CoCoCo: Improving Text-Guided Video Inpainting for Better Consistency, Controllability and Compatibility

Bojia Zi[1], Shihao Zhao[2], Xianbiao Qi[*3], Jianan Wang[3], Yukai Shi[3], Qianyu Chen[1], Bin Liang[1], Rong Xiao[4], Kam-Fai Wong[1], Lei Zhang[3]

[1]The Chinese University of Hong Kong,
[2]The University of Hong Kong,
[3]International Digital Economy Academy,
[4]IntelliFusion Inc.
{bjzi, qychen, kfwong}@se.cuhk.edu.hk, bin.liang@cuhk.edu.hk, shzhao@cs.hku.hk,
{qixianbiao, rongxiao, wendyjnwang}@gmail.com, {shiyukai, zhanglei}@idea.edu.cn

Abstract

Video inpainting is a crucial task with diverse applications, including fine-grained video editing, video recovery, and video dewatermarking. However, most existing video inpainting methods primarily focus on visual content completion while neglecting text information. There are only a limited number of text-guided video inpainting techniques, and these techniques struggle with maintaining visual quality and exhibit poor semantic representation capabilities. In this paper, we introduce **CoCoCo** a text-guided video inpainting diffusion framework. To address the aforementioned challenges, we enhance both the training data and model structure. Specifically, we devise an instance-aware region selection strategy for masked area sampling and develop a novel motion block that incorporates efficient 3D full attention and textual cross attention. Additionally, our **CoCoCo** framework can be seamlessly integrated with various personalized text-to-image diffusion models through a delicate training-free transfer mechanism. Comprehensive experiments demonstrate that **CoCoCo** can create high-quality visual content with enhanced temporal consistency, improved text controllability, and better compatibility with personalized image models.

Code — https://github.com/zibojia/COCOCO
Extended version — https://arxiv.org/abs/2403.12035

Introduction

With the advancements in diffusion models, the realm of video generation has witnessed remarkable progress (Ho et al. 2022b; Singer et al. 2022; Wang et al. 2023a; Pika Labs 2023; Brooks et al. 2024). Techniques such as VideoCrafter(Chen et al. 2024) and AnimateDiff(Guo et al. 2023) have demonstrated exceptional capabilities in generating videos. Recent groundbreaking works like Sora(Brooks et al. 2024) and Gen-3(Gen-3 2024) have elevated the field to new heights. Beyond these video generation models, video inpainting (Kim et al. 2019; Lee et al. 2019; Xu et al. 2019; Zeng, Fu, and Chao 2020; Liu et al. 2021; Ouyang, Wang, and Chen 2021; Li et al. 2022; Zhou et al. 2023) has also experienced a surge in development recently. As a pivotal subtask within video generation, video inpainting involves restoring or replacing missing or corrupted regions within a video clip, given specified masks. This technology boasts a diverse array of applications, encompassing video completion, video recovery, and video dewatermarking (Zhou et al. 2023).

Video inpainting methods can be summarised as two types: unconditional and conditional. The unconditional approach primarily aims to restore corrupted visual content within a video based solely on its background or surrounding information (Kim et al. 2019; Lee et al. 2019; Li et al. 2022; Zhou et al. 2023). Conversely, conditional video inpainting relies on additional information, such as semantic details (Zhang et al. 2023b), and sketches (Wang et al. 2023c), to generate visual content within specified masked regions across frames. Notably, text prompts emerge as a pivotal condition, since they are more intuitive and easily obtainable. Furthermore, the recent advancements in text-guided image inpainting (Ju et al. 2024) and text-to-video generation (Wang et al. 2023a; Guo et al. 2023; Yang et al. 2024) have significantly motivated our exploration into text-guided video inpainting.

Regarding the field of text-guided video inpainting, prior research has made initial explorations. VideoComposer (Wang et al. 2023c), for example, has focused on creating a unified video generation model that can simultaneously receive various conditions to control video generation. Among these conditions lies the utilization of mask conditions in conjunction with text conditions to achieve text-guided video inpainting. AVID (Zhang et al. 2023b) is another well-established method for text-guided video inpainting, which introduces a motion module with temporal capability into the image inpainting model, and leverages the image inpainting model's ability to control text to achieve text-guided video inpainting.

Now, we rethink the task of text-guided video inpainting. Compared to the well-studied text-to-video generation task, text-guided video inpainting faces much more challenges. Unlike text-to-video generation, video inpainting not only requires consistency across frames, but also consistency be-

*Corresponding Author

Figure 1: Results of our **CoCoCo** . *To view videos, please check our extended version.*

tween the inpainted region and its surroundings. Meanwhile, the prompt should match the masked region instead of the entire video, thus placing more stringent requirements on the text conditions. With this in mind, the existing VideoComposer and AVID models exhibit some problems. Firstly, they only use spatial-temporal decomposition, preventing effective interaction between spatial and temporal information and impacting visual consistency. Additionally, their motion module lacks guidance from text, significantly weakening the model's ability to control the results in text-guided inpainting. And these issues have been verified in our subsequent experimental sections.

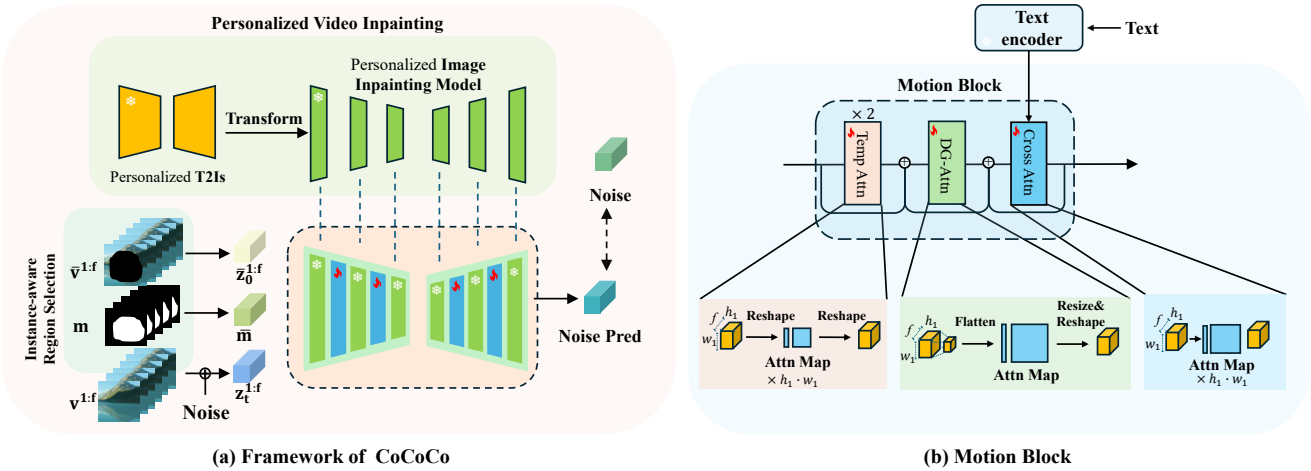
To mitigate the aforementioned drawbacks, we introduce **CoCoCo** a novel framework to improve text-guided video inpainting for better temporal consistency, text controllability, and compatibility with text-to-image models. Drawing inspiration from the design of text-to-video diffusion models like AnimateDiff (Guo et al. 2023), which leverages pretrained text-to-image models by freezing their weights and incorporating a motion module for video generation, we have made improvements in both the training data and model structure. Specifically, we design a more fine-grained mask region selection strategy, termed instance-aware region selection, instead of the widely used random masking, which allows the text information to better correspond to the inpainting regions. Additionally, we improve the motion module by 1) designing a damped global attention layer, an efficient full 3D attention mechanism, and 2) introducing more textual information through cross-attention. To better leverage the developments in the image generation field, we further design a training-free strategy, inspired by the task vector concept (Ilharco et al. 2022), to make various personalized text-to-image models or LoRAs (Hu et al. 2021) compatible with our video inpainting model. This allows users to create customized content in the masked regions of the given video.

