SparseCtrl: Adding Sparse Controls to Text-to-Video Diffusion Models

Yuwei Guo[1] Ceyuan Yang[2†] Anyi Rao[3] Maneesh Agrawala[3] Dahua Lin[1,2] Bo Dai[2]

[1]The Chinese University of Hong Kong [2]Shanghai Artificial Intelligence Laboratory [3]Stanford University

{gy023,dhlin}@ie.cuhk.edu.hk {yangceyuan,daibo}@pjlab.org.cn
{anyirao,maneesh}@cs.stanford.edu

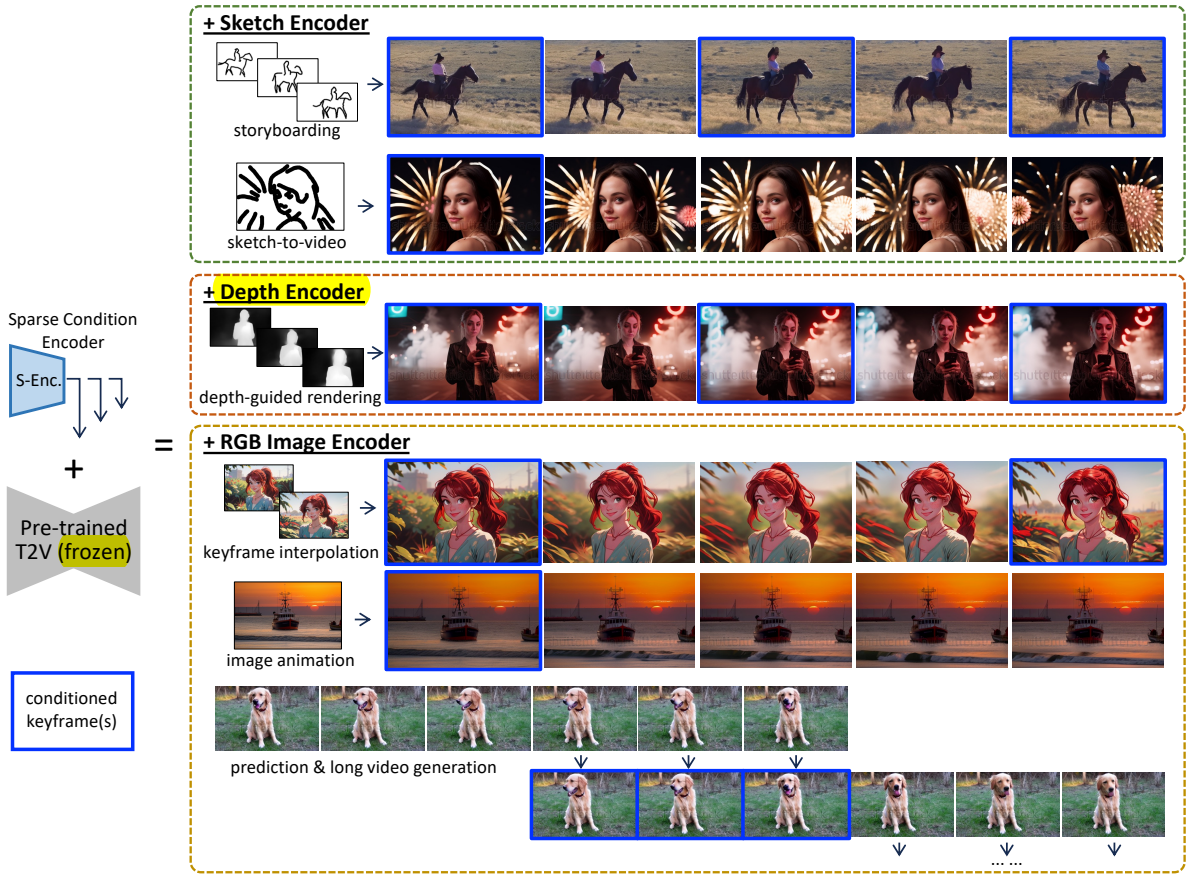

Figure 1. We present `SparseCtrl`, an add-on encoder network upon pre-trained text-to-video (T2V) diffusion models to accept additional temporally sparse conditions for specific keyframe(s), *e.g.*, sketch/depth/RGB image. Through integration with various modality encoders, `SparseCtrl` enables the pre-trained T2V for various applications including storyboarding, sketch-to-video, image animation, long video generation, etc. When combined with AnimateDiff [18] and enhanced personalized image backbones [5, 42], `SparseCtrl` also achieves controllable, high-quality generation results, as shown in the 2/3/4-th rows.

Abstract

The development of text-to-video (T2V), i.e., *generating videos with a given text prompt, has been significantly advanced in recent years. However, relying solely on text prompts often results in ambiguous frame composition due to spatial uncertainty. The research community thus leverages the* `dense` *structure signals,* e.g., *per-frame depth/edge sequences to enhance controllability, whose collection accordingly increases the burden of inference. In this work, we present* `SparseCtrl` *to enable flexible structure control with temporally* `sparse` *signals, requiring only one or few inputs, as shown in Fig. 1. It incorporates an additional condition encoder to process these sparse signals while leaving the pre-trained T2V model* `untouched`*. The proposed approach is compati-*

†Corresponding Author.

ble with various modalities, including sketches, depth, and RGB images, providing more practical control for video generation and promoting applications such as storyboarding, depth rendering, keyframe animation, and interpolation. Extensive experiments demonstrate the generalization of `SparseCtrl` *on both original and personalized T2V generators. Codes and models will be publicly available at https://guoyww.github.io/projects/SparseCtrl.*

1. Introduction

With the advance of text-to-image (T2I) generation [3, 9, 17, 27, 31, 33, 41, 54, 57] and large-scale text-video paired datasets [2], there has been a surge of progress in the field of text-to-video (T2V) generative models [4, 21, 43]. These developments enable users to generate compelling videos through textual descriptions of the desired content. Nonetheless, textual prompts, being inherently abstract expressions, struggle to accurately define complex structural attributes such as spatial layouts, poses, and shapes. This lack of precise control impedes its practical application in more demanding and professional contexts, such as anime creation and filmmaking. Consequently, users often find themselves engrossed in numerous rounds of random trial-and-error to achieve their desired outputs. This process can be time-consuming, especially since there is no straightforward method to guide the synthetic results toward the expected direction during the iterative trying process.

