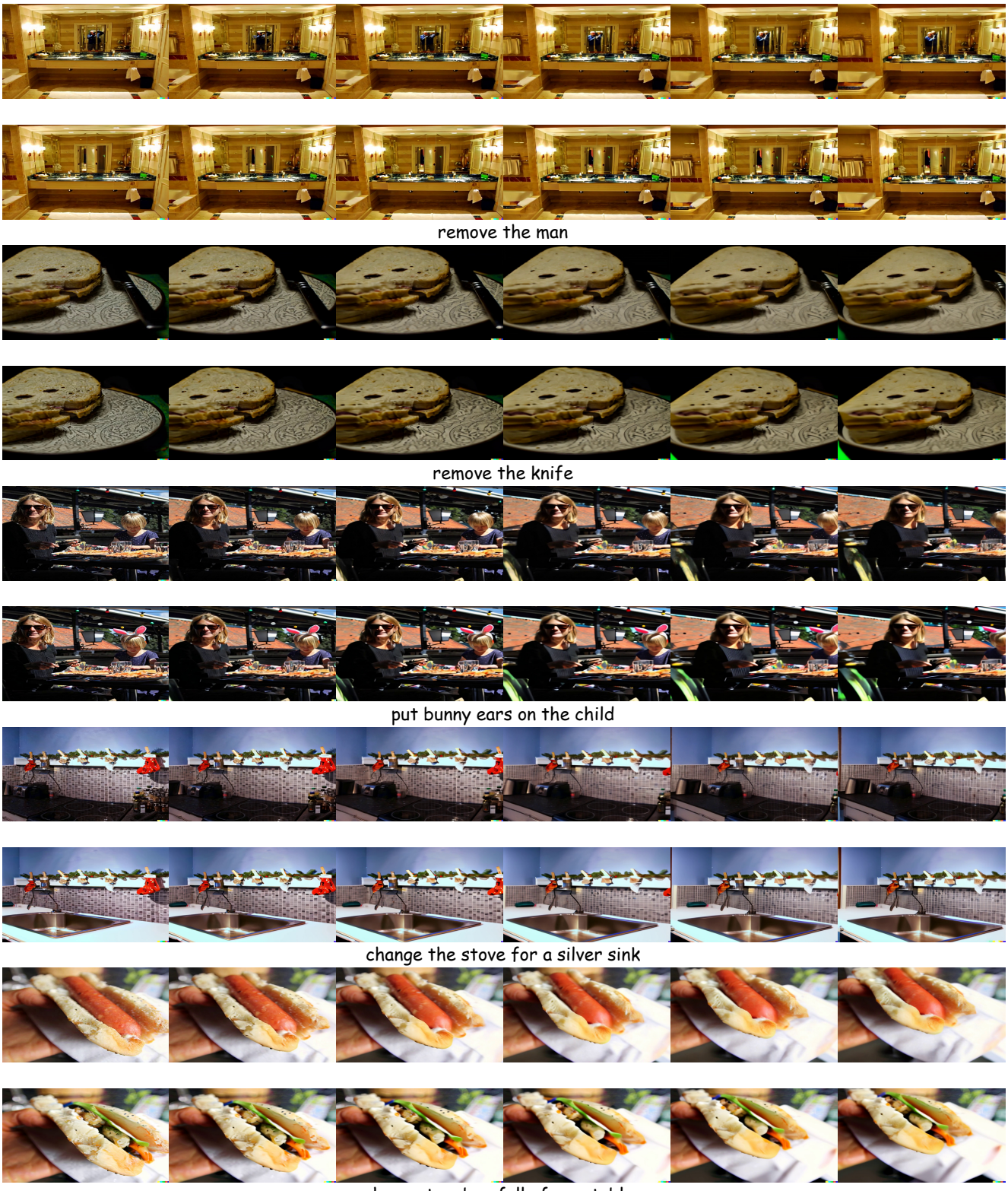

Figure 6. Sample triplets of our InsViE-1M dataset. For each sample, from top to bottom: original video, edited video, instruction.

