

# V-Stylist: Video Stylization via Collaboration and Reflection of MLLM Agents

## Supplementary Material

### A. TVSBench Details

Due to the lack of a suitable complex video stylization dataset that includes open user queries, multiple scene transitions, and challenging cases (e.g., large motion, small object), we have constructed our benchmark, namely TVSBench (Text-driven Video Stylization Benchmark), as depicted in Fig. 1.

Given the simplicity and static nature of video samples in current academic video editing, which struggle to meet the complexities of real-world video applications, we have collected 50 more challenging video samples targeting five key challenges in video stylization: large motion, occlusion and overlaying, small objects, similar foreground backgrounds, and multiple object interactions. The average duration of the full-length videos is 30 seconds at 30 fps, while the highlight versions average 5 seconds at 30 fps. The videos in the database are categorized into five distinct YouTube genres, with the subjects evenly distributed across Humans, Landscapes, Animals, and Vehicles, with an equal proportion for each category.

Each video is paired with a single text user query indicating the desired style preference. These queries follow four modal patterns established by [6], and the construction process involves the meticulous creation of four types of open user queries:

- **Prompt-based:** These queries are straightforward requests for a specific artistic style without additional context. They are typically used when the user has a clear vision of the desired outcome and wishes to communicate it directly to the system. e.g., “*Pixel art style.*” This prompt indicates the user wants the video to be stylized in the manner of pixel art, a style characterized by small, block-like images.
- **Inspiration-based:** These queries provide a broader context or a thematic inspiration for the style, often referencing a setting or scenario that embodies the desired aesthetic. They are used when the user wants to convey a mood or atmosphere that aligns with their vision. e.g., “*I would love to see a western realistic style video set in a baseball game.*” This prompt suggests the user is looking for a video that captures the realism of a western genre, specifically within the context of a baseball game.
- **Instruction-based:** These queries are more directive, specifying actions or subjects within the video content that need to be stylized in a particular way. They are used when the user has specific instructions for how elements within the video should be treated stylistically. e.g., “*Render this man who is practicing kung fu in a clayma-*

*tion style.*” This prompt instructs the system to stylize a specific action (a man practicing kung fu) in the style of claymation, a technique that uses models made from clay or other malleable materials.

- **Hypothesis-based:** These queries propose a potential style as a hypothesis, often with a degree of uncertainty or suggestion. They are used when the user is unsure of the best style or is open to suggestions from the system. e.g., “*Perhaps a Japanese anime style is the best choice to enhance this video’s aesthetics.*” This prompt hypothesizes that applying a Japanese anime style could improve the video’s visual appeal, leaving room for the system to confirm or propose alternatives.

To enrich the dataset, we utilized GPT4 [1] to mimic and generate an additional 40 similar texts, which were then refined by human experts to ensure quality, diversity, and consistency.

The TVSBench quantitative metrics span three dimensions: *Condition Alignment*, *Temporal Consistency*, and *Video Quality*. Condition Alignment evaluates CLIP-T, measuring CLIP [7] score between frame content and text prompts, and CLIP-W, assessing the match between style words and frames by CLIP [7]. Temporal Consistency measures structural coherence with the SSIM Score [10] and semantic preservation with the CLIP Score [7]. Video Quality assesses image-level aesthetics and technical appeal using Aesthetic Quality-I and Distortion Quality-I, and video-level aesthetics by the LAION aesthetic predictor [9] and MUSIQ image quality predictor [5], while DOVER [11] provides technical integrity with Aesthetic Quality-V and Distortion Quality-V.

### B. Implementation details of Style Reflection

In the paper, the Style Reflection Algorithm is meticulously designed to optimize the weights of ControlNet for precise style transformation in videos. The process commences with the initialization of weights for softedge, tile, depth, and lineart, all set to a random value between 0.1 and 0.3. This initial setup lays the groundwork for subsequent reflection processes, where a human preference example for each style category serves as in-context guidance for the MLLM to facilitate the reflection process.

The algorithm accepts a video shot  $\mathcal{X}_t$ , along with a style model  $\mathcal{M}_L$ , a content prompt  $\mathcal{P}_t$  derived from the Video Parser, and a set of ControlNets  $\mathcal{C}_{1:N}$  with their respective weights  $\mathcal{W}_{1:N}$ . The video shot is then stylized by applying the style model  $\mathcal{M}_L$ , which incorporates the content prompt  $\mathcal{P}_t$  and the weighted ControlNets to produce the