To unlock the potential of T2V generation, efforts have been made to incorporate more precise control through structural information. For instance, Gen-1 [11] pioneers using monocular depth maps as structural guidance. VideoComposer [51] and DragNUWA [59] investigate the domain of compositional video generation, employing diverse modalities such as depth, sketch, and initial image as control signals. Furthermore, prior studies [18, 26, 65] utilize the image ControlNet [62] to introduce various controlling modalities to video generation. By harnessing additional structural sequences, these approaches provide enhanced control capabilities. However, for precise output control, existing works necessitate temporally *dense* structural map sequences, which means that users need to furnish condition maps for each frame in the generated video, thereby increasing the practical costs. Additionally, most approaches towards controllable T2V typically redesign the model architecture to accommodate the extra condition input, which demands costly model retraining. Such practice is inefficient when a well-trained T2V model is already available or when there is a requirement to incorporate a new control modality into a pre-trained generator.

In this paper, we introduce `SparseCtrl`, an efficient approach that targets controlling text-to-video generation via temporally sparse condition maps with an add-on encoder. More specifically, in order to control the synthesis, we apply the philosophy of ControlNet [62], which implements an auxiliary encoder while preserving the integrity of the original generator. This design allows us to incorporate additional conditions by merely training the encoder network on top of the pre-trained T2V model, thereby eliminating the need for comprehensive model retraining. Additionally, this design facilitates control over not only the original T2V but also the derived personalized models when combined with the plug-and-play motion module of AnimateDiff [18]. To achieve this, we design a condition encoder equipped with temporal-aware layers that propagate the sparse condition signals from conditioned keyframes to unconditioned frames. Significantly, we find that purging the noised sample input in the vanilla ControlNet further prevents potential quality degradation in our scenario. Moreover, we apply widely used masking strategies [4, 8, 60, 61] during training to accommodate varying degrees of sparsity and tackle a broad range of application scenarios.

We evaluate `SparseCtrl` by training three encoders on sketches, depth, and RGB images. Experimental results show that users can manipulate the structure of the synthetic videos by providing just one or a few input condition maps. Comprehensive ablation studies are performed to investigate the contribution of each component. We additionally show that by integrating with plug-and-play video generation backbone such as AniamteDiff [18], our method exhibits compatibility and excellent visual quality with various personalized text-to-image models. Leveraging this sparse control approach, `SparseCtrl` enables a broad range of applications. For instance, the sketch encoder empowers users to transform hand-drawn storyboards into dynamic videos; The depth encoder provides the ability to render videos by supplying a minimum number of depth maps; Furthermore, the RGB image encoder unifies multiple tasks, including image animation, keyframe interpolation, video prediction, etc. We anticipate that this work will contribute towards bridging the gap between text-to-video research and real-world content creation processes.

2. Related Works

Text-to-video diffusion models. The field of text-to-video (T2V) generation [1, 6, 15, 16, 19, 23, 25, 37, 50, 64] has witnessed significant progression recently, driven by advancements in diffusion models [10, 20, 44, 45] and large-scale text-video paired datasets [2]. Initial attempts in this area focus on training a T2V model from scratch. For example, Video Diffusion Model [22] expands the standard image architecture to accommodate video data and trains on both image and video together. Imagen Video [21] employs a cascading structure for high-resolution T2V generation, while Make-A-Video [43] uses a text-image prior model to reduce reliance on text-video paired data. Oth-

ers turn to build T2V models upon powerful text-to-image (T2I) models such as Stable Diffusion [31], by incorporating additional layers to model cross-frame motion and consistency [14, 52, 68]. Among these, MagicVideo [68] utilizes a causal design and executes training in a compressed latent space to mitigate computational demands. Align-Your-Latents [4] efficiently turns T2I into video generators by aligning independently sampled noise maps. AnimateDiff [18] utilizes a pluggable motion module to enable high-quality animation creation on personalized image backbones [12, 27, 38, 39]. Other contributions include noise prior modeling [14], training on high-quality datasets [52], and latent-pixel hybrid space denoising [61], all leading to remarkable pixel quality. However, current text-conditioned video generation techniques lack fine-grained controllability over synthetic results. In response to this challenge, our work aims to enhance the control of T2V models via an add-on encoder.

Controllable text-to-video generation. Given that a text prompt can often result in ambiguous guidance to the video motion, content, and spatial structure, such controllabilities become crucial factors in T2V generation. For high-level video motion control, several studies propose learning LoRA [24] layers for specific motion patterns [18, 66], while others employ extracted trajectories [59], motion vectors [51], or pose sequence [28]. To manage specific synthetic keyframes for animation or interpolation, recent explorations include encoding the image separately to the generator [55], concatenating with the noise input [8, 51], or utilizing multi-level feature injection [59]. For fine-grained spatial structure control, some low-level representations are introduced. Gen-1 [11] is the first to use monocular depth sequences as structural guidance. VideoComposer [51] encodes sketch and depth sequences via a shared encoder, facilitating flexible combinations at inference. Additionally, some approaches utilize readily available image controlling models [30, 62] for controllable video generation [7, 18, 26, 65]. Though these methods achieve fine-grained controllability, they necessitate providing conditions for every synthetic frame, which incurs prohibitive costs in practical applications. In this study, we aim to control video generation through temporally *sparse* conditions by inputting only a few condition maps, thus making T2V more practical in a broader range of scenarios.

Add-on network for additional control. Training foundational T2I/T2V generative models is computationally demanding. Therefore, a preferred approach to incorporate extra control into these models is to train an additional condition encoder while maintaining the integrity of the original backbone [13, 56, 67]. ControlNet [62] pioneered the potential of training plug-and-play condition encoders for pre-trained T2I models. It involves creating a trainable duplicate of the pre-trained layers that accommodates the condition input. The encoder output is then reintegrated into the T2I model through zero-initialized layers. Similarly, T2I-Adapter [30] utilizes a lightweight structure to infuse control. IP-Adapter [58], integrates the style condition by transposing the reference image into supplementary embeddings, which are subsequently concatenated with the text embeddings. Our approach aligns with the principles of these works and aims to achieve sparse control through an auxiliary encoder module.

3. SparseCtrl

To enhance the controllability of a pre-trained text-to-video (T2V) model with temporally sparse signals, we introduce add-on sparse encoders to control the video generation process, leaving the original T2V generator untouched. This section is thus organized as follows: Sec. 3.1 presents the background of T2V diffusion models; Sec. 3.2 discuss the design of our sparse condition encoder, followed by the supported modalities and applications in Sec. 3.3.