Our contributions can be summarised as following:

- We propose a novel text-guided video inpainting framework, called **CoCoCo** . Compared to existing methods, **CoCoCo** exhibits better temporal consistency and more powerful text-controllability. Moreover, through our training-free transfer design, **CoCoCo** can be effectively integrated with various personalized text-to-image models, significantly expanding its usability and versatility.
- We have redesigned the training data paradigm for the video inpainting task, adopting an instance-aware region selection approach to better align the training videos and their captions. Additionally, we have designed a novel motion module, incorporating an efficient full 3D attention mechanism and leveraging more textual information, to enhance motion consistency and text-video alignment.
- Through extensive visualizations, quantitative comparisons, and experimental analyses, we have thoroughly demonstrated the performance of our **CoCoCo** framework. We have also conducted detailed ablation studies to analyze and verify the contributions of each of the modules we have proposed.

Related Work

Video Generation. Recently, numerous video generation methods have emerged (Ho et al. 2022a; Esser et al. 2023; Brooks et al. 2024; Khachatryan et al. 2023; Blattmann et al. 2023a; Kondratyuk et al. 2023; Chen et al. 2024; Xing et al. 2023; Guo et al. 2023; Chen et al. 2023; Yin et al. 2023; Chen et al. 2023; Wang et al. 2023a; Fan et al. 2023; Zhang et al. 2023a; Jiang et al. 2023; Ma et al. 2024; Xie et al. 2023; Ma et al. 2023; Yang et al. 2024; Bao et al. 2024). Closed-source products, such as Sora (Brooks et al. 2024), Pika (Pika Labs 2023), VideoPoet (Kondratyuk et al. 2023), Luma (Luma 2024), Kling (Kling 2024), and Gen-3(Gen-3 2024), deliver impressive visual results with high resolution and long durations. However, the details of their methods and training data remain undisclosed to the public. Open-source methods have also witnessed rapid advancements. Tune-A-Video (Wu et al. 2023) achieves zero-shot video generation by adapting a small proportion of parameters and making minor architectural modifications to image diffusion models. ModelScope (Wang et al. 2023a) introduces a text-to-video synthesis model that incorporates spatial-temporal decomposition blocks to maintain consistency and achieve smooth movement transitions. AnimateDiff (Guo et al. 2023), building upon existing text-to-image



(a) Framework of CoCoCo (b) Motion Block

Figure 2: The overall framework of our proposed **CoCoCo**. Subfigure (a) shows the framework of **CoCoCo**, Subfigure (b) illustrates the motion block.

models, introduces a motion module to enable video generation, effectively leveraging existing personalized models like LoRAs (Hu et al. 2021) and Dreambooths (Ruiz et al. 2023). Furthermore, CogVideoX (Yang et al. 2024), which is based on transformer diffusion models (Peebles and Xie 2023), utilizes 3D full attention and a 3D VAE, delivering impressive visual results with remarkable frame consistency and text-alignment capabilities at a new level.

Text-Guided Image Inpainting. Text-Guided image inpainting (Andonian et al. 2021; Avrahami, Lischinski, and Fried 2022; Ding et al. 2022; Couairon et al. 2022; Wang et al. 2023b; Ju et al. 2024) is a technique that fills in missing or damaged areas of an image based on a textual description provided by the user. Blended Diffusion (Avrahami, Fried, and Lischinski 2022), an early work exploring this domain using diffusion models, executes CLIP-guided diffusion processes concurrently on both foreground and background, merging the results through element-wise aggregation. DiffEdit (Couairon et al. 2022) innovates with a "masked yet mask-free" approach, simultaneously performing masking segmentation and masked diffusion to achieve seamless masked inpainting. Additionally, Imagen Editor (Wang et al. 2023b) utilizes a cascaded diffusion model for image inpainting by fine-tuning an Imagen model. Beyond these methods, the widely used Stable Diffusion (Blattmann et al. 2023b) also has its own inpainting model, Stable Diffusion Inpainting (Blattmann et al. 2023b), which is fine-tuned on Stable Diffusion for inpainting tasks and delivers impressive results. This model, based on Stable Diffusion, modifies the input channel size from 4 to 9, adding 5 extra input channels (4 for the encoded masked image and 1 for the mask itself), to predict the noise for the entire image. We adopts Stable Diffusion Inpainting as the base model for our **CoCoCo**.

Text-Guided Video Inpainting. Video inpainting aims to complete corrupted regions in videos, and can be categorized into two types: unconditional and conditional. Unconditional video inpainting (Kim et al. 2019; Lee et al. 2019; Liu et al. 2021; Xu et al. 2019; Li et al. 2022; Ouyang, Wang, and Chen 2021; Zeng, Fu, and Chao 2020; Zhou et al. 2023) relies solely on the video information itself for completion without receiving any additional input. This paper focuses on conditional video inpainting, specifically text-conditioned inpainting. VideoComposer (Wang et al. 2023c), a unified video generation and editing method, incorporates text-guided video inpainting as one of its subtasks. It employs a 3D U-Net and concatenates the noise and mask sequences as input. Furthermore, AVID (Zhang et al. 2023b), a recently proposed text-guided video inpainting method based on diffusion models, follows AnimateDiff (Guo et al. 2023) by initializing the image module with an image inpainting model and fine-tuning the motion module. Consequently, AVID inherits the capabilities of the base text-to-image model, enabling text-based control over the inpainting task. However, both methods suffer from inconsistencies between foreground and background, poor visual quality, and limitations in text alignment. Their reliance on spatial-temporal decomposition hinders effective interaction between spatial and temporal information, impacting visual consistency. Additionally, their motion modules lack guidance from text, significantly weakening the model's ability to control the results in text-guided inpainting. Our **CoCoCo** aims to achieve superior text-guided video inpainting by enhancing both the training data and model structure, and by designing a transformation mechanism to improve temporal consistency, enhance text controllability, and foster better compatibility with image models.