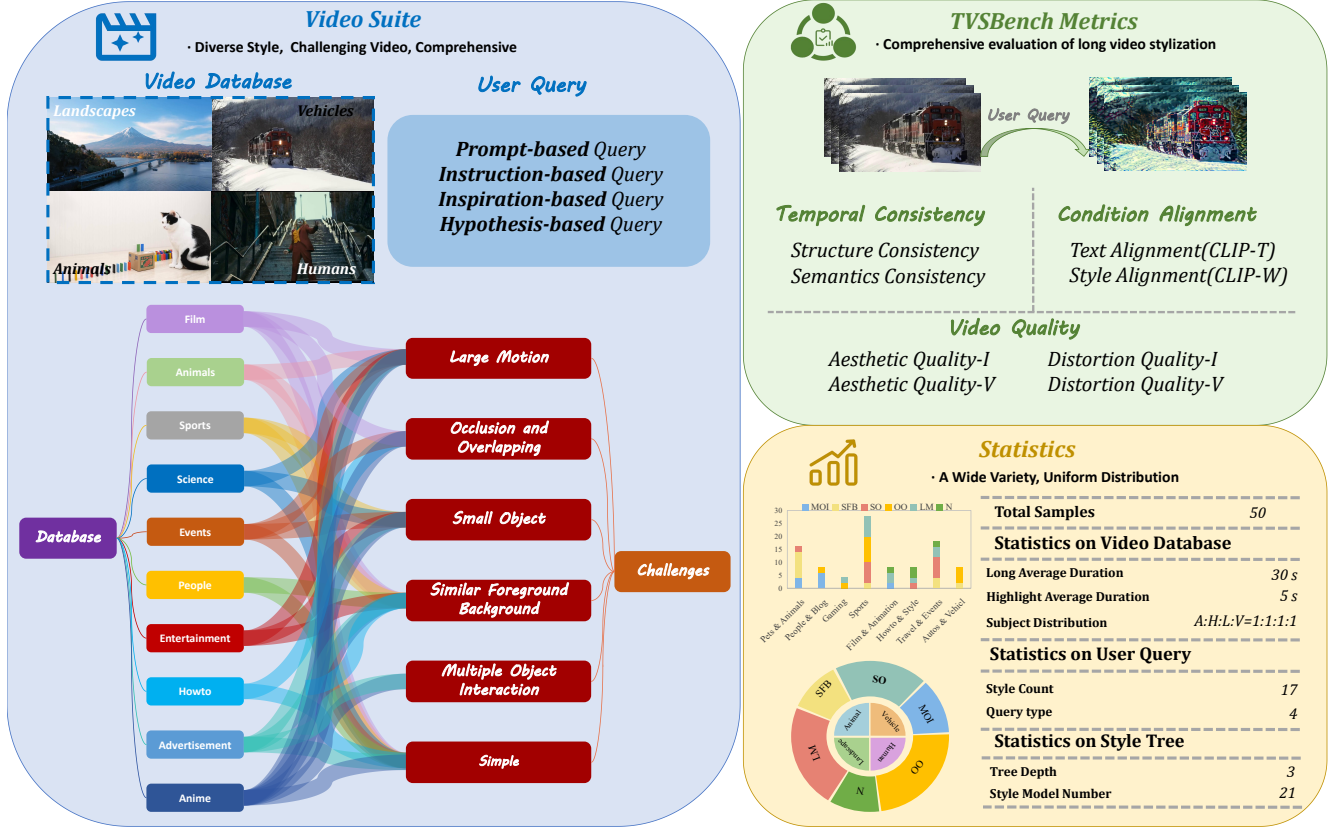


Figure 1. **Overview of the TVSBench.** The left portion of the figure showcases the extensive video database, encompassing a multitude of genres including landscapes, vehicles, animals, and human subjects. It underscores the complexities involved in stylizing videos with significant motion, instances of occlusion and overlapping, the presence of small objects, challenging foreground and background similarities, intricate multiple object interactions, and the pursuit of stylistic simplicity. Furthermore, it elaborates on the spectrum of open user queries, ranging from those driven by prompts-based, instructions-based, inspirations-based, to hypothesis-based. The upper right section delineates the evaluative metrics integral to TVSBench, which encompass temporal and structural consistency, semantic coherence, condition and text alignment, style congruence, as well as the dual aspects of aesthetic and distortion qualities within video content. The lower right section presents a comprehensive set of statistics. These include a diverse and evenly distributed array of video samples, alongside an in-depth analysis of style counts, the categorization of user queries, the architectural depth of the style tree, and the inventory of style models, collectively providing a robust framework for video stylization assessment.

stylized shot  $\mathcal{Y}_t$ . The stylized shot  $\mathcal{Y}_t$  is then evaluated by an MLLM, which assigns a style score  $\mathcal{R}^{(i)}$  based on the style match, aesthetics, and other criteria. This score serves as a measure of how well the stylized shot meets the desired style criteria. If the score is below the threshold of 60, indicating that the stylized shot is not satisfactory, the algorithm enters the Style Reflection phase.

In the Style Reflection phase, the algorithm adaptively adjusts the weight scores  $\mathcal{W}_{1:N}$  through a multi-round self-reflection process. The weights at the  $i$ -th round are denoted as  $\mathcal{W}_{1:N}^{(i)}$ , and initially, all weights are set to the same value. Based on these weights, a new stylized shot  $\mathcal{Y}_t^{(i)}$  is generated, and its style is evaluated by the MLLM, resulting in a new style score  $\mathcal{R}^{(i)}$ .

If the style score  $\mathcal{R}^{(i)}$  is higher than the threshold, the

stylized shot  $\mathcal{Y}_t^{(i)}$  is considered satisfactory and is used as the final output. If not, the MLLM is used again to generate new weights  $\mathcal{W}_{1:N}^{(i+1)}$ , taking into account the previous stylized shot  $\mathcal{Y}_t^{(i)}$  and its style score  $\mathcal{R}^{(i)}$ . This initiates another round of style rendering and reflection.

The algorithm sets a maximum number of rounds  $T$  to prevent an infinite loop. If the stylized shot remains unsatisfactory after the maximum number of rounds, the algorithm terminates and selects the round with the highest style score as the final output. This approach ensures that the V-Stylist can dynamically adjust to different video content and style requirements, enhancing the flexibility and effectiveness of the video stylization process.

Through this iterative process, the Style Reflection Algorithm allows for a more nuanced and adaptive control over

---

**Algorithm 1** Style Reflection Algorithm

---

```
1: Given video shot  $\mathcal{X}_t$ , style model  $\mathcal{M}_L$ , content prompt  $\mathcal{P}_t$ , and ControlNets  $\mathcal{C}_{1:N}$  with weights  $\mathcal{W}_{1:N}$ 
2:  $\mathcal{Y}_t \leftarrow \mathcal{M}_L(\mathcal{X}_t, \mathcal{P}_t \mid \mathcal{C}_{1:N}, \mathcal{W}_{1:N})$   $\triangleright$  Stylize the shot
3: Initialize weights  $\mathcal{W}_{1:N}^{(1)}$  and set  $i = 1$ 
4:  $\mathcal{Y}_t^{(i)} \leftarrow \mathcal{Y}_t$ 
5:  $\mathcal{R}^{(i)} \leftarrow \text{MLLM}(\mathcal{Y}_t^{(i)})$   $\triangleright$  Evaluate style
6: while  $\mathcal{R}^{(i)} < 60$  and  $i \leq T$  do
7:    $\mathcal{W}_{1:N}^{(i+1)} \leftarrow \text{MLLM}(\mathcal{Y}_t^{(i)}, \mathcal{R}^{(i)})$   $\triangleright$  Refine weights
8:    $\mathcal{Y}_t^{(i+1)} \leftarrow \mathcal{M}_L(\mathcal{X}_t, \mathcal{P}_t \mid \mathcal{C}_{1:N}, \mathcal{W}_{1:N}^{(i+1)})$   $\triangleright$  Re-stylize the video shot
9:    $\mathcal{R}^{(i+1)} \leftarrow \text{MLLM}(\mathcal{Y}_t^{(i+1)})$ 
10:  if  $\mathcal{R}^{(i+1)} > \mathcal{R}^{(i)}$  then
11:     $\mathcal{R}^{(i)} \leftarrow \mathcal{R}^{(i+1)}$ 
12:     $\mathcal{W}_{1:N}^{(i)} \leftarrow \mathcal{W}_{1:N}^{(i+1)}$ 
13:  end if
14:   $i \leftarrow i + 1$ 
15: end while
16: Output the final stylized shot  $\mathcal{Y}_t^{(i)}$  with the highest score
```

---

the visual details of the stylized video shots, moving beyond a one-size-fits-all approach to style transformation. This results in a more personalized and higher-quality stylization that aligns with the diverse and complex nature of video content and user preferences.

### C. Style Tree Details

When constructing the style tree, we gathered a number of models of 17 various styles and their corresponding model cards from CivitAI [3]. These models were categorized into two major classes: Artistic and Realistic. Each distinct model was then mapped to the appropriate style category as leaf nodes within the branches of the tree, as illustrated in the Fig. 2. This systematic approach allowed us to create a comprehensive and organized structure that represents the diversity of styles available, making it easier to navigate and select the desired style for specific applications or projects. Naturally, this tree is designed to be dynamically scalable, accommodating additional styles as they are developed.