3.1. Text-to-Video Diffusion Models

Leveraging powerful text-to-image generators. Text-to-image (T2I) generation has been dramatically advanced by powerful image generators like Stable Diffusion [35]. A practical path for T2V tasks is to leverage such powerful T2I priors. Recent T2V model [4, 14, 61] typically extend a pre-trained T2I generator for videos by incorporating temporal layers between the 2D image layers, as illustrated in the lower part of Fig. 2 (a). This arrangement enables cross-frame information exchange, thereby effectively modeling the cross-frame motion and temporal consistency.

Training objectives. The training objectives of T2V models are generally aligned with their image counterparts. Specifically, the model tries to predict the noise scale added to the clean RGB video (or latent features) $\boldsymbol{z_0^{1:N}}$ with N frames, encouraged by an MSE loss:

$$\mathbb{E}_{\boldsymbol{z_0^{1:N}},\boldsymbol{c_t},\boldsymbol{\epsilon},t}\left[\|\epsilon-\epsilon_\theta(\alpha_t\boldsymbol{z_0^{1:N}}+\sigma_t\boldsymbol{\epsilon},\boldsymbol{c_t},t)\|_2^2\right], \quad (1)$$

where $\boldsymbol{c_t}$ is the embeddings of the text description, $\boldsymbol{\epsilon}$ is the sampled Gaussian noise in the same shape of $\boldsymbol{z_0^{1:N}}$, α_t and σ_t are terms that control the added noise strength, $t = 1, ..., N$ is a uniformly sampled diffusion step, T is the number of total steps. In the following context, we also adopt this objective for training.

3.2. Sparse Condition Encoder

To enable efficient sparse control, we introduce an add-on encoder capable of accepting sparse condition maps as inputs, which we call sparse condition encoders. In the T2I domain, ControlNet [62] successfully adds structure control to the pre-trained image generator by partially replicating a copy of the pre-train model and its input, then adding the

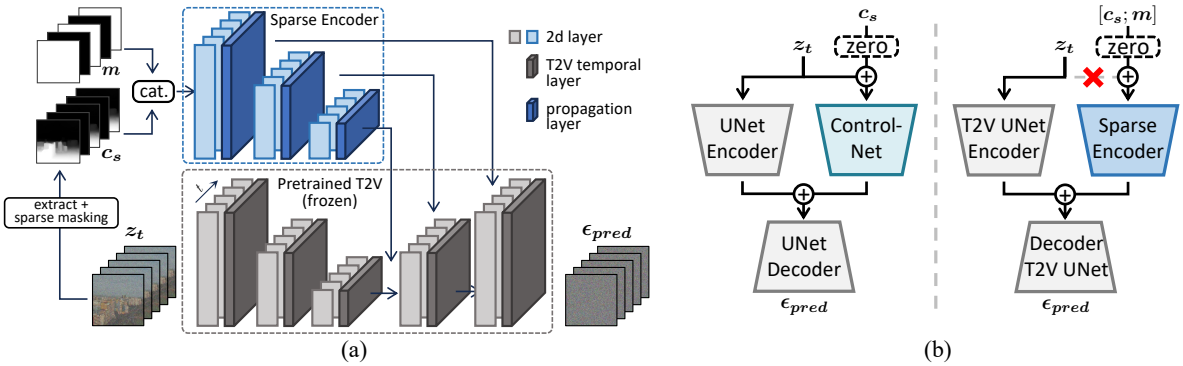

Figure 2. (a) Overview of the `SparseCtrl` pipeline. (b) Comparison between vanilla ControlNet (left) and our sparse condition encoder (right), where "zero" stands for zero-initialized layers; $[\cdot;\cdot]$ denotes channel-wise concatenation. Detailed structures are omitted for clarity.

conditions and reintegrating the output back to the original model through zero-initialized layers, as shown in the left of Fig. 2 (b). Inspired by its success, we start with a similar design to enable sparse control in the T2V setting.

Limited controllability of frame-wise encoder. We start with a straightforward solution: training a ControlNet-like encoder to incorporate sparse condition signals. To this end, we build a frame-wise encoder akin to ControlNet, replicate it across the temporal dimension, and add the conditions to the desired keyframes through this auxiliary structure. For frames that are not directly conditioned, we input a zero image to the encoder and indicate the unconditioned state through an additional mask channel. However, experimental results in Sec. 4.4.1 show that such frame-wise conditions sometimes fail to maintain temporal consistency when used with sparse input conditions, *e.g.*, in the image animation scenario where only the first frame is conditioned. In such cases, only keyframes react to the condition, leading to abrupt content changes between the conditioned and unconditioned frames.

Condition propagation across frames. Considering the sparsity and temporal relationship of given inputs, we hypothesize that the above problem arises because the T2V backbone has difficulty inferring the intermediate condition states for the unconditioned frames. To solve this, we propose to add temporal layers (*e.g.*, temporal attention [47] with position encoding) to the sparse condition encoders that allow the conditional signal to propagate from frame to frame. Intuitively, although not identical, different frames within a video clip share similarities in both appearance and structure. The temporal layers can thus propagate such implicit information from the conditioned keyframes to the unconditioned frames, thereby enhancing consistency. Our experiments confirm that this design significantly improves the robustness and consistency of the generated results.

Quality degradation caused by manually noised latents. Although the sparse condition encoder with temporal layers could tackle the sparsity of inputs, it sometimes leads to visual quality degradation of the generated videos, as shown in Sec. 4.4.1. When examining the design of the vanilla ControlNet, we find that simply applying the ControlNet in our scenario is unsuitable due to the copying of noised sample inputs. Concretely, as illustrated in Fig. 2 (b), the original ControlNet copies not only the UNet [36] encoder but also the noised sample input $\boldsymbol{z_t}$. Namely, the input for the ControlNet encoder is the sum between the condition (after zero-initialized layers) and the noised sample. This design stabilizes the training and accelerates the model convergence in its original scenario. However, in terms of the unconditioned frames in our setting, the informative input of the sparse encoder becomes only the noised sample. This might encourage the sparse encoder to overlook the condition maps and rely on the noised sample $\boldsymbol{z_t}$ during training, which contradicts our goal of controllability enhancement. Accordingly, as shown in Fig. 2 (b), our proposed sparse encoder eliminates the noised sample input and only accepts the condition maps $[\boldsymbol{c_s}, \boldsymbol{m}]$ after concatenation. This straightforward yet effective method eliminates the observed quality degradation in our experiments.

