GenVideo: One-shot target-image and shape aware video editing using T2I diffusion models

Supplementary Material

6. Architecture details

This section provides an overview of the underlying model and explains how the features are passed through the pipeline during training and inference.

Base model: We use the SD-unCLIP model, a fine-tuned version of the Stable Diffusion v2.1 text-to-image model that accepts CLIP image embedding and the text prompt as conditional input. The network broadly consists of the VAE autoencoder $\{\mathcal{E}, \mathcal{D}\}$, the latent denoising UNet $\varepsilon_\theta(\cdot)$, and CLIP conditional models (image branch and text branch) which extract an image embedding $\mathcal{J}$ and a text embedding $\mathcal{C}$. As shown in Fig. 10, the UNet network consists of down-block, mid-block, and up-block. Each of these blocks has 4, 1, and 4 subblocks respectively. Each of these subblocks typically constitutes two ResNet blocks and two inflated attention network blocks arranged as shown in Fig. 10. The only trainable components of the network belong to the inflated attention modules explained in the subsequent subsections.

Feature resolutions: The VAE encoder $\mathcal{E}$ reduces the spatial dimensions from 768 to 96. The down-blocks further reduce the spatial dimensions to 24 while increasing the channel dimensions from 4 to 1280. The mid-blocks maintain the spatial dimensions and channel dimensions. The up-blocks increase the spatial dimensions to 96, while reducing the channel dimensions to 4. The VAE decoder then increases the spatial resolution back to 768. The CLIP text embedding and image embedding is a vector of size 768.

UNet forward pass: The inputs to the UNet network $\varepsilon_\theta(\cdot)$ are the latent noise from previous timestep z_t, the sinusoidal timestep embedding t_{emb}, an optional mask $\mathcal{M}$, the CLIP image embedding $\mathcal{J}$ and the CLIP text embedding $\mathcal{C}$. The latent noise z_t is forwarded into layers of UNet network. At the end of a ResNet block, the hidden states are updated with the timestep embedding and the optional image embedding information based on the input mask –

- **When an input mask is not provided, *i.e.,* $\mathcal{M} = \mathcal{M}_\phi$:** In this situation, the hidden states are updated by adding the timestep embedding t_{emb} and the image embedding $\mathcal{J}$ to all spatial locations of the hidden states.
- **When an input mask is provided, *i.e.,* $\mathcal{M} \neq \mathcal{M}_\phi$:** In this situation, the regions that correspond to the background (*i.e.,* where $\mathcal{M} = 0$) are updated by adding the t_{emb} and source image embedding $\mathcal{J}^{src}$. Similarly, for the regions corresponding to the foreground (*i.e.,* where $\mathcal{M} = 1$), the hidden states are updated by adding the timestep embedding t_{emb} and the target image embedding $\mathcal{J}^{trg}$ as shown in Fig. 10.

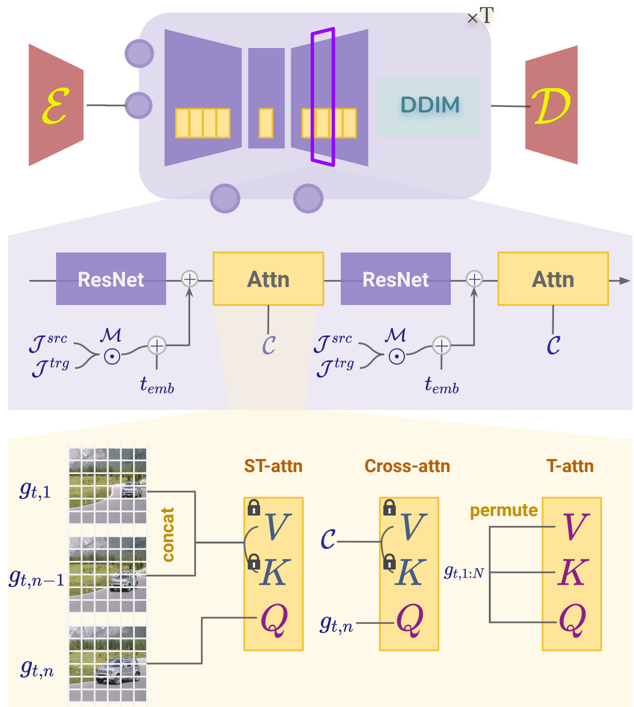

Figure 10. Architectural diagram. *Top to bottom*: UNet architecture with VAE, UNet block architecture, Attention layer inflation of ST-attn, Cross-attn, and T-attn.