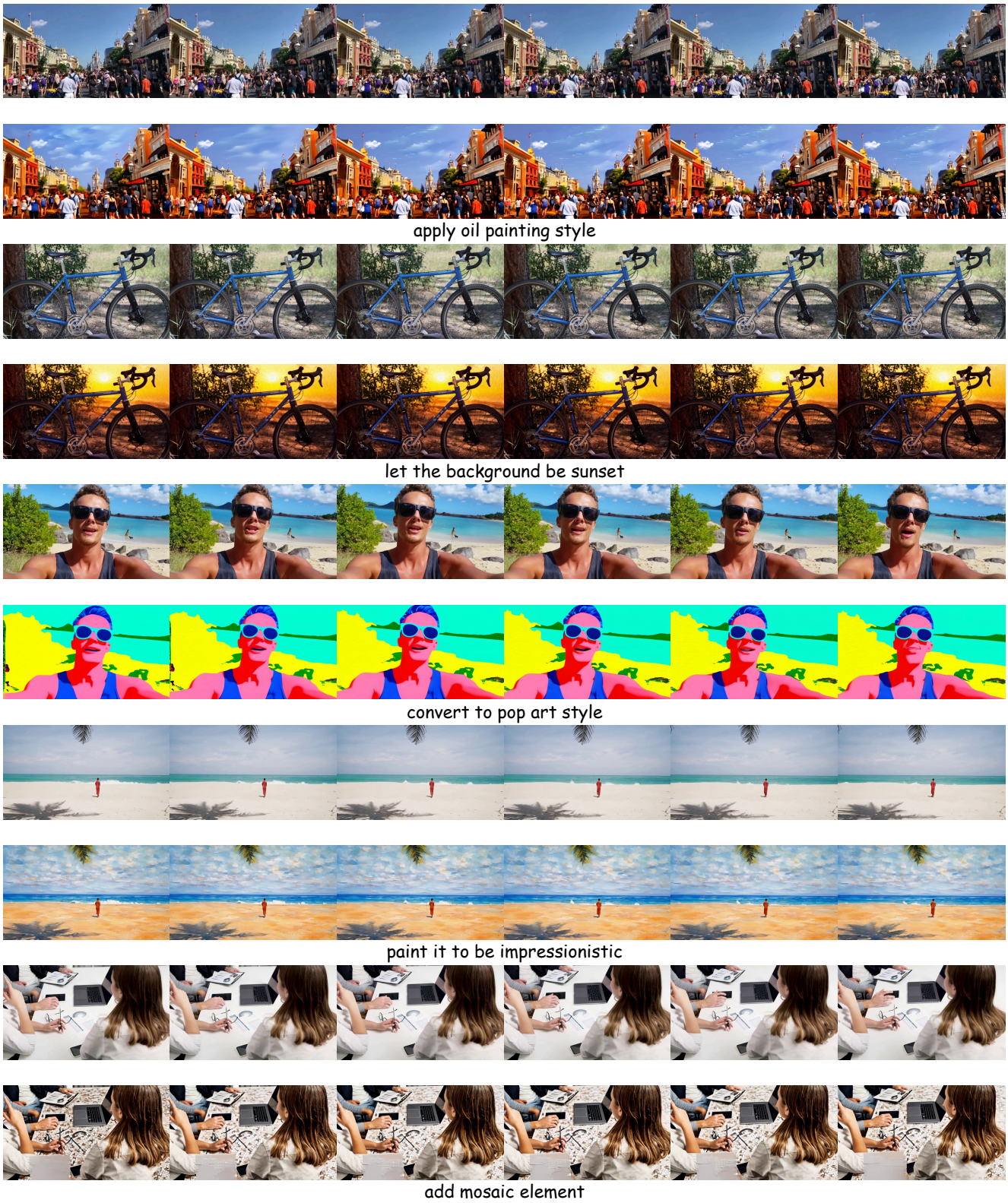

Figure 7. Sample triplets of our InsViE-1M dataset. For each sample, from top to bottom: original video, edited video, instruction.