Under the Artistic, we find styles that lean towards creative expression and abstract representation, such as oil painting, expressionism, and various forms of anime, including flat anime, western anime, and japanese anime. This category also includes unique styles like ukiyo-e, pixel art, and abstract art, each with its own set of models and characteristics. For instance, the “pixel art style” is an example within this category, with the model “pixel.f2.safetensors” that is tagged with “artistic” and “pixel style” and is triggered by the keyword “pixel”.

The Realistic category, on the other hand, encompasses styles that aim to replicate or enhance the appearance of

real-world visuals. This includes styles like asian realistic, western realistic, and photolistic, etc. For the “asian realistic style,” we have the model “majicmixRealistic.v6.safetensors” which falls under the realistic category, specifically tailored to capture the essence of Asian scenes. This model is a checkpoint merge type, indicating that it combines the capabilities of multiple models to produce highly realistic outputs. It is tagged with “realistic” and “asian scenes,” and it operates on version “v6” of the base model “SD 1.5.”

Each distinct model within the Artistic and Realistic categories is mapped as a leaf node within the corresponding branch of the Style Tree. This hierarchical structure not only aids in navigation but also provides a clear overview of the relationships between different styles. As illustrated in Fig. 2, the tree is designed to be dynamically scalable, allowing for the incorporation of new styles and models as they emerge, ensuring that the Style Tree remains a comprehensive resource for style-based applications and projects.

### D. More Visualization of V-Stylist on Long Video Stylization

Fig. 4 demonstrates the qualitative results of our V-Stylist on longer video stylization. Our systematic collaborative approach yields videos with high condition alignment, temporal consistency, and video quality.

### E. More Qualitative Comparisons

Fig. 5, Fig. 6, and Fig. 7 respectively display the stylized results of different SOTA methods under various styles. The second to fourth rows are the condition images for Lineart, Depth, and Softedge, while the condition images for Tile are the original images. It can be observed that Control-A-Video [2] exhibits a noticeable color degradation phenomenon in long videos, while other methods, due to their use of a single ControlNet [14] and rigid weight settings, cannot balance the structure and color information as well as the alignment of styles. Our V-Stylist, by dynamically combining different types of structure controls, consistently demonstrates the highest condition alignment, temporal consistency, and video quality. Moreover, for some styles that are difficult to describe through text or are scarce in the training data, relying solely on text-driven rendering of the base model is challenging to achieve effective results. Different methods may employ various models. For uniformity, we use the original version of Stable Diffusion v1.5 [8] for Rerender-A-Video [12], FRESCO [13], ControlVideo [15], FLATTEN [4], and Control-A-Video [2]. For structure control, we utilize the default settings for each method: Rerender-A-Video employs ControlNet-HED, FRESCO uses ControlNet-Depth and DDIM Inversion, FLATTEN uses DDIM Inversion, ControlVideo uses

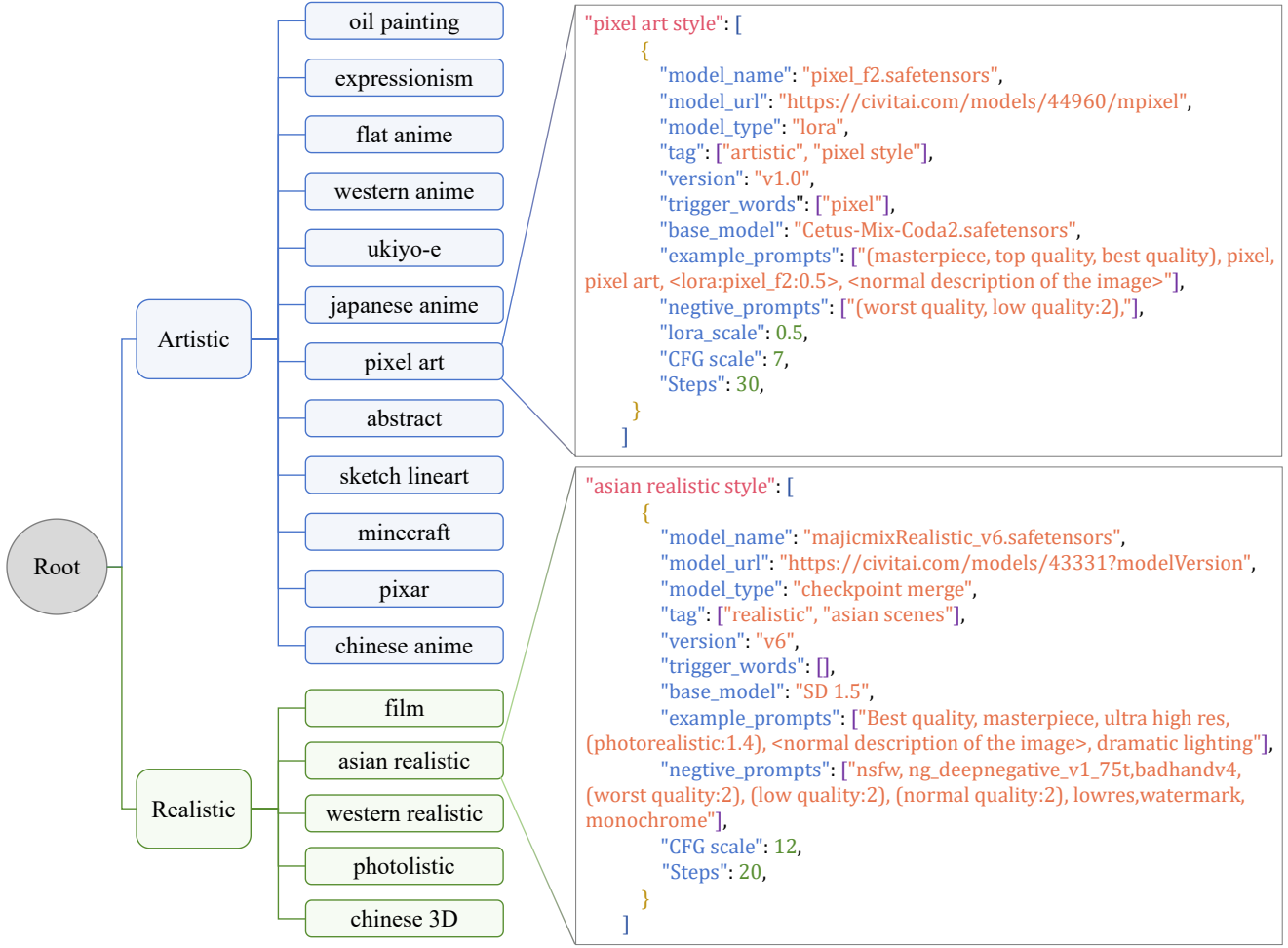


Figure 2. **Style Tree.** The Style Tree categorizes a variety of artistic and realistic styles within a hierarchical framework. It extends into branches that represent the subcategories falling under the broader categories of Artistic and Realistic styles. The entire tree is organized in JSON format, where each leaf node corresponds to a model card for a specific stylization model. These model cards include details such as model names, URLs, types, tags, and other parameters that are instrumental in directing the stylization process.