Methodology

Preliminary

AnimateDiff. We adopt the design of AnimateDiff, which freezes the pretrained text-to-image model and inserts trainable motion module to capture temporal information for

video generation. Specifically, AnimateDiff uses the pre-trained Stable Diffusion (SD) as its base model to modelling the spatial information and inserts trainable motion blocks into frozen base model to modelling the temporal information. The spatial module and the motion module alternately appear in the U-Net.

Stable Diffusion Inpainting. We chose the Stable Diffusion inpainting (SD-IP) model as our base model instead of the original Stable Diffusion. SD-IP is a fine-tuned version of SD, with the key change is increasing input channels to accommodate mask information for inpainting. While the original model has 4 input channels, SD-IP-1.5 uses 9: 4 for the latent z_t, 4 for the masked latent $z_{0,m}$, and 1 for the resized mask $\bar{m}$, where t is the time step. So, the input latent of the U-Net in SD-IP is:

$$z = \text{concat}(z_t, \bar{z}_0, \bar{m}) \tag{1}$$

Task Vectors. To perform personalized video inpainting, we first need personalized inpainting LoRAs or Dreambooths. However, most are trained for image generation, with few made for inpainting. To address this, we use the *task vector* technique to transform personalized image generation models into inpainting models without additional tuning. A task vector is created by subtracting the weights of a pre-trained model from its fine-tuned version. Adding multiple task vector on a pretrained model allows the model to handle multiple tasks simultaneously.

Overview. As we mentioned in the Introduction, the current text-guided video inpainting works still have significant drawbacks, including motion occlusion, inconsistency between inpainting region and surroundings and poor text-alignment. To this end, we design a new training data paradigm and a more powerful motion block to improve text-alignment and inpainting consistency. Besides, we devised an transfer mechanism, enabling our **CoCoCo** create personalized visual content in the given region by using text-image generation LoRA without specific tuning. Therefore, in the following sections, we will introduce these three components separately, with the complete framework diagram shown in Figure 2.

New Training Data Paradigm

Existing text-guided video inpainting methods, like AVID (Zhang et al. 2023b), rely on random mask selection, which might not cover any object, causing misalignment between the masked region and the caption. To ensure consistency between the mask and caption, we introduce an instance-aware region selection strategy during training. This strategy involves three stages: detecting object names in the first frame, detecting objects throughout the video, and generating an instance-aware mask.

Specifically, our approach starts by using GroundingDINO (Liu et al. 2023) and video captions to detect objects and their names in the first frame. These names are then used to identify corresponding bounding boxes across all frames. We generate a mask sequence that randomly shapes but covers these bounding boxes, along with textual descriptions crafted from the object names. During training, we mix precise masks from our instance-aware region selection with random masks to avoid overfitting. We also occasionally drop text conditioning, hinting at classifier-free guidance. The workflow is illustrated in Figure 3.

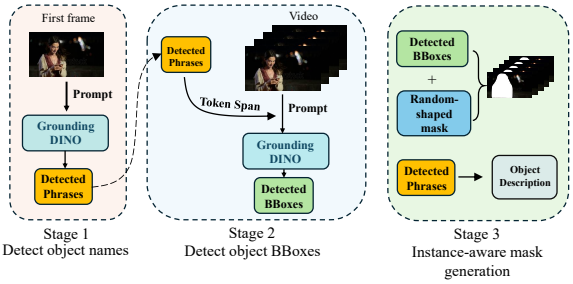


Figure 3: The framework of instance-aware region selection.

Better Motion Module Design

Damped Global Attention. The video inpainting task requires more than just video generation. Video inpainting not only demands consistency between frames but also the inpainted region and its surroundings. Methods like VideoComposer and AVID struggle with this due to limited interaction between spatial and temporal attention, leading to poor global information capture and less adaptive inability to adapt the inpainting region based on the surrounding areas.

Recent video generation methods, such as Sora(Brooks et al. 2024) and CogVideoX(Yang et al. 2024), have demonstrated that 3D full attention has superior modeling ability for motion capture, achieving impressive results for video generation and significantly outperforming traditional approaches that separate spatial and temporal attention. This motivates us to use 3D full attention for video inpainting task to perform the attention in the motion block, which can help maintain consistency. However, employing full 3D attention significantly increases GPU memory consumption, making it too costly for training and inference. To address this challenge, we have developed an efficient 3D full attention mechanism that can improve consistency while reducing memory consumption.

Given the input tensor $x \in \mathbb{R}^{f \times w_1 \times h_1}$ for each attention layer in the U-Net, we first resize the spatial dimensions from $w_1 \times h_1$ to $w_1' \times h_1'$. The resized tensor is then flattened into a one-dimensional vector x with length $f \cdot w_1' \cdot h_1'$, combining temporal and spatial dimensions. This vector is passed through a multi-head self-attention layer, where standard attention computation is performed. Finally, the attention output is reshaped and resized back to its original dimensions $f \times w_1 \times h_1$. This process can be formulated as:

$$\begin{aligned} x \in \mathbb{R}^{b \cdot f \cdot c \cdot w_1 \cdot h_1} &\xrightarrow[\text{\& reshape}]{\text{resize}} \mathbb{R}^{b \cdot (w_1' \cdot h_1' \cdot f) \cdot c} \\ &\xrightarrow{\text{SA}} \mathbb{R}^{b \cdot (w_1' \cdot h_1' \cdot f) \cdot c} \xrightarrow[\text{\& reshape}]{\text{resize}} \mathbb{R}^{b \cdot f \cdot c \cdot w_1 \cdot h_1} \end{aligned} \tag{2}$$

Consider $w_1 = 4w_1'$ and $h_1 = 4h_1'$, resizing the input tensor z reduces sequence length by 16 times, and therefore de-