Once the hidden states are updated, they are passed into inflated attention blocks and the subsequent network layers. At the end of each denoising step of UNet, a latent fusion step is performed when an input mask is provided. In the next subsections, we explain the latent fusion method and the inflated model architecture.

Latent fusion: Our latent fusion method follows from Make-A-Protagonist [46]. The latent fusion step helps improve the quality of the rendered object in the edited video. First, UNet features are obtained using only the target image embedding $\mathcal{J}^{trg}$ with no mask input to UNet to obtain $\varepsilon_\theta(z, t, \mathcal{C}, \mathcal{J}^{trg}, \mathcal{M}_\phi)$. Next, UNet features are obtained using source image embedding $\mathcal{J}^{src}$ and target image embedding $\mathcal{J}^{trg}$ along with a mask $\mathcal{M}$ to obtain $\varepsilon_\theta(z, t, \mathcal{C}, \{\mathcal{J}^{src}, \mathcal{J}^{trg}\}, \mathcal{M})$. Note here $\mathcal{J}^{src}$ is used for $\mathcal{M} = 0$ region and $\mathcal{J}^{trg}$ is used for $\mathcal{M} = 1$ region. These

Figure 11. Correspondence Error (CE) maps computed using ground truth (source video correspondences) before correction across all blocks of UNet. We find that `Up-block-2` has the lowest CE.

Figure 12. Zero-shot image editing results on the brown bear using *InvEdit* mask. Background preservation is not used here.

outputs are combined using the mask in the following manner:

$$z_{t-1} = \frac{1}{1+\mathcal{M}}\big(\mathcal{M} \odot \mathrm{DDIM}(\varepsilon_\theta(z_t, t, \mathcal{C}, \mathcal{J}^{trg}, \mathcal{M}_\phi)) \\ + \mathrm{DDIM}(\varepsilon_\theta(z, t, \mathcal{C}, \{\mathcal{J}^{src}, \mathcal{J}^{trg}\}, \mathcal{M}))\big)$$

Note that when the background is allowed to be changed (like in Fig. 14 and Fig. 12), $\mathcal{J}^{src}$ is replaced with the CLIP image embedding of DALLE-2 prior obtained from the target text $\mathcal{P}^{trg}$. In all the other cases where the background is to be kept the same as the source, it is the CLIP image embedding of the source video frame, *i.e.,* $\mathcal{J}^{src}$. More details can be found in [46].

Inflated attention layers: We follow the inflation strategy laid out by Tune-A-Video [43]. We expand the self-attention layers into spatio-temporal attention (ST-attn) layers by inputting the features from the first frames $g_{t,1}$ along with $g_{t,n-1}$ as shown in Fig. 10 for computation of attention matrix. Here, g denotes features of hidden states in the UNet. Cross-attention layers continue to accept the text tokens from prompt $\mathcal{C}$ along with $g_{t,n}$. We additionally introduce temporal self-attention (T-attn) layers which are trained after permuting the temporal dimensions and spatial dimensions of the mini-batch. The only trainable weights in the entire pipeline are the query weights of ST-attn, query weights of Cross-attn, and all the weights in the T-attn as shown in Fig. 10.

Additional details of training and inference pipeline: During training, the source video is mapped into the VAE encoder's latent space. A random timestep is sampled and noise is added to the latents according to the forward diffusion process. The text embeddings of the source prompt and the image embeddings of a random source video frame are passed (as $\mathcal{C}$ and $\mathcal{J}$ respectively) into the UNet and the mask is $\mathcal{M}_\phi$. The reconstruction loss is imposed at the given timestep as shown in Fig. 2 of the main paper. The gradients are backpropagated using the AdamW optimizer to update the parameters of inflated attention modules described earlier. The inference pipeline consists of two stages - *InvEdit* mask computation and the *latent correction*. While computing the *InvEdit* mask, the mask inputs to UNet are absent, *i.e.,* $\mathcal{M} = \mathcal{M}_\phi$. After computing the *InvEdit* mask $\mathcal{M}^{inv}$ it is passed into the UNet for mask guided inference and *latent correction*, *i.e.*, $\mathcal{M} = \mathcal{M}^{inv}$. See Algorithm 1 for inference pseudo-code.

7. Additional results

Selection of the UNet block for *latent correction* field: We compute the correspondence error (CE) map across all blocks of UNet as per Sec. 3.3 and find that the correspondence errors of `Up-block-2` are generally lower than other blocks as shown in Fig. 11. Across all experiments, we assign the feature with minimal Euclidean distance to the original feature as the corresponding feature. The correspondences obtained by computing RAFT optical flow on the source video serve as the ground truth since the object in the source video and the expected target object have the same shape. For computing the CE, we compare the correspondences obtained in the feature space with the ground truth from RAFT.