ControlNet-Depth, and Control-A-Video uses ControlNet-Depth.

Although the CLIP-W score of V-Stylist is slightly lower compared to Rerender-A-Video and ControlVideo, its actual performance is by no means inferior. This is evident when we examine Fig. 3, where both ControlVideo and Rerender-A-Video incorrectly render the yacht’s wake as green square land, a characteristic of the Minecraft style but a misrepresentation of the original video’s content. However, V-Stylist not only applies the Minecraft style effectively but also preserves the original content by accurately rendering the wake as waves, aligning with the natural depiction of a yacht’s movement on water. V-Stylist also excels in temporal consistency, ensuring that the style transformation is smooth and coherent across frames, which is vital for the viewer’s experience in video content. While

Rerender-A-Video and ControlVideo may achieve a higher Style Alignment score according to CLIP, their aesthetic quality and fidelity to the original video’s content are compromised. V-Stylist’s holistic approach to style transfer considers not only the aesthetic style but also the importance of content alignment and temporal consistency, making it a superior choice for high-quality video transformations that respect the original video’s essence. Incorporating other examples in Fig. 5, Fig. 6, and Fig. 7, we hypothesize that the lower CLIP metric scores for V-Stylist imply that CLIP might not be fully equipped to capture the subtleties of styles, such as Expressionism and other abstract art styles, that are not adequately represented in the training data for CLIP. This insight will also inspire us to further optimize our benchmark metrics to better evaluate the performance on diverse and less conventional artistic styles.



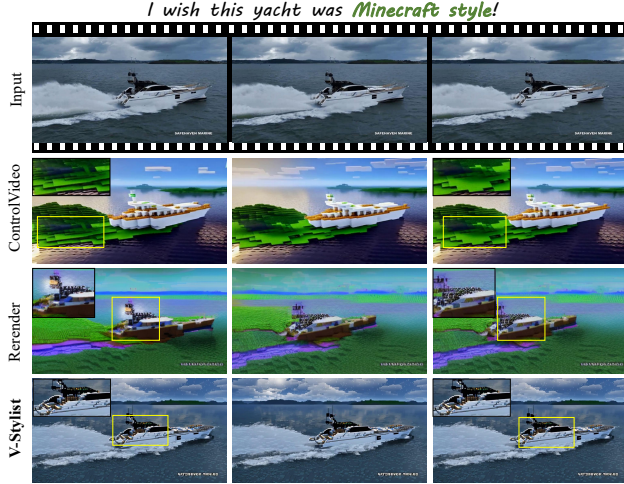


Figure 3. **Qualitative Comparison with ControlVideo and Rerender-A-Video.** It can be observed that both ControlVideo and Rerender-A-Video have rendered the yacht’s wake as green square land, whereas V-Stylet accurately rendered the wake as waves. Although the CLIP calculates a higher Style Alignment for them, their aesthetic quality of style, as well as the precision of transformation and temporal consistency, are inferior to ours.

## F. More Ablation Study

Ablation Study of Video Parser compares three different methods of generating stabel diffusion prompts: using only style words, combining raw captions with style words, and using prompts with style words. The results, as shown in Tab. 1, demonstrate the incremental improvements in both text alignment and video quality metrics. The table indicates that the integration of captions with style words and the use of prompts with style words both lead to significant enhancements in video quality metrics compared to using only style words. This suggests that the context provided by captions and prompts is crucial for improving the model’s performance in generating high-quality videos that align well with the given conditions.

Ablation Study of Style Parser compares the effectiveness of direct LLM decisions and a progressive LLM search strategy. The results, presented in Tab. 2, highlight the differences in performance between these two approaches. The findings from this study suggest that the progressive search strategy (Tree Search) outperforms both the base model and the direct search method, indicating that a more nuanced approach to style model selection can lead to better alignment and higher video quality. This underscores the importance of a structured search process in achieving optimal style model.

Models	Condition Alignment		Video Quality			
	CLIP-T $\uparrow$	CLIP-W $\uparrow$	Aesthetic-I $\uparrow$	Aesthetic-V $\uparrow$	Distortion-I $\uparrow$	Distortion-V $\uparrow$
Only Style Word	0.2556	0.1166	0.5569	0.6294	0.5756	0.6204
Caption + Style Word	0.2592	0.1300	0.5628	0.6383	0.5800	0.6284
Prompts + Style Word	<b>0.2627</b>	<b>0.1519</b>	<b>0.5687</b>	<b>0.6473</b>	<b>0.5844</b>	<b>0.6364</b>

Table 1. **Ablation Study of Video Parser.** Experiments conducted based on the Stable Diffusion v1.5-base model demonstrate that Shot Captioner and Shot Translator have achieved improvements in text alignment and video quality through step-by-step optimization of the input text prompts.

Models	Condition Alignment		Video Quality			
	CLIP-T $\uparrow$	CLIP-W $\uparrow$	Aesthetic-I $\uparrow$	Aesthetic-V $\uparrow$	Distortion-I $\uparrow$	Distortion-V $\uparrow$
Base Model	0.2627	0.1166	0.5687	0.6473	0.5844	0.6364
Direct Search	0.2655	0.1300	0.5780	0.6600	0.5900	0.6500
Tree Search	<b>0.2662</b>	<b>0.1519</b>	<b>0.5950</b>	<b>0.6887</b>	<b>0.5895</b>	<b>0.7028</b>

Table 2. **Ablation Study of Style Parser.** SD1.5 indicates the base model of Stable Diffusion v1.5 without style model search. Direct Search indicates exporting all style model names and tags, conducting a single round of LLM Q&A to select the style model; if the search fails (LLM’s answer is not in the model list, then the base model is used directly). Tree Search indicates the results obtained using the Style Tree progressive search.

## G. Full Prompts for V-Stylet

We present our complete LLM and MLLM prompts for V-Stylet, including Video Parser in Fig. 8, Style Parser in Fig. 10 and Fig. 10, Style Artist in Fig. 11 and Fig. 12, We fully take advantage of In-context learning, Chain-of-thoughts and Tree-of-thoughts prompting techniques to enhance LLM’s reasoning and decision-making capabilities.