Unifying sparsity via masking. In practice, to unify different sparsity with a single model, we use zero images as the input placeholder for unconditioned frames and concatenate a binary mask sequence to the input conditions, which is a common practice in video reconstruction and prediction [4, 8, 46, 60, 61]. As shown in Fig. 2 (a), we concatenate a mask $\boldsymbol{m} \in \{0,1\}^{h\times w}$ channel-wise in addition to the condition signals $\boldsymbol{c_s}$ at each frame to form the input of the sparse encoder. Setting $\boldsymbol{m} = \boldsymbol{0}$ indicates the current frame is unconditioned and vice versa. In this way, different sparse input cases can be represented with a unified input format.

3.3. Multiple Modalities and Applications

In this paper, we implement `SparseCtrl` with three modalities: sketches, depth maps, and RGB images. Notably, our method is potentially compatible with other

modalities, such as skeleton and edge map, which we leave for future developments.

Sketch-to-video generation. Sketches [48, 49] can serve as an efficient guiding tool for T2V due to their ease of creation by non-professional users. With `SparseCtrl`, users can supply any number of sketches to shape the video content. For instance, a single sketch can establish the overall layout of the video, while sketches of the first, last, and selected intermediate frames can define coarse motion, making the method highly beneficial for storyboarding.

Depth guided generation. Integrating depth conditions with the pre-trained T2V enables depth-guided generation. Consequently, users can render a video by directly exporting sparse depth maps from engines or 3D representations [29] or conduct video translation using depth as an intermediate representation.

Image animation and transition; video prediction and interpolation. Within the context of RGB video, numerous tasks can be unified into a single problem of video generation with RGB image conditions. In this scheme, image animation corresponds to video generation conditioned on the first frame; Transition is conditioned by the first and last frames; Video prediction is conditioned on a small number of beginning frames; Interpolation is conditioned on uniformly sparsed keyframes.

4. Experiments

In this section, we evaluate `SparseCtrl` under various settings. Sec. 4.1 present the detailed implementations. Sec. 4.2 showcases the results and applications given one or few conditions. Sec. 4.3 suggests that `SparseCtrl` could achieve comparable performances on chosen popular tasks with baseline methods, *e.g.*, sparse depth-to-video generation and image animation. Sec. 4.4 present comprehensive ablation studies and evaluate `SparseCtrl`'s response to textual prompts and unrelated conditions.

4.1. Implementation Details

Text-to-video generator. We implement `SparseCtrl` upon AnimateDiff [18], which can serve as a general T2V generator when integrated with its pretraining image backbone, Stable Diffusion V1.5 [40], or function as a personalized generator when combined with personalized image backbones such as RealisticVision [42] and ToonYou [5]. We test with both settings and showcase the results.

Training. The training objective of `SparseCtrl` aligns with Eq. (1). The only difference is the integration of the proposed sparse condition encoder into the pre-trained text-to-video (T2V) backbone. To help the condition encoder learn robust controllability, we adopted a simple strategy to mask out conditions during training. In each iteration, we first randomly sample a number N_c between 1 and N to determine how many frames will receive the condition. Subsequently, we draw N_c indices without repeating from $\{1, 2, ..., N\}$ and keep the conditions for the corresponding frames. We train `SparseCtrl` on WebVid-10M [2] and extract the corresponding conditions on the fly. More details can be found in the supplementary material.

4.2. Main Results

We showcase the qualitative results and applications of `SparseCtrl` with three modalities in Fig. 1, 3, and the supplementary material, covering original and personalized T2V settings. As shown in the figure, with `SparseCtrl`, the synthetic videos closely adhere to control signals and maintain an excellent temporal consistency, being robust to different numbers of conditioning frames.

Remarkably, by drawing a single sketch, we can trigger the capability of the pre-trained T2V model to generate rare semantic compositions, such as a panda standing on a surfboard shown in the first row of Fig. 3. In contrast, the pre-trained T2V model struggles to generate such complex samples using textual descriptions alone. This suggests that the full potential of the T2V, pre-trained on large-scale datasets, may not be fully unlocked with only textual guidance. Additionally, we show that with well-learned real-world motion knowledge, the pre-trained T2V is capable of inferring the intermediate states with as few as two conditions, as illustrated in 3/5-th rows in Fig. 3. This indicates that temporally dense control might not be necessary.

4.3. Comparisons on Popular Tasks

Since it is challenging to compare `SparseCtrl` against prior efforts on all applications that we could enable, we choose two popular tasks for evaluation: sparse depth-to-video generation and image animation. For the first task, dense depth condition mode of VideoComposer (VC) [51] and Text2Video-Zero (Zero) [26] serve as the baseline. We also implement a baseline by combining AnimateDiff (AD) [18] with ControlNet [62] via applying frame-wise control signals to the conditioned keyframes. For image animation, we compare `SparseCtrl` against two open-sourced image animation baselines: DynamiCrafter (DC) [55] and VideoComposer's initial frame mode.

4.3.1 Sparse Depth-to-Video Generation

Providing a dense depth sequence for video generation helps specify structural information to some extent. We thus evaluate our method on this task with much more challenging yet practical settings: only a few depths are given for the synthesis. The controlling fidelity under different sparsity of input is measured for the quantitative comparison. Specifically, we first select 20 videos from the validation set of WebVid-10M [2] that are not seen during training. There-