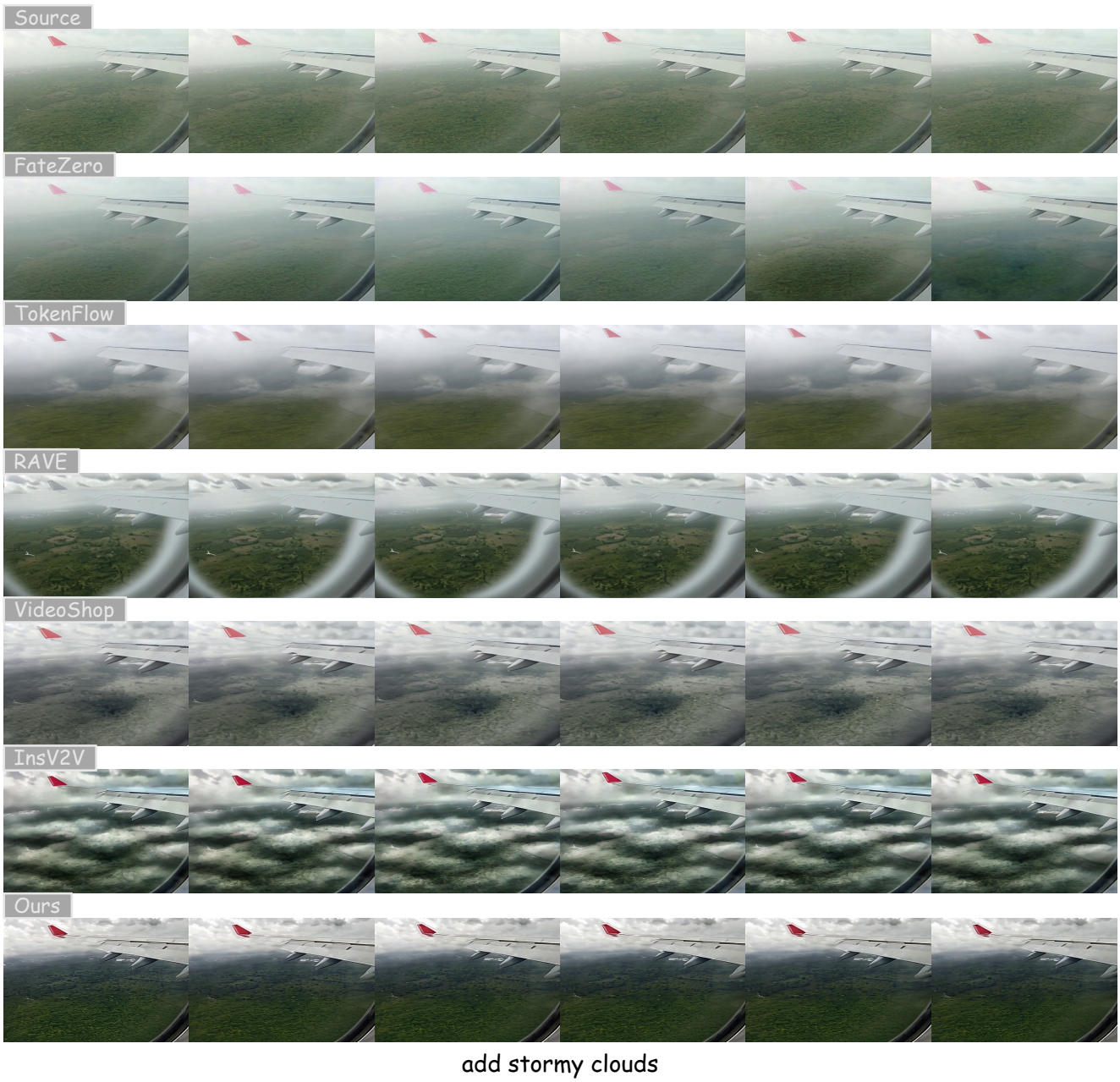

Figure 8. Visual comparison between our InsViE model and state-of-the-art methods.