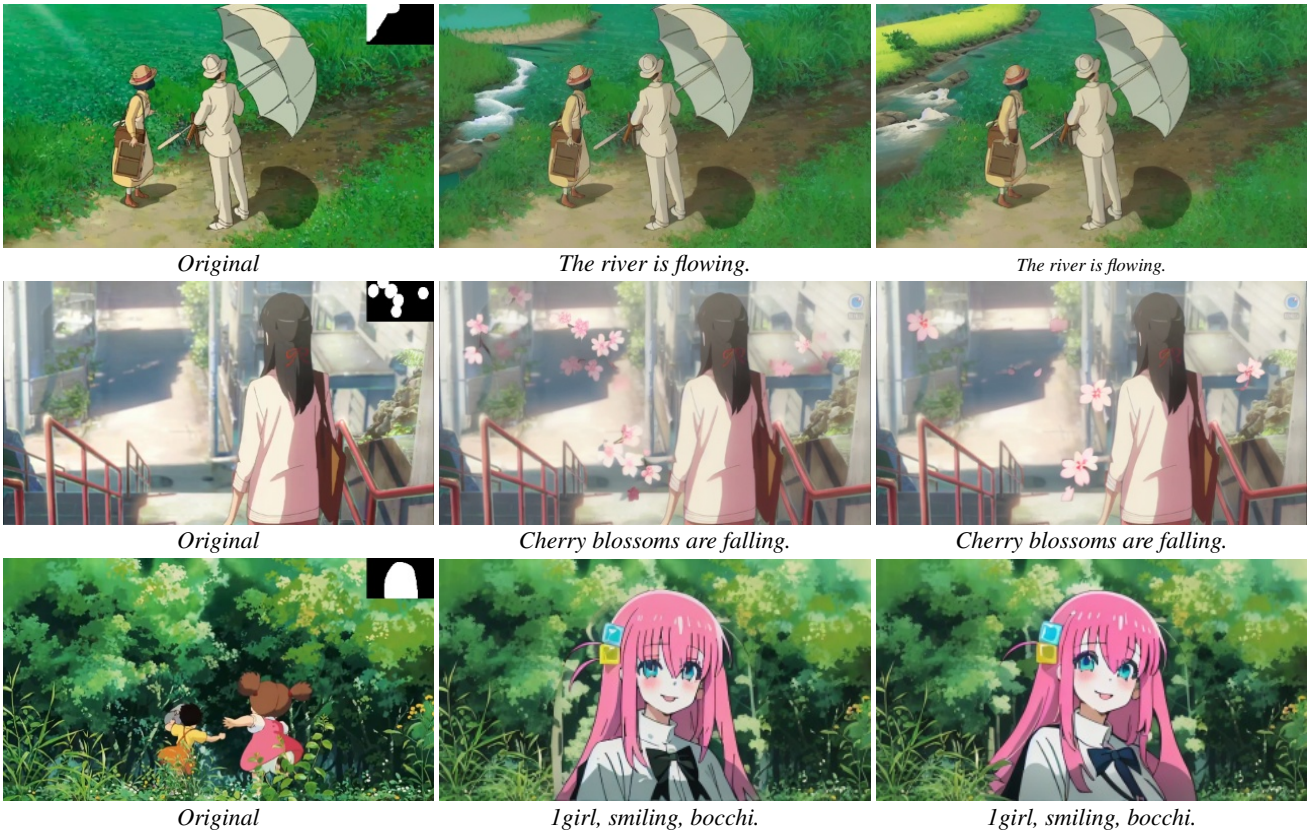


Figure 4: The personalized video inpainting results of our **CoCoCo** with personalized generation model CounterfeitV3.0 and Bocchi. *To view videos, please check our extended version.*

creasing memory consumption by 256 times in Self Attention. To prevent the potential loss of high-frequency information due to the interpolation involved in the resizing operation, we use residual connections to connect the damped global attention with other attention mechanisms. This allows our model to effectively grasp global information while also maintaining the necessary local details.

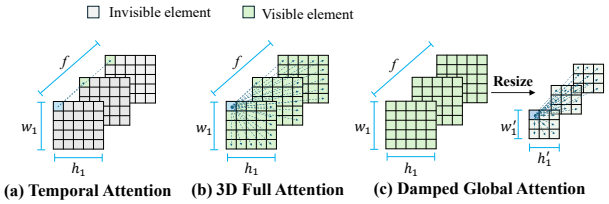


Figure 5: The comparison between temporal attention, 3D full attention and damped global attention.

Other Details. AnimateDiff and AVID utilized text embeddings solely in the spatial layers, which presented a notable limitation: the inability to incorporate motion details from the text prompts. To overcome this, we have incorporated a textual cross-attention mechanism into our motion capture module, thereby enhancing the representation of motion information. Furthermore, for the weight initialization of our motion capture module, we have adopted a two-pronged approach. Specifically, we initialize the temporal attentions with AnimateDiffV2, while the remaining Damped Global Attention and textural cross-attention components are initialized using the Kaiming method (He et al. 2015).