Additional results of GenVideo. In Fig. 12, Fig. 13 and

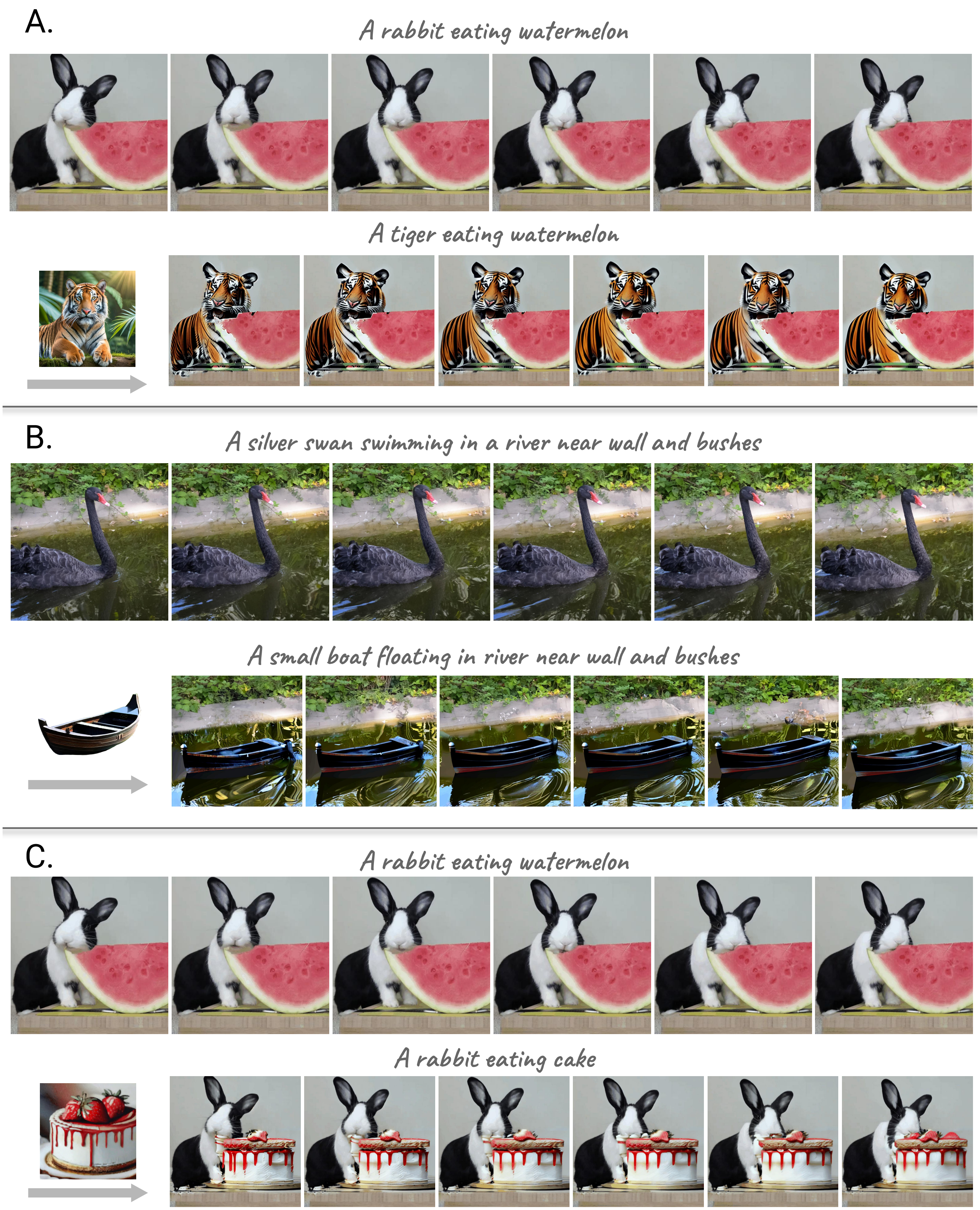

Figure 13. Additional results of *GenVideo*. Our approach can do object edits when target-object has substantially different shape and size.