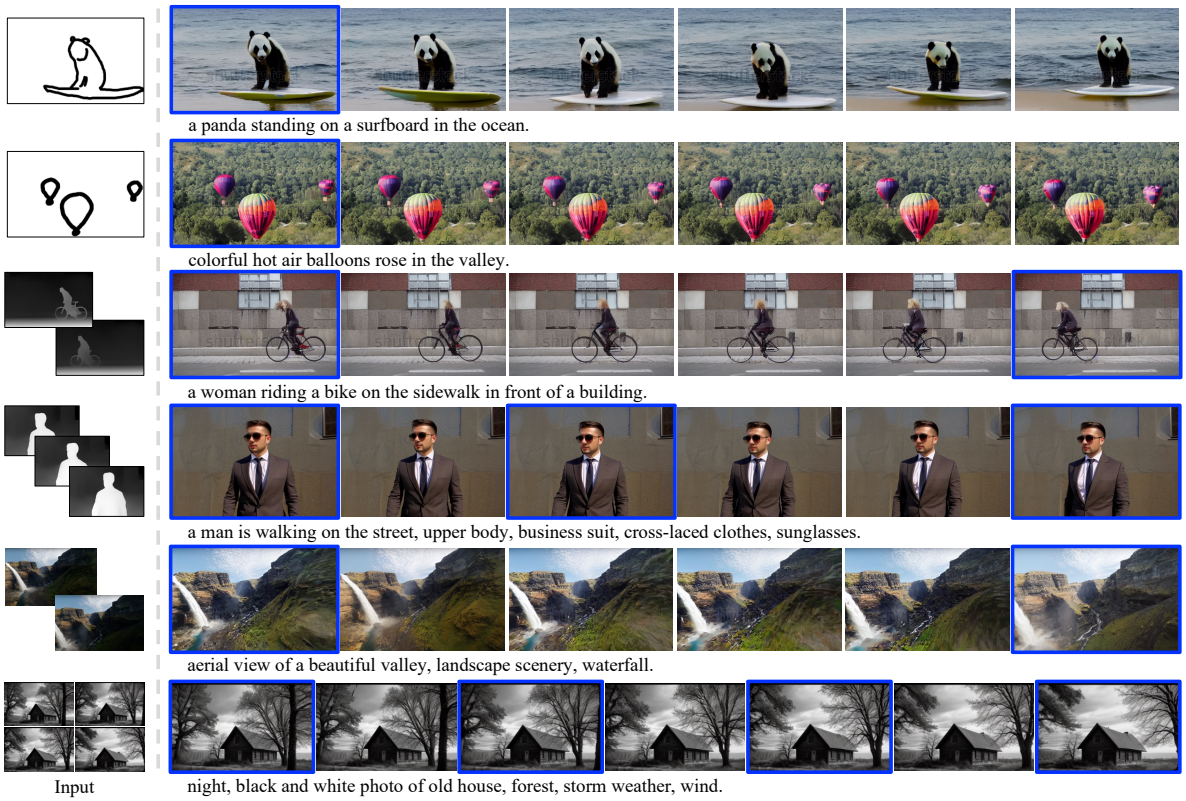

Figure 3. Qualitative results with sketch/depth/RGB image sparse condition encoders. Videos in 4/6-th rows are generated with personalized backbone, RealisticVision [42]. The input conditions are shown on the left; the conditioned keyframes are denoted by ***blue*** border.

Table 1. Evaluation on sparse control fidelity. "err." stands for MAE error; "cons." stands for temporal consistency. All numbers are scaled up $100\times$.

r_{mask}	0		1/2		3/4		7/8	
	err. (↓)	cons. (↑)	err.	cons.	err.	cons.	err.	cons.
VC [51]	8.26	96.02	-	-	-	-	-	-
Zero [26]	8.24	97.05	-	-	-	-	-	-
AD [18, 62]	8.37	96.82	9.25	96.68	12.38	93.35	14.84	94.66
Ours	8.92	96.54	8.09	96.75	7.30	96.48	7.40	95.56

after, we estimate the corresponding depth sequences with the off-the-shelf MiDaS [34] model, evenly mask out some of them with a ratio r_{mask}, and use the remaining depth maps as conditions to generate videos. We then estimate the depth maps from the conditioned keyframes in generated videos and, following the metrics in previous work, we perform scale shift realignment and compute the mean absolute error (MAE) against the depth maps extracted from the original videos. On the other hand, to prevent the model from learning a shot cut by solely controlling the keyframes and ignoring temporal consistency, we also report cross-frame CLIP [32] similarity following previous works [26, 53].

The quantitative results are shown in Tab. 1. To stay close to the original implementation, we only report results of $r_{mask} = 0$ for VideoComposer and Text2Video-Zero, where the controls for every frame are provided. As shown in the table, as the control sparsity, *i.e.*, the masking rate r_{mask}, increases, our method maintains a comparable error rate with dense control baselines. In contrast, the error of AnimateDiff with frame-wise ControlNet increases, indicating that this baseline method tends to ignore the condition signals when the control becomes sparser.

4.3.2 Image animation

By providing the RGB image as the first frame condition, `SparseCtrl` can handle the task of image animation. To validate our method's effectiveness, we further compare it with two baselines in this domain. We collect eight in-the-wild images and animate them using the three methods to generate 24 samples in total. Similar in Sec. 4.3.1, our metrics lie in two aspects: the first frame fidelity to the input image measured by LPIPS [63], and temporal consistency measured by CLIP similarity. Additionally, we invited 20 users to rank the results individually in terms of the fidelity to the given image and the overall quality preference. We obtained 160 ranking results for each aspect. We use average human ranking (AHR) as a preference metric and report

Figure 4. Ablation study on network design. *Left*: the results of wild image animation with pre-trained T2V; *Right*: the results of *in-domain* image animation with personalized T2I backbone ToonYou [5], where the input image is generated by the corresponding image model. The input conditions are shown on the left; the conditioned keyframes are denoted by ***blue*** border.

Table 2. Evaluation of image animation.

	LPIPS (↓)	CLIP (↑)	fidelity(user) (↑)	preference(user) (↑)
DC [55]	0.5346	98.49	2.137	2.310
VC [51]	0.3346	91.90	1.815	1.696
Ours	0.1467	95.25	2.048	1.994

the results in Tab. 2. The result shows that our method can achieve comparable performance with specifically designed animation pipelines while being favored in terms of fidelity to the first frame.

4.4. Ablative Study

4.4.1 Design of Sparse Encoder

We ablate on the sparse encoder architecture to verify our choice. Specifically, we experiment with four designs: **(1) frame-wise condition encoder**, where we repeat the 2D ControlNet [62] along the temporal axis and encode the control signals to the keyframes, as depicted in Sec. 3.2; **(2) condition encoder with propagation layers**, where we add temporal layers upon (1) to propagate conditions across frames, as discussed in Sec. 3.2; **(3) our full model**, where we further eliminate the noised sample input to the condition encoder in (2). To better compare the effectiveness of these three choices, we consider the most challenging case, *i.e.*, the RGB image conditions, because compared to other abstract modalities, here the synthetic results need to faithfully demonstrate the fine-grained details of the condition signal and propagate it to other unconditioned frames to ensure temporal consistency. With AnimateDiff [18], we additionally show the result on personalized image backbone, which further assists us in distinguishing the merits and shortcomings of different choices.