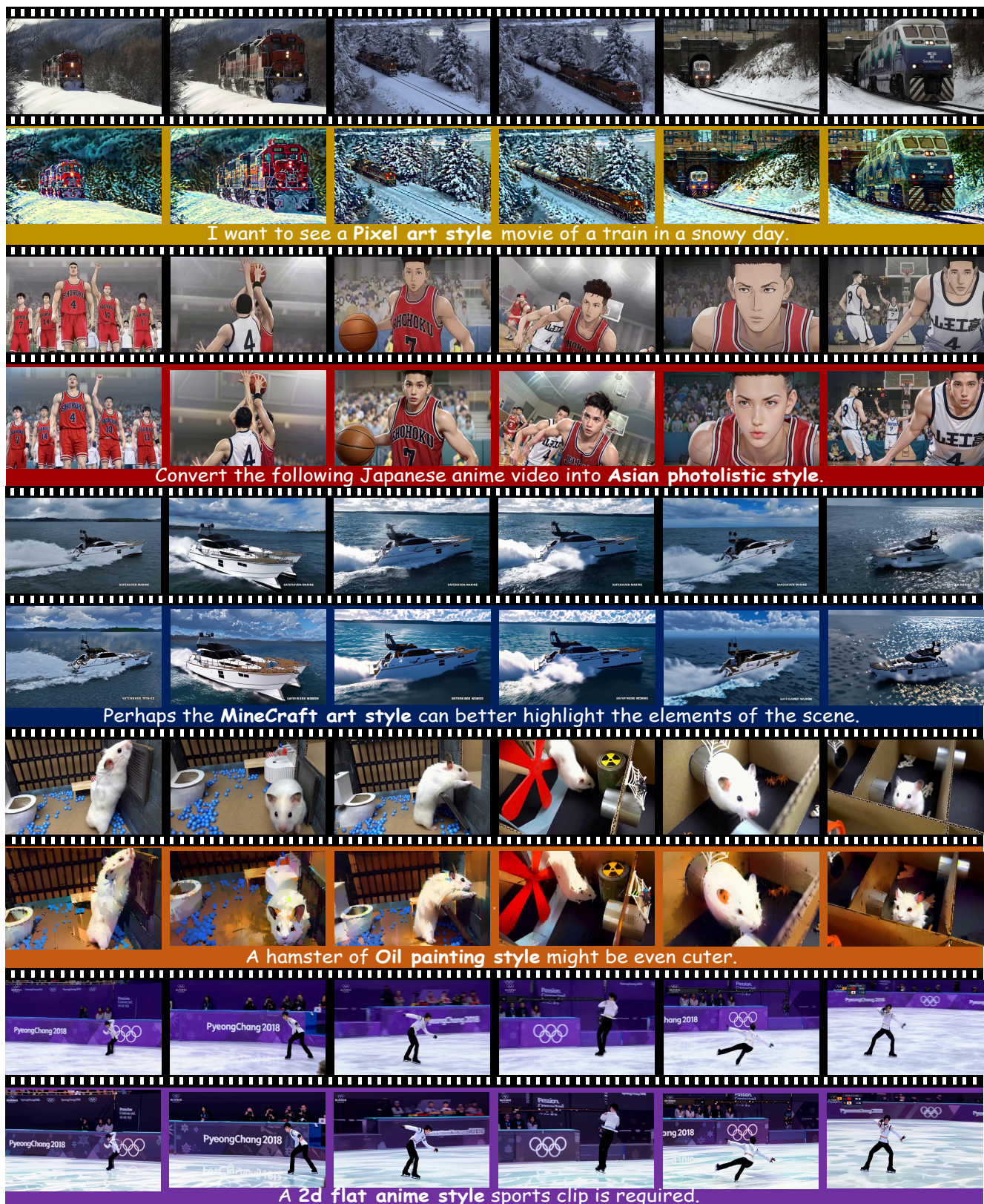


Figure 4. Visualization of different video stylizations of V-Stylist.