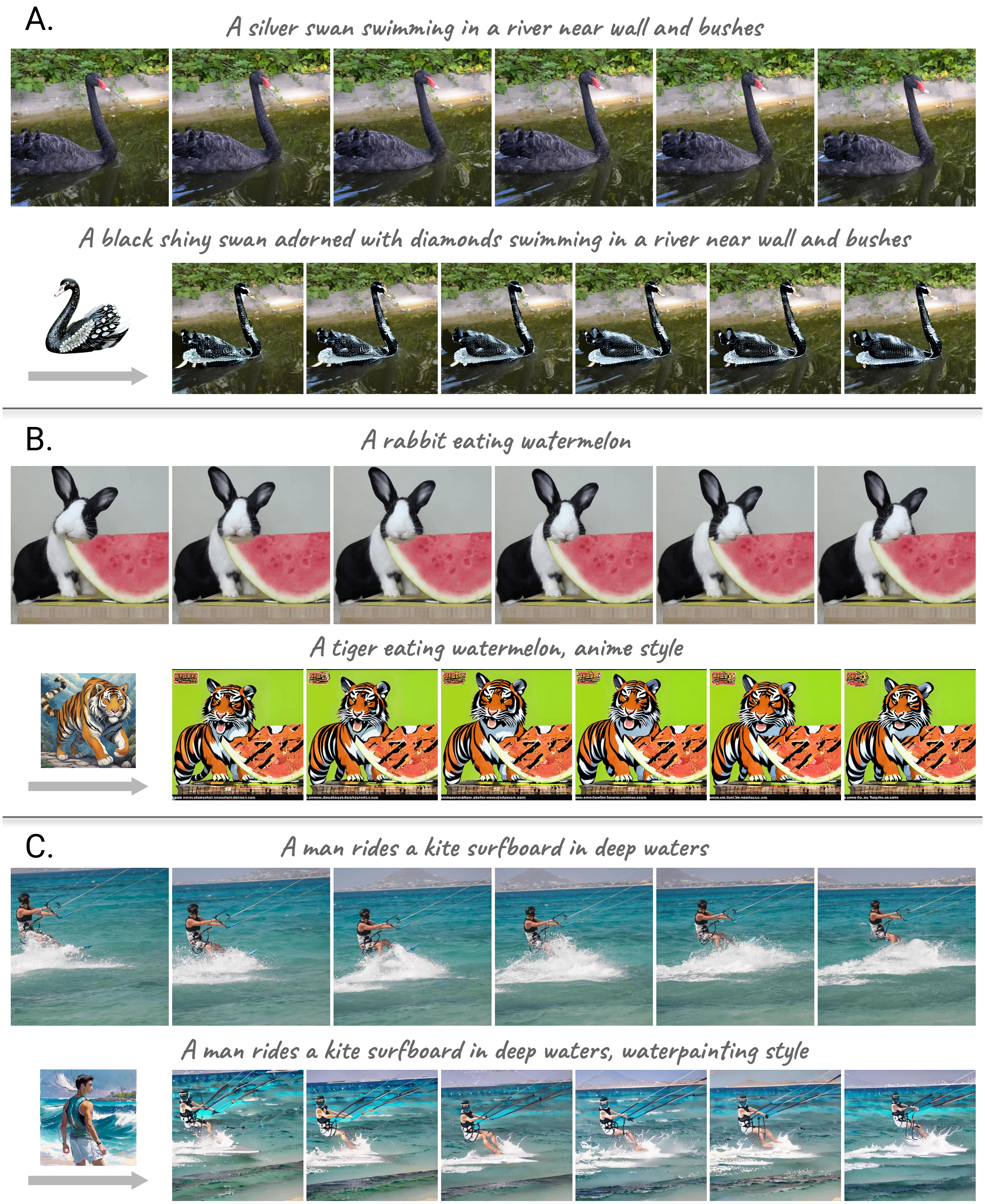

Figure 14. Additional results of *GenVideo* on style editing of videos. Background preservation is not used in **B.** and **C.** since the entire video is being edited.

Algorithm 1 GenVideo Inference

Require: $\mathcal{V}^{src} := [I^{src}_{1:N}], \mathcal{P}^{src}, \mathcal{P}^{trg}, I^{trg}$
Require: $\varepsilon_\theta(\cdot)$(finetuned inflated UNet), $\mathcal{E}, \mathcal{D}, \mathrm{CLIP_t}, \mathrm{CLIP_v}$