In Fig. 4, we show the qualitative image animation results. According to the figure, with all three variations, the first frame in the generated videos is fidelity to the input image control. The frame-wise encoder, under the personalized generation setting, fails to propagate the control to the unconditioned frames (1st row, right), leading to temporal inconsistency where the details of the character (*e.g.*, hair, and clothes color) change over time. Upon the pre-trained T2V, the encoder with propagation layers, as stated in Sec. 3.2, suffers quality degradation (2nd row, left), and we hypothesize that this is because the noised sample input to the encoder provides misleading information for the condition tasks. Finally, with propagation layers and eliminating the noised sample input, our full model works well under the two settings (3rd row), maintaining both fidelity to condition and temporal consistency.

4.4.2 Unrelated Conditions

Besides the common usages, we experiment with an extreme case where the input conditions are unrelated or contradicted. Regarding this, we input two unrelated images to the RGB image encoder and require the model to interpolate between them, as shown in the first-row in Fig. 5. Surprisingly, the sparse encoder can still help generate smooth transitions between the input images, which further verifies the robustness of the `SparseCtrl` and shows potential in visual effects synthesis.

4.4.3 Response to Textual Prompt

Another interesting question is, with the additional information provided by the sparse condition encoder, to what extent does the final generated outcome respond to the input text description? To answer this, we experiment with different textual prompts with the same input and demonstrate the results in Fig. 5. In the image animation setting, we compare the prompt that faithfully describes the image content (2nd row) and the prompt that describes a slightly different content (3rd row). The results show that the input text prompts do influence the outcome by leading the

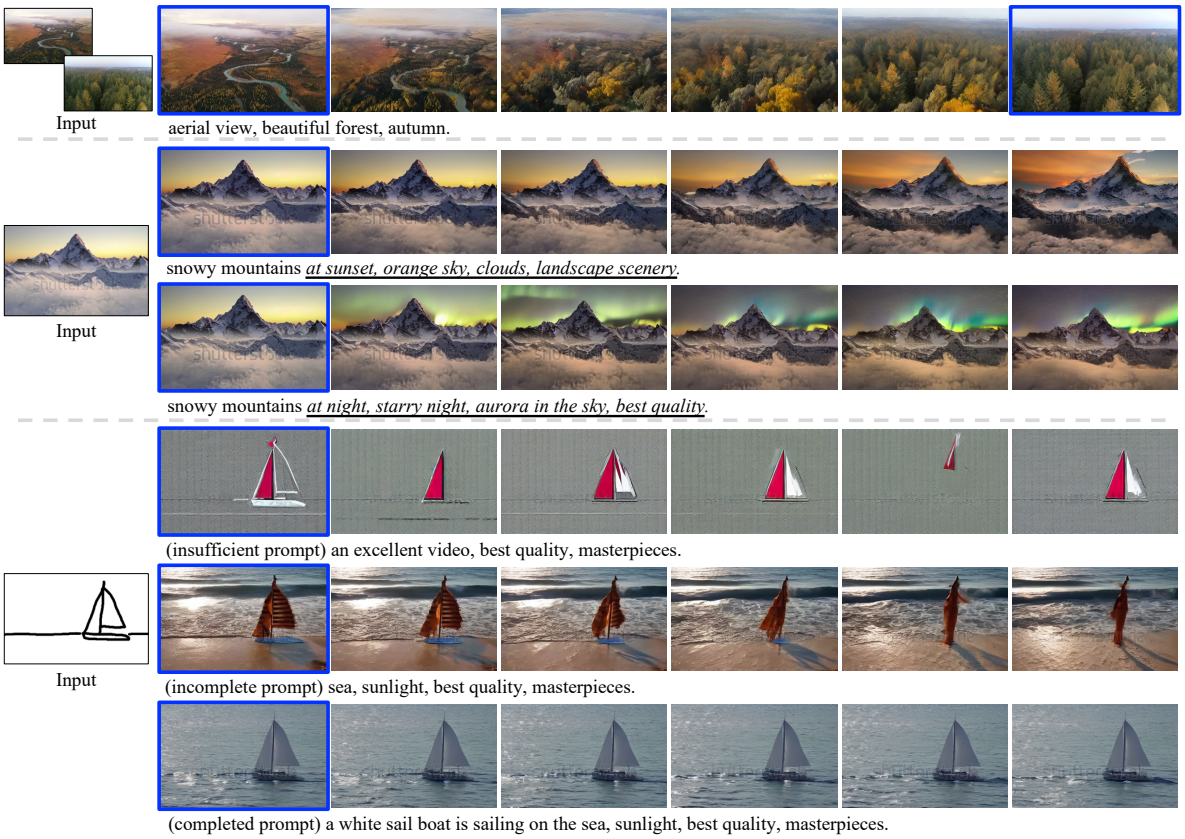

Figure 5. Ablation study on unrelated conditions and response to textual prompt. The first row demonstrates how the model deals with unrelated conditions; The lower five rows show how the model reacts to different textual prompts. The input conditions are shown on the left; the conditioned keyframes are denoted by ***blue*** border.

contents towards the corresponding directions.

In the sketch-to-video setting, we construct three types of prompts: (1) insufficient prompt with no useful information (4th row), *e.g.*, "*an excellent video, best quality, masterpieces*"; (2) incomplete prompt that partially describes the desired content (5th row), *e.g.*, "*sea, sunlight, ...*", ignoring the central object "sailboat"; (3) completed prompt that describes every content (6th row). As shown in Fig. 5, with the sketch condition, the content can be properly generated only when the prompt is completed, showing that the text input still plays a significant role when the provided condition is highly abstract and insufficient to infer the content.

5. Discussion and Conclusion

We present `SparseCtrl`, a unified approach of adding temporally sparse controls to pre-trained text-to-video generators via an add-on encoder network. It can accommodate various modalities, including depth, sketches, and RGB images, greatly enhancing practical control for video generation. This flexibility proves invaluable in diverse applications like sketch-to-video, image animation, keyframe interpolation, etc. Extensive experiments have validated method's effectiveness and generalizability across original and personalized text-to-video generators, making it a promising tool for real-world usage.

Limitations. Though with `SparseCtrl`, the visual quality, semantic composition ability, and domain of the generated results are limited by the pre-trained T2V backbone and the training data. In experiments, we find that the failure cases mostly come from out-of-domain input, such as anime image animation, since such data is scarce in the T2V and sparse encoder's pre-training dataset WebVid-10M [2], whose contents are mainly real-world videos. Possible solutions for enhancing the generalizability could be improving the training dataset's domain diversity and utilizing some domain-specific backbone, such as integrating `SparseCtrl` with AnimateDiff [18].