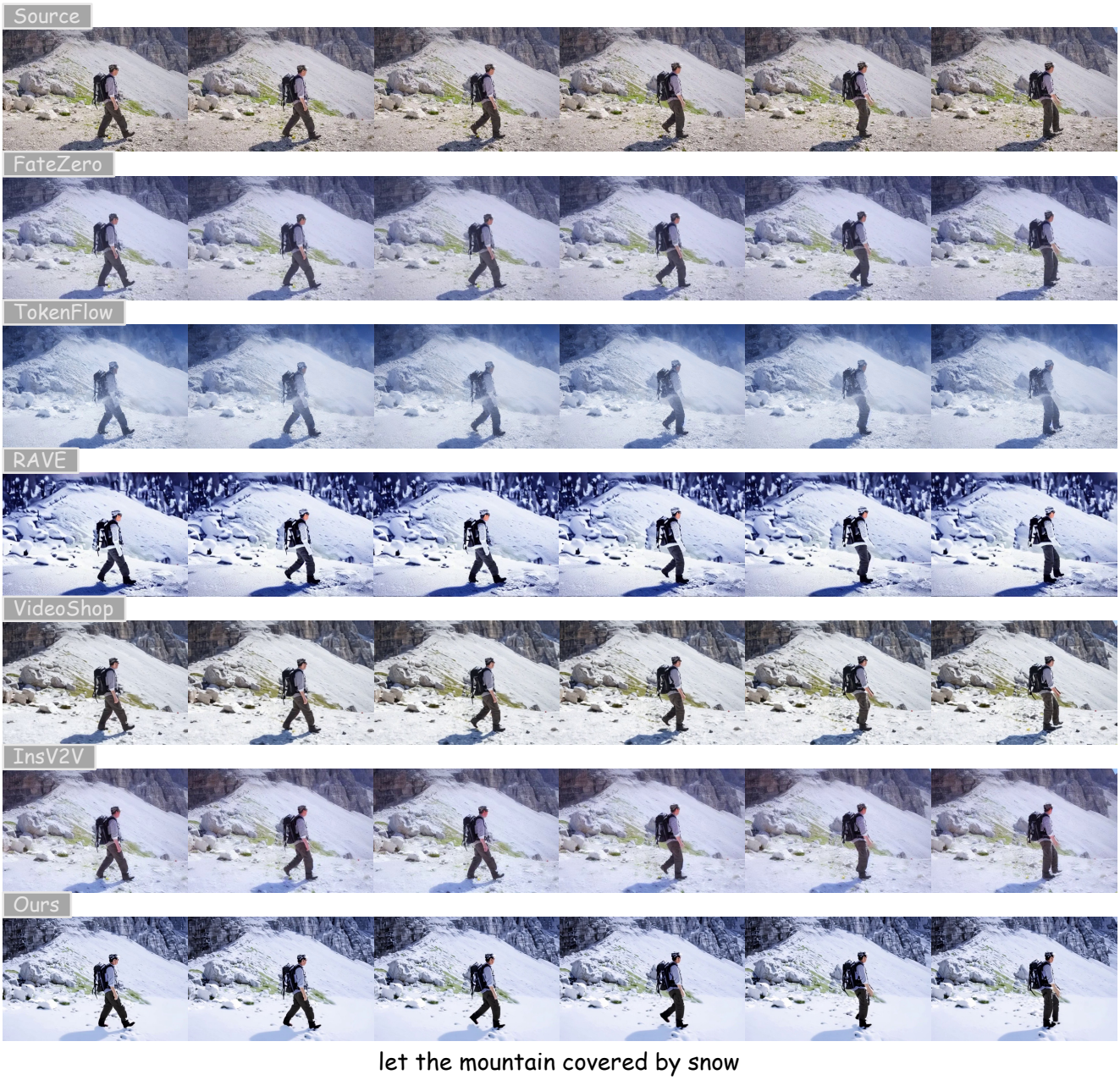

Figure 9. Visual comparison between our InsViE model and state-of-the-art methods.

Figure 10. Visual comparison between our InsViE model and state-of-the-art methods.