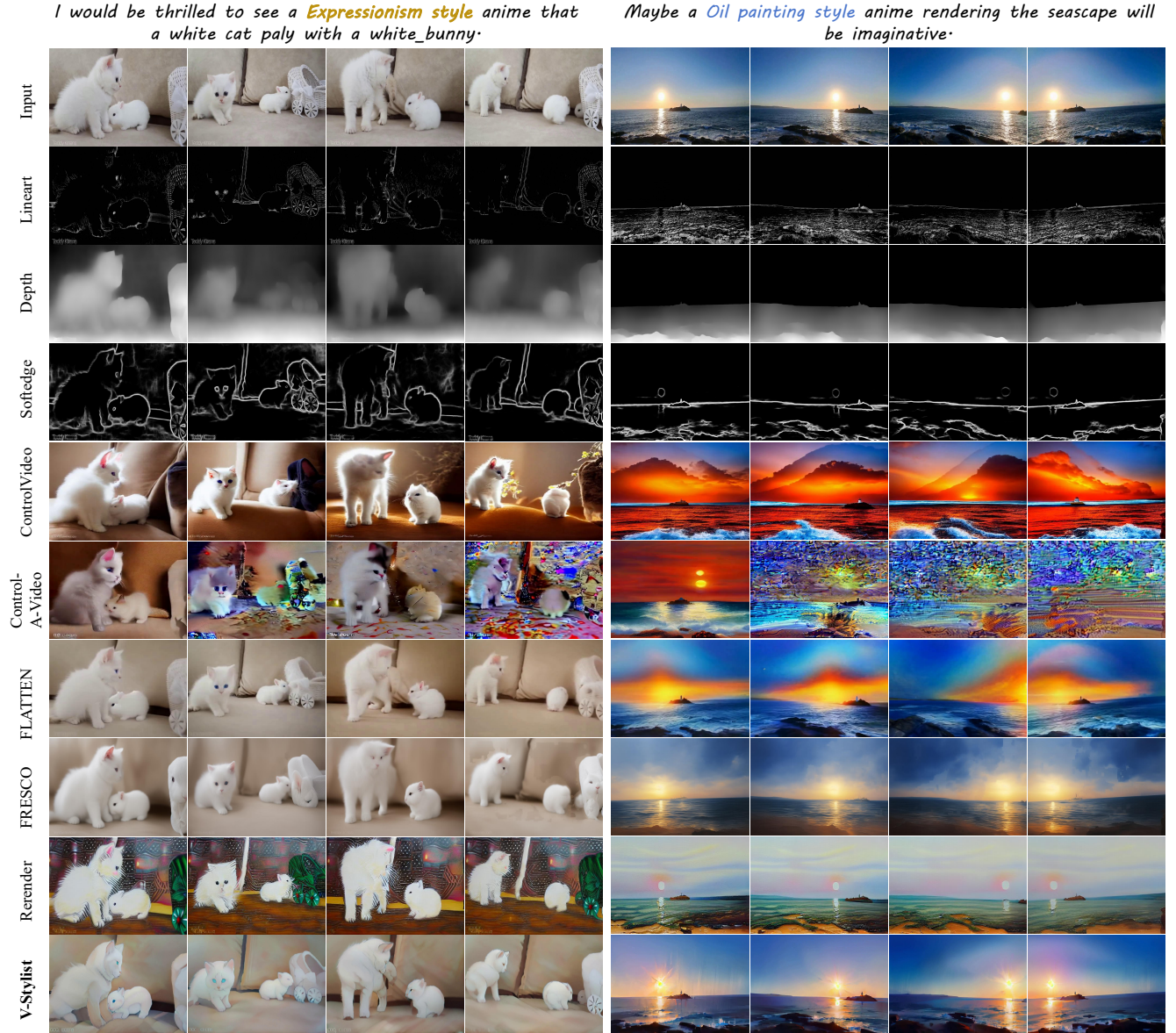


Figure 5. **Qualitative Comparison (1) With Existing SOTA Methods.** Our V-Stylist consistently demonstrates the highest condition alignment, temporal consistency, and video quality. We compare our V-Stylist with SOTA open-sourced models, including Rerender-A-Video, FRESCO, ControlVideo, FLATTEN, Control-A-Video. The first row is the original video frame, the second to third rows are different structure control images, the fourth to ninth rows are results from different SOTA methods, and the last row is the result from V-Stylist. Color-marked texts at the top indicate specific style preferences.

*Pixel art style, colorful, delicate snowboarding sport.*

*Flat anime style.*

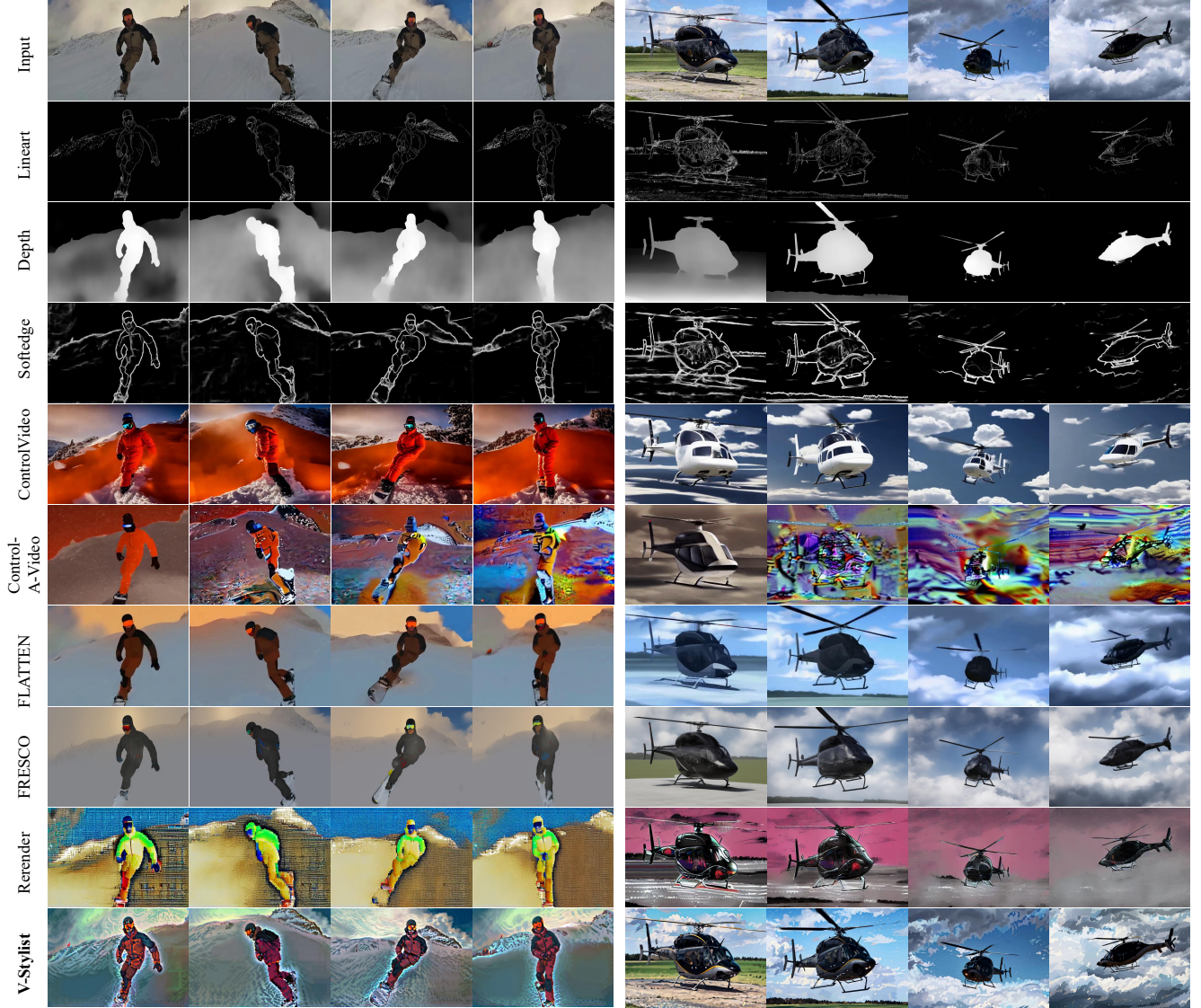


Figure 6. **Qualitative Comparison (2) With Existing SOTA Methods.** Our V-Stylist consistently demonstrates the highest condition alignment, temporal consistency, and video quality. We compare our V-Stylist with SOTA open-sourced models, including Rerender-A-Video, FRESCO, ControlVideo, FLATTEN, Control-A-Video. The first row is the original video frame, the second to third rows are different structure control images, the fourth to ninth rows are results from different SOTA methods, and the last row is the result from V-Stylist. Color-marked texts at the top indicate specific style preferences.