Acknowledgement. The project is supported by the Shanghai Artificial Intelligence Laboratory (P23KN00601, P23KS00020, 2022ZD0160201), CUHK Interdisciplinary AI Research Institute, and the Centre for Perceptual and Interactive Intelligence (CPII) Ltd under the Innovation and Technology Commission (ITC)'s InnoHK.

References

[1] Jie An, Songyang Zhang, Harry Yang, Sonal Gupta, Jia-Bin Huang, Jiebo Luo, and Xi Yin. Latent-shift: Latent diffusion with temporal shift for efficient text-to-video generation. *arXiv preprint arXiv:2304.08477*, 2023. 2

[2] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1728–1738, 2021. 2, 5, 8

[3] Yogesh Balaji, Seungjun Nah, Xun Huang, Arash Vahdat, Jiaming Song, Karsten Kreis, Miika Aittala, Timo Aila, Samuli Laine, Bryan Catanzaro, et al. ediffi: Text-to-image diffusion models with an ensemble of expert denoisers. *arXiv preprint arXiv:2211.01324*, 2022. 2

[4] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22563–22575, 2023. 2, 3, 4

[5] Bradcatt. Toonyou, https://civitai.com/models/30240/toonyou, 2023. 1, 5, 7

[6] Haoxin Chen, Menghan Xia, Yingqing He, Yong Zhang, Xiaodong Cun, Shaoshu Yang, Jinbo Xing, Yaofang Liu, Qifeng Chen, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter1: Open diffusion models for high-quality video generation, 2023. 2

[7] Weifeng Chen, Jie Wu, Pan Xie, Hefeng Wu, 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

[8] Xinyuan Chen, Yaohui Wang, Lingjun Zhang, Shaobin Zhuang, Xin Ma, Jiashuo Yu, Yali Wang, Dahua Lin, Yu Qiao, and Ziwei Liu. Seine: Short-to-long video diffusion model for generative transition and prediction, 2023. 2, 3, 4

[9] Xiaoliang Dai, Ji Hou, Chih-Yao Ma, Sam Tsai, Jialiang Wang, Rui Wang, Peizhao Zhang, Simon Vandenhende, Xiaofang Wang, Abhimanyu Dubey, et al. Emu: Enhancing image generation models using photogenic needles in a haystack. *arXiv preprint arXiv:2309.15807*, 2023. 2

[10] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021. 2

[11] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. *arXiv preprint arXiv:2302.03011*, 2023. 2, 3

[12] Rinon Gal, Yuval Alaluf, Yuval Atzmon, Or Patashnik, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. An image is worth one word: Personalizing text-to-image generation using textual inversion. *arXiv preprint arXiv:2208.01618*, 2022. 3

[13] Rinon Gal, Moab Arar, Yuval Atzmon, Amit H Bermano, Gal Chechik, and Daniel Cohen-Or. Encoder-based domain tuning for fast personalization of text-to-image models. *ACM Transactions on Graphics (TOG)*, 42(4):1–13, 2023. 3

[14] Songwei Ge, Seungjun Nah, Guilin Liu, Tyler Poon, Andrew Tao, Bryan Catanzaro, David Jacobs, Jia-Bin Huang, Ming-Yu Liu, and Yogesh Balaji. Preserve your own correlation: A noise prior for video diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22930–22941, 2023. 3

[15] Rohit Girdhar, Mannat Singh, Andrew Brown, Quentin Duval, Samaneh Azadi, Sai Saketh Rambhatla, Akbar Shah, Xi Yin, Devi Parikh, and Ishan Misra. Emu video: Factorizing text-to-video generation by explicit image conditioning. *arXiv preprint arXiv:2311.10709*, 2023. 2

[16] Jiaxi Gu, Shicong Wang, Haoyu Zhao, Tianyi Lu, Xing Zhang, Zuxuan Wu, Songcen Xu, Wei Zhang, Yu-Gang Jiang, and Hang Xu. Reuse and diffuse: Iterative denoising for text-to-video generation. *arXiv preprint arXiv:2309.03549*, 2023. 2

[17] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan, and Baining Guo. Vector quantized diffusion model for text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022. 2

[18] Yuwei Guo, Ceyuan Yang, Anyi Rao, Yaohui Wang, Yu Qiao, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023. 1, 2, 3, 5, 6, 7, 8

[19] Yingqing He, Tianyu Yang, Yong Zhang, Ying Shan, and Qifeng Chen. Latent video diffusion models for high-fidelity video generation with arbitrary lengths. *arXiv preprint arXiv:2211.13221*, 2022. 2

[20] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 2

[21] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 2

[22] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022. 2

[23] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022. 2

[24] 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. *arXiv preprint arXiv:2106.09685*, 2021. 3

[25] Johanna Karras, Aleksander Holynski, Ting-Chun Wang, and Ira Kemelmacher-Shlizerman. Dreampose: Fashion video synthesis with stable diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 22680–22690, 2023. 2

[26] Levon Khachatryan, Andranik Movsisyan, Vahram Tadevosyan, Roberto Henschel, Zhangyang Wang, Shant Navasardyan, and Humphrey Shi. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. *IEEE*

International Conference on Computer Vision (ICCV), 2023. 2, 3, 5, 6

[27] Nupur Kumari, Bingliang Zhang, Richard Zhang, Eli Shechtman, and Jun-Yan Zhu. Multi-concept customization of text-to-image diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1931–1941, 2023. 2, 3

[28] Yue Ma, Yingqing He, Xiaodong Cun, Xintao Wang, Ying Shan, Xiu Li, and Qifeng Chen. Follow your pose: Pose-guided text-to-video generation using pose-free videos. *arXiv preprint arXiv:2304.01186*, 2023. 3

[29] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 5

[30] Chong Mou, Xintao Wang, Liangbin Xie, Jian Zhang, Zhongang Qi, Ying Shan, and Xiaohu Qie. T2i-adapter: Learning adapters to dig out more controllable ability for text-to-image diffusion models. *arXiv preprint arXiv:2302.08453*, 2023. 3

[31] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. Sdxl: improving latent diffusion models for high-resolution image synthesis. *arXiv preprint arXiv:2307.01952*, 2023. 2, 3

[32] 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

[33] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1 (2):3, 2022. 2

[34] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1623–1637, 2020. 6

[35] 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

[36] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 4

[37] Ludan Ruan, Yiyang Ma, Huan Yang, Huiguo He, Bei Liu, Jianlong Fu, Nicholas Jing Yuan, Qin Jin, and Baining Guo. Mm-diffusion: Learning multi-modal diffusion models for joint audio and video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10219–10228, 2023. 2