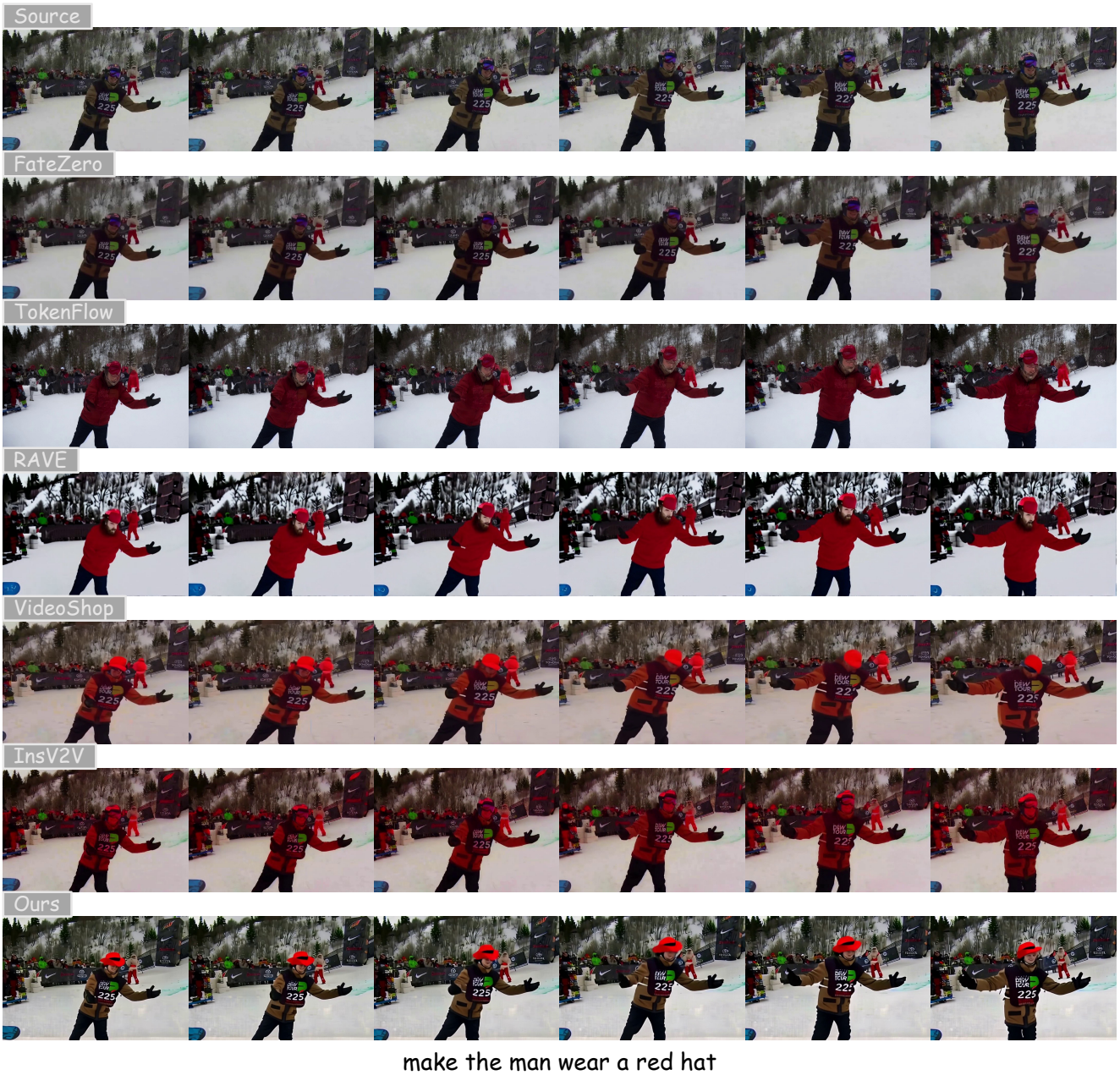

Figure 11. Visual comparison between our InsViE model and state-of-the-art methods.

Figure 12. Visual comparison between our InsViE model and state-of-the-art methods.