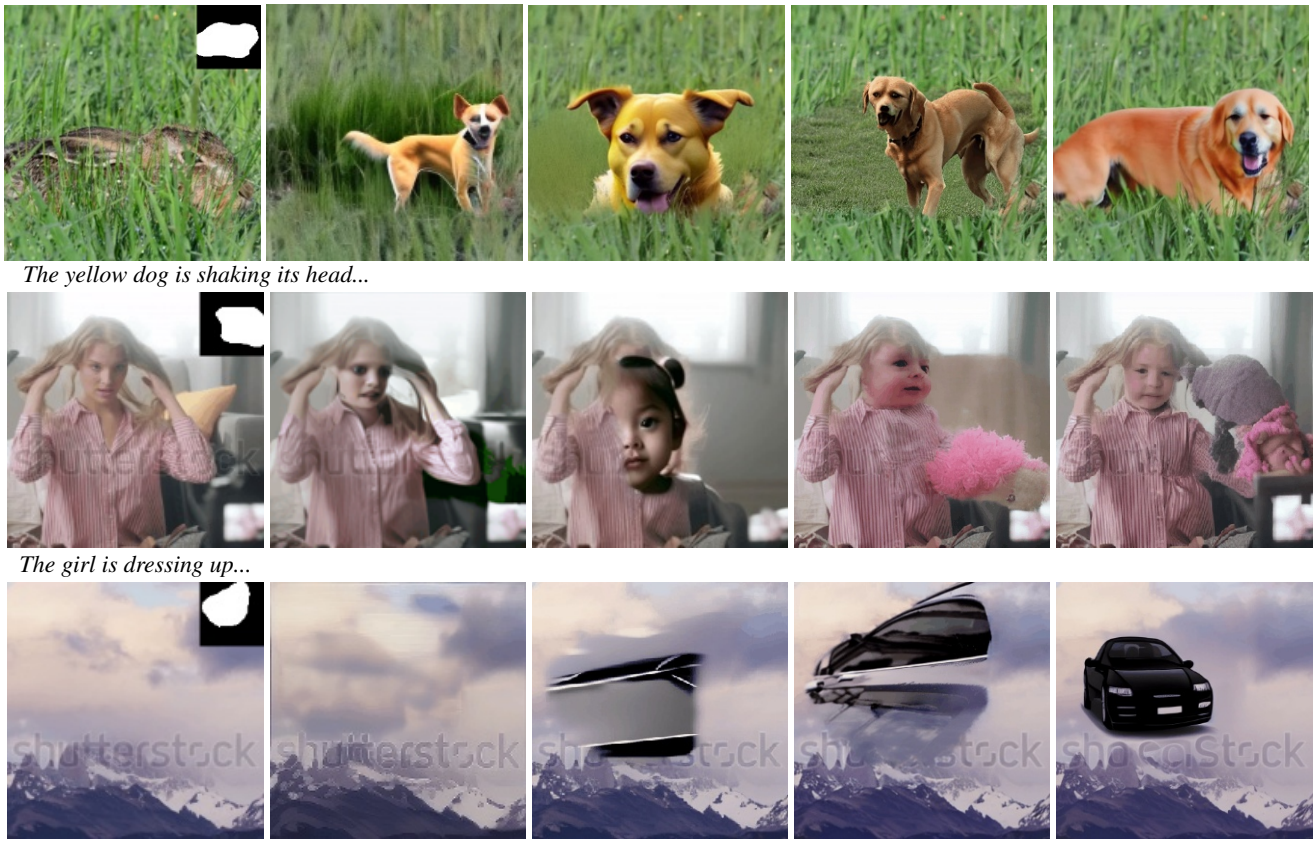
Personalized Video Inpainting

After training **CoCoCo** on SD-IP, we aim to combine it with personalized models to allow users to create personalized visual content. Following AnimateDiff, we train only the temporal layers, keeping spatial layers frozen, enabling us to use LoRAs or Dreambooths trained on SD. However, most personalized LoRAs are trained for image generation, only a small proportion is made for inpainting. This inspires us to explore a strategy to adapt SD LoRAs for SD-IP without additional fine-tuning and further apply transformed personalized inpainting LoRAs on our **CoCoCo**.

Inspired by the concept of task vectors (Ilharco et al. 2022), which we mentioned in Preliminary, here is the personalized task vector:

$$\tau_{\mathrm{p}} = \theta_{\mathrm{p}} - \theta_{\mathrm{base}} \tag{3}$$

where θ_{p} represents the personalized diffusion model (SD with LoRAs or SD Dreambooths) and θ_{base} represents the



(a) Original (b) VideoComposer (c) VideoCrafter2* (d) AnimateDiff V3* (e) Ours

Figure 6: The visual comparison results of our **CoCoCo** . * indicates zero-shot inpainting. *To view videos, please check our extended version.*

base SD. Similarly, we define the task vector has ability to perform general inpainting as:

$$\tau_{\text{ip}} = \theta_{\text{ip}} - \theta_{\text{base}} \tag{4}$$

where θ_{ip} represents the SD-IP. Importantly, the SD-IP has a different input channel size compared to the SD. To address this mismatch, we conduct zero padding to increase the channel size of SD from 4 to 9, thus overcoming the channel discrepancy. The personalized inpainting image model is then derived from:

$$\theta = \theta_{\text{base}} + \alpha\tau_{\text{ip}} + \beta\tau_{\text{p}} \tag{5}$$

Here, α and β are hyperparameters. We recommend setting α within the range of [0.5, 1.5] and β within the range of [1, 2]. To create a personalized video inpainting model, we simply integrate the transformed personalized image inpainting model with our **CoCoCo** based on the base SD-IP, which means we replace the base model in **CoCoCo** from SD-IP to the personalized SD-IP, without requiring any further fine-tuning.

Training Objectives

Given a video clip $v^{1:f} \in \mathbb{R}^{f\times c\times w\times h}$ and its masked counterpart $\bar{v}^{1:f} = v^{1:f} \odot m^{1:f}$, we encode both into latent codes $z_0^{1:f}$ and $\bar{z}_0^{1:f}$ frame by frame using a VAE encoder, where $z_0^{1:f}, \bar{z}_0^{1:f} \in \mathbb{R}^{f\times c\times w_l\times h_l}$. Simultaneously, the mask m is resized to $\bar{m}$. Here, f is the number of frames, c is the channel count, w and h are the video's width and height, and w_l and h_l represent the dimensions of the latent codes. During the forward diffusion process, noise is added to the latent codes $z_0^{1:f}$, producing $z_t^{1:f}$ with time step t through standard diffusion. The U-Net model then takes $z_t^{1:f}$, the binary mask $\bar{m}$, the masked latents $\bar{z}_0^{1:f}$, and the prompt y encoded by text encoder c_θ as input and predicts the added noise ϵ. The final training objective of our model is:

$$\begin{aligned} \mathcal{L} &= \mathbb{E}_{z^{1:f},\bar{z}^{1:f},\bar{m},y,\epsilon^{1:f}\sim\mathcal{N}(0,I),t}[||\epsilon^{1:f} - \epsilon_p^{1:f}||_2^2] \\ \epsilon_p^{1:f} &= \epsilon_\theta(z_t^{1:f}, \bar{m}, \bar{z}_0^{1:f}, t, c_\theta(y)) \end{aligned} \tag{6}$$

Experiments

Implementation Details

Data. We chose WebVid-10M as our training set. For data cleaning, we use the *Scenedetect* library with a threshold of 20 to guarantee videos with a single scene and discard those with multiple scenes. For instance-aware region selection,

we set the detection resolution to 396×512 and a bounding box and phrase selection threshold of 0.2. Training clips are sampled from three data types as described in Section with probabilities of 0.7, 0.2, and 0.1, respectively.

Training Details. We use the AdamW (Loshchilov and Hutter 2017) optimizer with a learning rate of 1×10^{-4} and a constant scheduler. The model is trained for 1 epoch with a batch size of 256 using gradient accumulation. Following DDPM (Ho, Jain, and Abbeel 2020), we use 1000 steps. The Stable Diffusion Inpainting V1.5 model initializes the spatial block, which remains unchanged while we train the motion block. Temporal attention layers are initialized with AnimateDiff V2, and the damped global attention and cross-attention layers with Kaiming initialization. Training is done at a resolution of 256×384, with a sample stride of 4 and 16 frames.