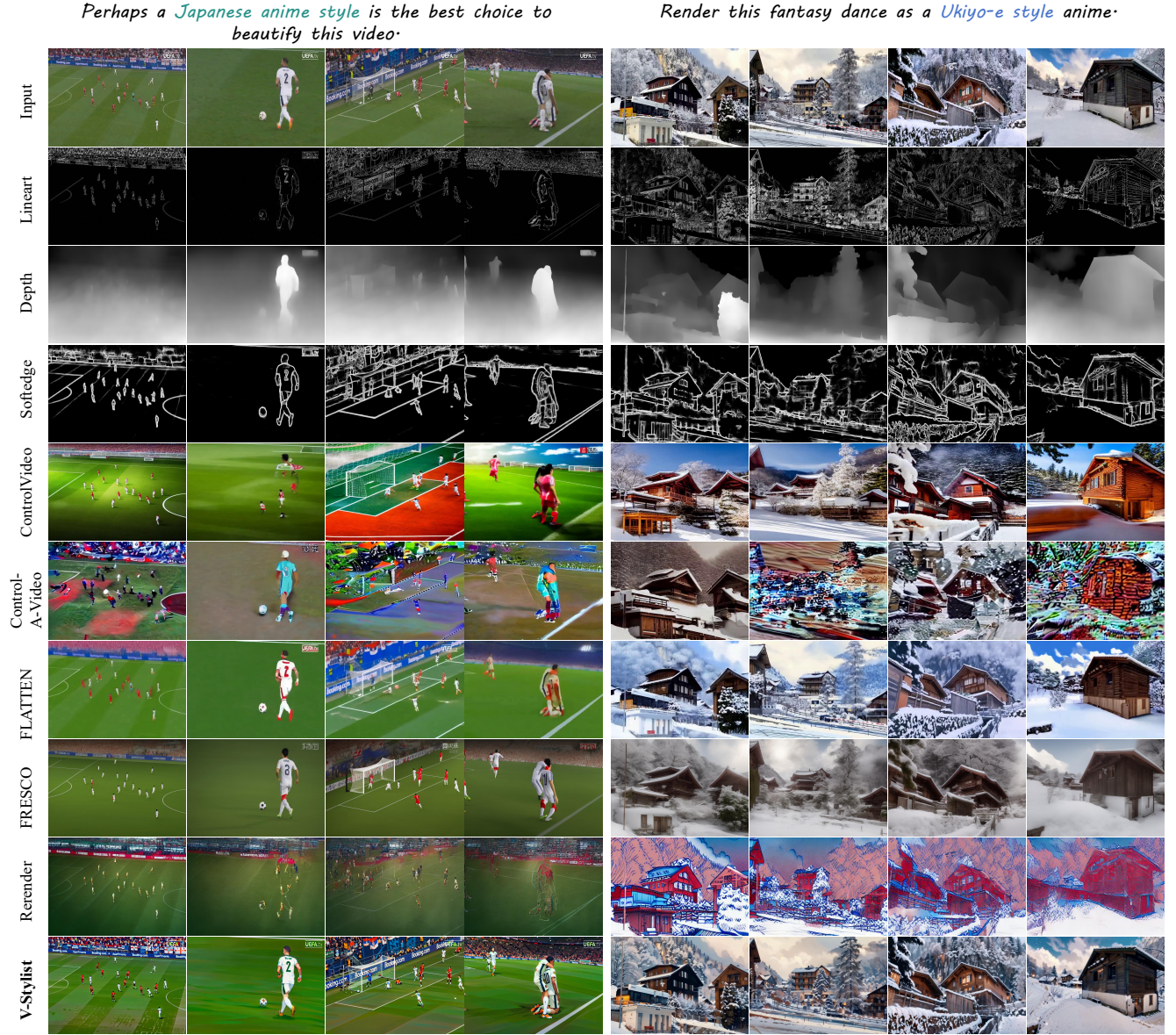


Figure 7. **Qualitative Comparison (3) With Existing SOTA Methods.** Our V-Stylist consistently demonstrates the highest condition alignment, temporal consistency, and video quality. We compare our V-Stylist with SOTA open-sourced models, including Rerender-A-Video, FRESCO, ControlVideo, FLATTEN, Control-A-Video. The first row is the original video frame, the second to third rows are different structure control images, the fourth to ninth rows are results from different SOTA methods, and the last row is the result from V-Stylist. Color-marked texts at the top indicate specific style preferences.

### Prompt for Shot Captioner:

You are a professional shot captioner skilled at observing and describing frame details.

Your **task** is to offer a detailed frame description, capturing Characters, Actions, Objects, Colors, Background, Position, and other details. Include specifics like clothing color, background setting, character positioning, hair and skin tones, clothing texture, and lighting. Aim to detail every visible aspect of the frame.

Output format: *{"Description": Frame\_Contents}*. Ensure accuracy and visibility; It is important to avoid describing any details that do not appear in the frame.

*\*\*Frame Pixel Values: {}*

Output:

### Prompt for Shot Translator:

You are a professional Shot Prompt Translator for stable diffusion.

Your **task** is to help me design a brief visual prompt for stable diffusion based on 3 frame descriptions. I will provide three scene frame descriptions. **First**, summarize them into a detailed video description, excluding unmentioned elements. Format the positive prompt with high-weight elements before commas and low-weight after, as: *"subject (character & object + details) + scene (environment) + style (quality + artistic elements)"*. **Then**, create a negative prompt excluding unwanted elements like low quality or redundant features.

Output format: *{"Positive\_Prompt": Prompt\_Content, "Negative\_Prompt": Prompt\_Content}*.

**Some examples are given below.**

Example 1: {}

Example 2: {}

Example3: {}

*\*\*Descriptions: {}*

Output:

Figure 8. Prompts of Shot Captioner and Shot Translator in our Video Parser.

### Prompt for Style Identifier:

You are a Style Identifier for deciphering user style preferences from vague user query.

Your **task** is to assess the genre of the input user query and then identify the primary Style and Subcategories of the user query based on the user query and its corresponding form. If you can't identify a specific Style, give an summarization of the Style Preference.

- Style categories: [artistic, realistic]

- Subcategories: []

The output should match the format: *{"Style Preference": Preference\_Content}*

**\*\*User Query:** {}

Output:

### Prompt for Style Tree Builder-1:

You are an information analyst who can create a Knowledge Tree according to the input categories.

Your **task** is to place the each Style category as subcategory under the SubStyle categories based on whether it can be well matched with a specific subject category to form a reasonable scene.

Below is a knowledge tree output template: *{"Artistic": [SubStyle1, SubStyle2, ...], "Realistic": [SubStyle1, SubStyle2,...]}*

**\*\*Style Input:** {}

**\*\*SubStyle Input:** {}

Output:

Figure 9. Prompts of Style Identifier and Style Tree Builder in our Style Parser.

### Prompt for Style Tree Builder-2:

You are an information analyst who can add some input models to an input knowledge tree according to the similarity of the model tags and the categories of the knowledge tree.

You **task** is to place each input model into the appropriate subcategory on the tree, one by one. You **MUST** keep the original content of the knowledge tree. Be sure to differentiate strictly by tag. artistic and realistic model must be strictly separated !

Please output the final knowledge tree as: *{**"Artistic"**: [{SubStyle1: [Model\_Name1, Model\_Name2, ...]}, {SubStyle2: [Model\_Name1, Model\_Name2, ...]}, ...], **"Realistic"**: [{SubStyle1: [Model\_Name1, Model\_Name2, ...]}, {SubStyle2: [Model\_Name1, Model\_Name2, ...]}, ...]}*

**\*\*Knowledge Tree Input:** {}

**\*\*Models Input:** {}

**\*\*Model Tags Input:** {}

Output:

### Prompt for Style Searcher:

You are a Style Searcher for selecting the best model based on user style preference.