[38] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Yael Pritch, Michael Rubinstein, and Kfir Aberman. Dreambooth: Fine tuning text-to-image diffusion models for subject-driven generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22500–22510, 2023. 3

[39] Nataniel Ruiz, Yuanzhen Li, Varun Jampani, Wei Wei, Tingbo Hou, Yael Pritch, Neal Wadhwa, Michael Rubinstein, and Kfir Aberman. Hyperdreambooth: Hypernetworks for fast personalization of text-to-image models. *arXiv preprint arXiv:2307.06949*, 2023. 3

[40] runwayml. Stable diffusion v1.5, https://huggingface.co/runwayml/stable-diffusion-v1-5, 2022. 5

[41] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems*, 35:36479–36494, 2022. 2

[42] SG_161222. Realistic vision v5.1, https://civitai.com/models/4201/realistic-vision-v51, 2023. 1, 5, 6

[43] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 2

[44] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, pages 2256–2265. PMLR, 2015. 2

[45] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. 2

[46] Zhan Tong, Yibing Song, Jue Wang, and Limin Wang. Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *Advances in neural information processing systems*, 35:10078–10093, 2022. 4

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 4

[48] Yael Vinker, Yuval Alaluf, Daniel Cohen-Or, and Ariel Shamir. Clipascene: Scene sketching with different types and levels of abstraction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4146–4156, 2023. 5

[49] Andrey Voynov, Kfir Aberman, and Daniel Cohen-Or. Sketch-guided text-to-image diffusion models. In *ACM SIGGRAPH 2023 Conference Proceedings*, pages 1–11, 2023. 5

[50] Jiuniu Wang, Hangjie Yuan, Dayou Chen, Yingya Zhang, Xiang Wang, and Shiwei Zhang. Modelscope text-to-video technical report. *arXiv preprint arXiv:2308.06571*, 2023. 2

[51] Xiang Wang, Hangjie Yuan, Shiwei Zhang, Dayou Chen, Jiuniu Wang, Yingya Zhang, Yujun Shen, Deli Zhao, and Jingren Zhou. Videocomposer: Compositional video synthesis with motion controllability. *arXiv preprint arXiv:2306.02018*, 2023. 2, 3, 5, 6, 7

[52] Yaohui Wang, Xinyuan Chen, Xin Ma, Shangchen Zhou, Ziqi Huang, Yi Wang, Ceyuan Yang, Yinan He, Jiashuo Yu, Peiqing Yang, Yuwei Guo, Tianxing Wu, Chenyang Si, Yuming Jiang, Cunjian Chen, Chen Change Loy, Bo Dai, Dahua Lin, Yu Qiao, and Ziwei Liu. Lavie: High-quality video generation with cascaded latent diffusion models, 2023. 3

[53] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Stan Weixian Lei, Yuchao Gu, Yufei Shi, Wynne Hsu, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Tune-a-video: One-shot tuning of image diffusion models for text-to-video generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7623–7633, 2023. 6

[54] Qiucheng Wu, Yujian Liu, Handong Zhao, Trung Bui, Zhe Lin, Yang Zhang, and Shiyu Chang. Harnessing the spatial-temporal attention of diffusion models for high-fidelity text-to-image synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7766–7776, 2023. 2

[55] Jinbo Xing, Menghan Xia, Yong Zhang, Haoxin Chen, Xintao Wang, Tien-Tsin Wong, and Ying Shan. Dynamicrafter: Animating open-domain images with video diffusion priors. *arXiv preprint arXiv:2310.12190*, 2023. 3, 5, 7

[56] Xingqian Xu, Jiayi Guo, Zhangyang Wang, Gao Huang, Irfan Essa, and Humphrey Shi. Prompt-free diffusion: Taking" text" out of text-to-image diffusion models. *arXiv preprint arXiv:2305.16223*, 2023. 3

[57] Xingqian Xu, Zhangyang Wang, Gong Zhang, Kai Wang, and Humphrey Shi. Versatile diffusion: Text, images and variations all in one diffusion model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7754–7765, 2023. 2

[58] Hu Ye, Jun Zhang, Sibo Liu, Xiao Han, and Wei Yang. Ip-adapter: Text compatible image prompt adapter for text-to-image diffusion models. *arXiv preprint arXiv:2308.06721*, 2023. 3

[59] Shengming Yin, Chenfei Wu, Jian Liang, Jie Shi, Houqiang Li, Gong Ming, and Nan Duan. Dragnuwa: Fine-grained control in video generation by integrating text, image, and trajectory. *arXiv preprint arXiv:2308.08089*, 2023. 2, 3

[60] Lijun Yu, Yong Cheng, Kihyuk Sohn, José Lezama, Han Zhang, Huiwen Chang, Alexander G Hauptmann, Ming-Hsuan Yang, Yuan Hao, Irfan Essa, et al. Magvit: Masked generative video transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10459–10469, 2023. 2, 4

[61] David Junhao Zhang, Jay Zhangjie Wu, Jia-Wei Liu, Rui Zhao, Lingmin Ran, Yuchao Gu, Difei Gao, and Mike Zheng Shou. Show-1: Marrying pixel and latent diffusion models for text-to-video generation. *arXiv preprint arXiv:2309.15818*, 2023. 2, 3, 4

[62] 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. 2, 3, 5, 6, 7

[63] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6

[64] Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023. 2

[65] Yabo Zhang, Yuxiang Wei, Dongsheng Jiang, Xiaopeng Zhang, Wangmeng Zuo, and Qi Tian. Controlvideo: Training-free controllable text-to-video generation. *arXiv preprint arXiv:2305.13077*, 2023. 2, 3

[66] Rui Zhao, Yuchao Gu, Jay Zhangjie Wu, David Junhao Zhang, Jiawei Liu, Weijia Wu, Jussi Keppo, and Mike Zheng Shou. Motiondirector: Motion customization of text-to-video diffusion models. *arXiv preprint arXiv:2310.08465*, 2023. 3

[67] Shihao Zhao, Dongdong Chen, Yen-Chun Chen, Jianmin Bao, Shaozhe Hao, Lu Yuan, and Kwan-Yee K Wong. Uni-controlnet: All-in-one control to text-to-image diffusion models. *arXiv preprint arXiv:2305.16322*, 2023. 3

[68] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv preprint arXiv:2211.11018*, 2022. 3