Inference Details. In the inference stage, we follow DDIM (Song, Meng, and Ermon 2021), use a 50 sampling steps and the classifier-free guidance scale is 14. The mask for per frame can be obtained by Grounding DINO (Liu et al. 2023) and SAM (Kirillov et al. 2023) automatically or provided by the users with any shape. For the video with resolution of 512×512 and 32 frames, the inpainting process can be finished within 1 minute on a Nvidia 4090 GPU.

A car driving on the road.

A can floating in the water.

(a) Original (b) AVID (c) Ours

Figure 7: Comparison with AVID. Red rectangles shows the inconsistency and poor text-alignment. *To view videos, please check our extended version.*

Experimental Results

We conduct extensive experiments to evaluate our method. In our experiments, we random select 1000 videos from the validation set of the WebVid-10M (Bain et al. 2021), and extract the first 16 frames in each video with the sample rate of 4. We randomly generate the mask and prompt, and ask model to generate the visual content in the masked region. For baselines, we choose AnimateDiff V3 (Guo et al. 2023) and VideoCrater2 (Chen et al. 2024) and use the zero-shot inpainting method to fill in the masked region. Besides, we compare our method with the text-guided inpainting module in VideoComposer (Wang et al. 2023c). Since AVID (Zhang et al. 2023b) is not open-source when we submit this manuscript, we can only compare our method with its demo videos.

Quantitative Comparison. We use the CLIP score (CS) to evaluate text alignment across methods, L1 distance to measure background preservation (BP), where lower values indicate better preservation, and cosine similarity in feature space (extracted by CLIP) to assess motion smoothness between consecutive frames.

As shown in Table 1, our model outperforms the other methods in BP and TC, with a BP value of 6.20 on a [0, 255] scale, significantly lower than VideoComposer, and better than AnimateDiff V3 and VideoCrafter2. Our method also performs best in temporal consistency, producing more plausible inpainted videos.

Regarding the CLIP score, AnimateDiff V3 and VideoCrafter2, both text-to-video models, perform better, indicating strong text-to-visual content alignment. However, these methods doesn't have a decent ability to keep consistency between inpainted region and its surroundings. Therefore, it's natural for VideoComposer scores much lower, since it focus on consistency between generated and external content, which weakens textual alignment. Our method, with a CLIP score of 24.9, significantly surpasses VideoComposer and closely approaches AnimateDiff V3, demonstrating the effectiveness of our instance-aware region selection strategy and textual cross-attention in the motion capture block.

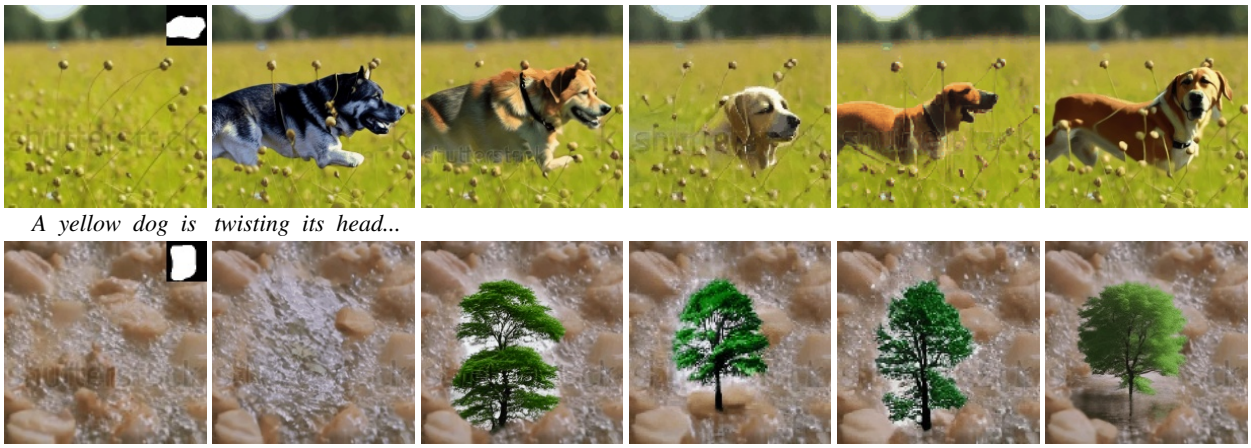
Qualitative Results. We conducted a blind evaluation comparing our method with baselines across four aspects: visual quality (VC), text alignment (TA), temporal consistency (TC), and background preservation (BP). As shown in Table 1, our model consistently ranks highest, especially in temporal consistency and background preservation. Figure 6 illustrates that while VideoComposer maintains some consistency between masked and unmasked regions, it alters the background significantly compared to the original videos. In contrast, our method, seen in the fifth column, delivers better motion consistency and higher visual quality than the other three methods. Additional qualitative results are available in Figures 1, 4 and 7.

Comparison with AVID. Since AVID remains closed-source, we compare our inpainting results with the samples provided by AVID. As shown in Figure 7, our method demonstrates improved inpainting consistency and text controllability. Red rectangles highlight inconsistencies or poor text alignment in AVID's video samples.

Ablation Study. We conducted ablation study to evaluate the effectiveness of each component, using four different settings. In the first, we applied random mask selection and stacked two temporal layers in the motion block. The second setting added instance-aware region selection. The third aimed to assess whether additional temporal attention layers would improve performance. The final setting tested if the performance gain was independent of textual cross-attention. As shown in Figure 8, the model under the first setting was not controllable by the given prompts. In con-

Method	Quantative Results			User Study			
	CS (↑)	BP (↓)	TC (↑)	VQ (↑)	TA (↑)	TC (↑)	BP (↑)
AnimateDiff V3* (Guo et al. 2023)	25.3	7.2	96.7	14.6	20.8	12.5	25.0
VideoCrafter2* (Chen et al. 2024)	**26.2**	7.8	96.8	2.1	8.3	2.1	6.2
VideoComposer (Wang et al. 2023c)	19.8	21.7	96.5	12.5	16.7	29.2	4.2
CoCoCo (ours)	24.9	**6.2**	**97.2**	**72.9**	**54.2**	**56.2**	**64.6**

Table 1: The comparison between our CoCoCo and three other methods. "CS", "BP", "TC", "VQ" and "TA" denotes CLIP score, background preservation, temporal consistency, visual quality, text alignment, respectively.