Your **task** is to choose the best matched model based on user style preference. **First**, Act as five different experts that are appropriate to select one element from the following Input list, which has best match tags for the following style preference. All experts will write down the selection result, then share it with the group. **Then**, you as chairman analyze all 5 analyses and output the consensus selected element or your best guess matched element.

The final selection output **MUST** be the same as: *{**"Selected"**: [the only one selected element]}*. [the only one selected element] **MUST** be only element in the Input list, and without other words!

**\*\*Style Preferenece:** {}

**\*\*Input List:** {}

Output:

Figure 10. Prompts of Style Tree Builder and Style Searcher in our Style Parser.



### **System Prompt for Style Artist:**

ControlNet is a neural network that guides style transformation by controlling image generation structures. Here's a concise explanation of its components and their scales (0-0.5):

- Lineart: Dictates fine details and structural lines, ideal for crisp comic or manga styles. Adjust the scale to control line dominance without impeding style transformation.
- Softedge: Suited for realistic styles requiring softer line delineation, allowing for gradual color transitions for a more organic look.
- Tile: Preserves original image colors, essential for styles needing color fidelity. Adjust the scale to balance color preservation with style transformation.
- Depth: Modifies object contouring to add or reduce depth, useful for styles emphasizing dimensionality.

### **To balance style and structure:**

- For structural transformations (e.g., Oil painting style), avoid high lineart scales to prevent overly strict adherence to original shapes.
- For texture and color transformations (e.g., Realistic style), avoid high tile scales to allow for style fidelity.
- For line art or anime styles, set higher lineart and tile scales to respect original lines and colors.

Understanding these controls enables the agent to adjust settings for desired style transformations while maintaining structural coherence.

Figure 11. System Prompts of Style Scorer and Control Refiner in our Style Artist.

### Prompt for Style Scorer in Style Artist:

You are a Style Scorer to evaluate and score the results of stylization.

Your **task** is to score the styled image generated by controlnet and give an analysis of the currently used control configs.

**First**, given an original image and a stylized image generated using controlnet based on the original image, and the corresponding control's config, from 0 to 100, what score do you think the stylized image should have for its generation? Do not dominate the scoring with a single attribute such as correctness of recognition, but rather give a comprehensive score on the degree of proximity to the target style, the aesthetics of the generated image, and the preservation of key features of the original image. **Second**, analyze the effect of control configs on the generation of stylized images (in terms of proximity to the target style, aesthetics of the generated image, and retention of key features of the original image), and analyze whether each control type and scale setting is reasonable, and whether it has a bad effect on the currently generated stylized image, and whether it should be larger or smaller. Should it be larger or smaller?

Your final answer must be in JSON format as `{"Final Score": score_value, "Control Analysis": asanalysis_string}`, where score\_value is the score from 0-100 and asanalysis\_string is the text of your analysis of the control configs. Follow the above two steps and give some explanation. Do not include any code.

**\*\*Target Style: {}**\n **\*\*Source Frame: {}**\n **\*\*Styled Frame: {}**\n **\*\*Control Configs: {}**\n

Output:

### Prompt for Control Refiner in Style Artist:

You are a Control Refiner to refine control weights for better stylization results.

Your **task** is to determine the optimal scale for each control type contained in the Control Configs based on Control Analysis and other information. Observe the Target Style, Source Image, and Styled Image yourself and analyze each control type according to the recommendations of Control Analysis. Observe the Target Style and Source Image and Styled Image yourself, and analyze whether the scale of each control type in Control Configs needs to be adjusted according to the recommendations of Control Analysis, and if not, keep its original value, and if it needs to be adjusted, give the value that you think is optimal. Note that the scale value of all controls must be between 0 and 0.5.

Your final answer must be a list of tuples as `[(control type, scale value), ...]`, where control type is the same as Control Configs. Follow the above steps and give some explanation.

**\*\*Target Style: {}**\n **\*\*Source Frame: {}**\n **\*\*Styled Frame: {}**\n **\*\*Control Configs: {}**\n **\*\*Control Analysis: {}** \n

Output:

Figure 12. Prompts of Style Scorer and Control Refiner in our Style Artist.

## References

- [1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1
- [2] Weifeng Chen, Yatai Ji, Jie Wu, Hefeng Wu, Pan Xie, Jiashi Li, Xin Xia, Xuefeng Xiao, and Liang Lin. Control-a-video: Controllable text-to-video generation with diffusion models. *arXiv preprint arXiv:2305.13840*, 2023. 3
- [3] Civitai. Civitai: The home of open-source generative ai. <https://civitai.com>, 2024. [Accessed 17-10-2024]. 3
- [4] Yuren Cong, Mengmeng Xu, Christian Simon, Shoufa Chen, Jiawei Ren, Yanping Xie, Juan-Manuel Perez-Rua, Bodo Rosenhahn, Tao Xiang, and Sen He. Flatten: optical flow-guided attention for consistent text-to-video editing. *arXiv preprint arXiv:2310.05922*, 2023. 3
- [5] Junjie Ke, Qifei Wang, Yilin Wang, Peyman Milanfar, and Feng Yang. Musiq: Multi-scale image quality transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 5148–5157, 2021. 1
- [6] Jie Qin, Jie Wu, Weifeng Chen, Yuxi Ren, Huixia Li, Hefeng Wu, Xuefeng Xiao, Rui Wang, and Shilei Wen. Diffusiongpt: Llm-driven text-to-image generation system. *arXiv preprint arXiv:2401.10061*, 2024. 1
- [7] 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. 1
- [8] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022. 3
- [9] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35:25278–25294, 2022. 1
- [10] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 1
- [11] Haoning Wu, Erli Zhang, Liang Liao, Chaofeng Chen, Jingwen Hou, Annan Wang, Wenxiu Sun, Qiong Yan, and Weisi Lin. Exploring video quality assessment on user generated contents from aesthetic and technical perspectives. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20144–20154, 2023. 1
- [12] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Rerender a video: Zero-shot text-guided video-to-video translation. In *SIGGRAPH Asia 2023 Conference Papers*, pages 1–11, 2023. 3
- [13] Shuai Yang, Yifan Zhou, Ziwei Liu, and Chen Change Loy. Fresco: Spatial-temporal correspondence for zero-shot video translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8703–8712, 2024. 3
- [14] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 3
- [15] Min Zhao, Rongzhen Wang, Fan Bao, Chongxuan Li, and Jun Zhu. Controlvideo: Adding conditional control for one shot text-to-video editing. *arXiv preprint arXiv:2305.17098*, 2(3), 2023. 3