(a) Original (b) Ablation-1 (c) Ablation-2 (d) Ablation-3 (e) Ablation-4 (f) Ours

Figure 8: The ablation study results of our **CoCoCo**. *To view videos, please check our extended version.*

Method	Instance-aware Region Selection	Motion Capture Block	CS (↑)	BP (↓)	TC (↑)
Ablation-1	×	2×Temp Attn	21.6	8.3	96.3
Ablation-2	✓	2×Temp Attn	23.6	8.6	96.8
Ablation-3	✓	4×Temp Attn	23.9	8.1	96.9
Ablation-4	✓	3×Temp Attn + Cross-Attn	24.2	7.9	96.7
CoCoCo (ours)	✓	2×Temp Attn + DGA + Cross-Attn	**24.9**	**6.2**	**97.2**

Table 2: The ablation study results of our model. "CS", "BP", "TC" denotes CLIP score, background preservation, temporal consistency, respectively.

trast, the second setting with instance-aware region selection showed better text-alignment. Settings three and four do not show significant improvement over setting two, indicating that more temporal attention layers do not enhance performance, while textual cross-attention indeed improves text-alignment. We also conduct some quantitative experiments to compare the above four settings with our **CoCoCo**. The results can be found in Table 2. We can see from Table 2 that using the instance-aware region selection can significantly increase CLIP score from 21.6 to 23.6, and adding damped global attention can largely improve background preservation from 7.9 to 6.2. Using both skills can obviously enhance the temporal consistency from 96.3 to 97.2.

Conclusion

In this paper, we presented **CoCoCo**, a novel text-guided video inpainting model via improving its motion consistency, textual controllability and compatibility. To achieve improved motion consistency, we developed a damped global attention mechanism. Additionally, to enhance textual controllability, we designed an instance-aware region selection strategy and inserted textual cross-attention layer into the motion capture block. For personalized video inpainting, we introduced a strategy that adapts personalized image generation models into inpainting models without training. This approach, combined with our CoCoCo model, enables effective personalized video inpainting. Quantitative results and user study experiments showed our method achieved better results compared to its counterparts. More importantly, qualitative results also demonstrated that the proposed model achieved better motion consistency, textual controllability and model compatibility.

References

Andonian, A.; Osmany, S.; Cui, A.; Park, Y.; Jahanian, A.; Torralba, A.; and Bau, D. 2021. Paint by word. arXiv:2103.10951.

Avrahami, O.; Fried, O.; and Lischinski, D. 2022. Blended latent diffusion. arXiv:2206.02779.

Avrahami, O.; Lischinski, D.; and Fried, O. 2022. Blended diffusion for text-driven editing of natural images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Bain, M.; Nagrani, A.; Varol, G.; and Zisserman, A. 2021. 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*.

Bao, F.; Xiang, C.; Yue, G.; He, G.; Zhu, H.; Zheng, K.; Zhao, M.; Liu, S.; Wang, Y.; and Zhu, J. 2024. Vidu: a Highly Consistent, Dynamic and Skilled Text-to-Video Generator with Diffusion Models. arXiv:2405.04233.

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

Blattmann, A.; Rombach, R.; Ling, H.; Dockhorn, T.; Kim, S. W.; Fidler, S.; and Kreis, K. 2023b. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Brooks, T.; Peebles, B.; Holmes, C.; DePue, W.; Guo, Y.; Jing, L.; Schnurr, D.; Taylor, J.; Luhman, T.; Luhman, E.; Ng, C.; Wang, R.; and Ramesh, A. 2024. Video generation models as world simulators. https://openai.com/research/video-generation-models-as-world-simulators. Accessed: 2024-07-10.

Chen, H.; Zhang, Y.; Cun, X.; Xia, M.; Wang, X.; Weng, C.; and Shan, Y. 2024. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. arXiv:2401.09047.

Chen, W.; Ji, Y.; Wu, J.; Wu, H.; Xie, P.; Li, J.; Xia, X.; Xiao, X.; and Lin, L. 2023. Control-A-Video: Controllable Text-to-Video Generation with Diffusion Models. arXiv:2305.13840.

Couairon, G.; Verbeek, J.; Schwenk, H.; and Cord, M. 2022. Diffedit: Diffusion-based semantic image editing with mask guidance. arXiv:2210.11427.

Ding, M.; Zheng, W.; Hong, W.; and Tang, J. 2022. Cogview2: Faster and better text-to-image generation via hierarchical transformers. *Advances in Neural Information Processing Systems*.

Esser, P.; Chiu, J.; Atighehchian, P.; Granskog, J.; and Germanidis, A. 2023. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Fan, F.; Guo, C.; Gong, L.; Wang, B.; Ge, T.; Jiang, Y.; Luo, C.; and Zhan, J. 2023. Hierarchical masked 3d diffusion model for video outpainting. In *Proceedings of the 31st ACM International Conference on Multimedia*.

Gen-3. 2024. Introducing Gen-3 Alpha: A New Frontier for Video Generation. https://runwayml.com/research/introducing-gen-3-alpha/. Accessed: 2024-07-10.

Guo, Y.; Yang, C.; Rao, A.; Wang, Y.; Qiao, Y.; Lin, D.; and Dai, B. 2023. AnimateDiff: Animate Your Personalized Text-to-Image Diffusion Models without Specific Tuning.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*.

Ho, J.; Chan, W.; Saharia, C.; Whang, J.; Gao, R.; Gritsenko, A.; Kingma, D. P.; Poole, B.; Norouzi, M.; Fleet, D. J.; et al. 2022a. Imagen video: High definition video generation with diffusion models. arXiv:2210.02303.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*.

Ho, J.; Salimans, T.; Gritsenko, A.; Chan, W.; Norouzi, M.; and Fleet, D. J. 2022b. Video Diffusion Models. arXiv:2204.03458.

Hu, E. J.; Shen, Y.; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2021. Lora: Low-rank adaptation of large language models. arXiv:2106.09685.

Ilharco, G.; Ribeiro, M. T.; Wortsman, M.; Gururangan, S.; Schmidt, L.; Hajishirzi, H.; and Farhadi, A. 2022. Editing models with task arithmetic. arXiv:2212.04089.

Jiang, Y.; Wu, T.; Yang, S.; Si, C.; Lin, D.; Qiao, Y.; Loy, C. C.; and Liu, Z. 2023. VideoBooth: Diffusion-based Video Generation with Image Prompts. arXiv:2312.00777.

Ju, X.; Liu, X.; Wang, X.; Bian, Y.; Shan, Y.; and Xu, Q. 2024. BrushNet: A Plug-and-Play Image Inpainting Model with Decomposed Dual-Branch Diffusion. arXiv:2403.06976.

Khachatryan, L.; Movsisyan, A.; Tadevosyan, V.; Henschel, R.; Wang, Z.; Navasardyan, S.; and Shi, H. 2023. Text2video-zero: Text-to-image diffusion models are zero-shot video generators. arXiv:2303.13439.

Kim, D.; Woo, S.; Lee, J.-Y.; and Kweon, I. S. 2019. Deep video inpainting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Kirillov, A.; Mintun, E.; Ravi, N.; Mao, H.; Rolland, C.; Gustafson, L.; Xiao, T.; Whitehead, S.; Berg, A. C.; Lo, W.-Y.; et al. 2023. Segment anything. arXiv:2304.02643.

Kling. 2024. KLING VIDEO MODEL. https://kling.kuaishou.com/en. Accessed: 2024-07-10.

Kondratyuk, D.; Yu, L.; Gu, X.; Lezama, J.; Huang, J.; Hornung, R.; Adam, H.; Akbari, H.; Alon, Y.; Birodkar, V.; et al. 2023. Videopoet: A large language model for zero-shot video generation. arXiv:2312.14125.

Lee, S.; Oh, S. W.; Won, D.; and Kim, S. J. 2019. Copy-and-paste networks for deep video inpainting. In *Proceedings of the IEEE/CVF international conference on computer vision*.

Li, Z.; Lu, C.-Z.; Qin, J.; Guo, C.-L.; and Cheng, M.-M. 2022. Towards an end-to-end framework for flow-guided video inpainting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*.

Liu, R.; Deng, H.; Huang, Y.; Shi, X.; Lu, L.; Sun, W.; Wang, X.; Dai, J.; and Li, H. 2021. Fuseformer: Fusing fine-grained information in transformers for video inpainting. In *Proceedings of the IEEE/CVF international conference on computer vision*.

Liu, S.; Zeng, Z.; Ren, T.; Li, F.; Zhang, H.; Yang, J.; Li, C.; Yang, J.; Su, H.; Zhu, J.; et al. 2023. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. arXiv:2303.05499.

Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. arXiv:1711.05101.

Luma. 2024. The Ultimate AI Video Generator. https://lumaai.video. Accessed: 2024-07-10.

Ma, Y.; He, Y.; Cun, X.; Wang, X.; Shan, Y.; Li, X.; and Chen, Q. 2023. Follow Your Pose: Pose-Guided Text-to-Video Generation using Pose-Free Videos. arXiv:2304.01186.

Ma, Z.; Zhou, D.; Yeh, C.-H.; Wang, X.-S.; Li, X.; Yang, H.; Dong, Z.; Keutzer, K.; and Feng, J. 2024. Magic-Me: Identity-Specific Video Customized Diffusion. arXiv:2402.09368.

Ouyang, H.; Wang, T.; and Chen, Q. 2021. Internal video inpainting by implicit long-range propagation. In *Proceedings of the IEEE/CVF international conference on computer vision*.

Peebles, W.; and Xie, S. 2023. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Pika Labs. 2023. Pika Labs. https://www.pika.art/. Accessed: 2024-07-10.

Ruiz, N.; Li, Y.; Jampani, V.; Pritch, Y.; Rubinstein, M.; and Aberman, K. 2023. 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*.

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

Song, J.; Meng, C.; and Ermon, S. 2021. Denoising Diffusion Implicit Models. In *9th International Conference on Learning Representations*.

Wang, J.; Yuan, H.; Chen, D.; Zhang, Y.; Wang, X.; and Zhang, S. 2023a. ModelScope Text-to-Video Technical Report. arXiv:2308.06571.

Wang, S.; Saharia, C.; Montgomery, C.; Pont-Tuset, J.; Noy, S.; Pellegrini, S.; Onoe, Y.; Laszlo, S.; Fleet, D. J.; Soricut, R.; et al. 2023b. Imagen editor and editbench: Advancing and evaluating text-guided image inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Wang, X.; Yuan, H.; Zhang, S.; Chen, D.; Wang, J.; Zhang, Y.; Shen, Y.; Zhao, D.; and Zhou, J. 2023c. VideoComposer: Compositional Video Synthesis with Motion Controllability.

Wu, J. Z.; Ge, Y.; Wang, X.; Lei, S. W.; Gu, Y.; Shi, Y.; Hsu, W.; Shan, Y.; Qie, X.; and Shou, M. Z. 2023. 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*.

Xie, S.; Zhao, Y.; Xiao, Z.; Chan, K. C. K.; Li, Y.; Xu, Y.; Zhang, K.; and Hou, T. 2023. DreamInpainter: Text-Guided Subject-Driven Image Inpainting with Diffusion Models. arXiv:2312.03771.

Xing, J.; Xia, M.; Zhang, Y.; Chen, H.; Wang, X.; Wong, T.-T.; and Shan, Y. 2023. Dynamicrafter: Animating open-domain images with video diffusion priors. arXiv:2310.12190.

Xu, R.; Li, X.; Zhou, B.; and Loy, C. C. 2019. Deep flow-guided video inpainting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Yang, Z.; Teng, J.; Zheng, W.; Ding, M.; Huang, S.; Xu, J.; Yang, Y.; Zhang, X.; Gu, X.; Feng, G.; Yin, D.; Hong, W.; Wang, W.; Cheng, Y.; Zhang, Y.; Liu, T.; Xu, B.; Dong, Y.; and Tang, J. 2024. CogVideoX: Text-to-Video Diffusion Models with An Expert Transformer.

Yin, S.; Wu, C.; Liang, J.; Shi, J.; Li, H.; Ming, G.; and Duan, N. 2023. DragNUWA: Fine-grained Control in Video Generation by Integrating Text, Image, and Trajectory. arXiv:2308.08089.

Zeng, Y.; Fu, J.; and Chao, H. 2020. Learning joint spatial-temporal transformations for video inpainting. In *European conference on computer vision*.

Zhang, D. J.; Wu, J. Z.; Liu, J.-W.; Zhao, R.; Ran, L.; Gu, Y.; Gao, D.; and Shou, M. Z. 2023a. Show-1: Marrying Pixel and Latent Diffusion Models for Text-to-Video Generation. arXiv:2309.15818.

Zhang, Z.; Wu, B.; Wang, X.; Luo, Y.; Zhang, L.; Zhao, Y.; Vajda, P.; Metaxas, D.; and Yu, L. 2023b. AVID: Any-Length Video Inpainting with Diffusion Model.

Zhou, S.; Li, C.; Chan, K. C.; and Loy, C. C. 2023. Propainter: Improving propagation and transformer for video inpainting. In *Proceedings of the IEEE/CVF international conference on computer vision*.