Set Hyperparameters:
  $T = 50$ ▷ *DDIM timesteps*
  $\alpha = 0.8$ ▷ *Mask binarization threshold*
  $w_{-1} = 0.1, w_0 = 0.8, w_{-1} = 0.1$ ▷ *Inter-frame blending weights*
$\mathcal{C}^{src}, \mathcal{C}^{trg} = \mathrm{CLIP_t}(\mathcal{P}^{src}), \mathrm{CLIP_t}(\mathcal{P}^{trg})$
$\mathcal{J}^{src}_{1:N}, \mathcal{J}^{trg} = \mathrm{CLIP_v}(I^{src}_{1:N}), \mathrm{CLIP_v}(I^{trg})$
$\mathcal{Z}^{src}_T := [z^{src}_{T,1}, \cdots, z^{src}_{T,N}] = \mathrm{DDIM}^{-1}(\mathcal{E}(\mathcal{V}^{src}))$
$z^{trg}_{T,1:N} = z^{src}_{T,1:N}$
for $t \in [0.8 \times T, T]$ **do** ▷ *Compute the InvEdit mask*
  $t_{emb} = \mathrm{Emb}(t)$ ▷ *sinusoidal timestep embedding*
  $\varepsilon^{src}_{t,1:N} = \varepsilon_\theta\left(z^{src}_{t,1:N}, t_{emb}, \mathcal{C}^{src}, \mathcal{J}^{src}_{1:N}, \mathcal{M}_\phi\right)$
  $\varepsilon^{trg}_{t,1:N} = \varepsilon_\theta\left(z^{trg}_{t,1:N}, t_{emb}, \mathcal{C}^{trg}, \mathcal{J}^{trg}, \mathcal{M}_\phi\right)$
  $\Delta\varepsilon_{t,1:N} = \mathrm{abs}(\varepsilon^{src}_{t,1:N} - \varepsilon^{trg}_{t,1:N})$
  $z^{src}_{t-1,1:N} = \mathrm{DDIM}(\varepsilon^{src}_{t,1:N})$
  $z^{trg}_{t-1,1:N} = \mathrm{DDIM}(\varepsilon^{trg}_{t,1:N})$
end for
$M_{1:N} = \mathrm{binarize}_\alpha(\mathrm{mean}_{t\in[0.8\times T,T]}(\Delta\varepsilon_{t,1:N}))$
$\mathcal{M}^{inv} = M_{1:N}$
for $t = T, T-1, \cdots, 2$ **do** ▷ *Infer using InvEdit mask*
  $[f^t_1, \cdots, f^t_N] \leftarrow$ get `Up-block-2` features
  $\mathcal{N}^t_{i\pm}[p] = \mathrm{argmax}_q d(f^t_i[p], f^t_{i\pm1}[q]), 1 \le i \le N$
  $\hat{\mathcal{N}}^t_{i\pm} = \mathrm{Upsample}(\mathcal{N}^t_{i\pm})$ ▷ *upsample to match the dim of $\mathcal{Z}$ space*
  $o_{t,1:N} = \varepsilon_\theta(z^{src}_{t,1:N}, t_{emb}, \mathcal{C}^{trg}, \mathcal{J}^{trg}, \mathcal{M}_\phi)$ ▷ *UNet forward pass as in Sec.6*
  $o'_{t,1:N} = \varepsilon_\theta(z^{src}_{t,1:N}, t_{emb}, \mathcal{C}^{trg}, \{\mathcal{J}^{src}, \mathcal{J}^{trg}\}, \mathcal{M}^{inv})$ ▷ *UNet forward pass as in Sec.6*
  $z_{t-1} = \frac{1}{1+\mathcal{M}^{inv}}(\mathcal{M}^{inv} \odot \mathrm{DDIM}(o_{t,1:N}) + \mathrm{DDIM}(o'_{t,1:N}))$ ▷ *latent fusion as in Sec. 6*
  $\tilde{z}_{t-1,i}[p] = w_{-1}(M_i \odot z_{t-1,(i-1)}[\hat{\mathcal{N}}^t_{i-}(p)]) + w_0(M_i \odot z_{t-1,i}) + w_1(M_i \odot z_{t-1,(i+1)}[\hat{\mathcal{N}}^t_{i+}(p)]) + (1-M_i) \odot z_{t-1,i}$, if $t \ge T-5$ ▷ *inter-frame latent correction*
  Apply optional background preservation
end for
Output video frames = $\mathcal{D}(\tilde{z}_{1,1:N})$

Fig. 14, we present some additional results.

- **Video object editing:** In Fig. 13, we find that *GenVideo* is able to accurately identify the region of interest to be modified. In Fig. 13A, the *InvEdit* mask accurately identified the region of edit and modified the region from the source *rabbit* to the target *tiger* while keeping the *watermelon* intact. Similarly, in Fig. 13C, the *rabbit* was retained correctly and the region corresponding to the *watermelon* was edited to *cake* which has a different shape than the *watermelon*. Thus, *InvEdit* correctly handles the edits for varying shapes and sizes of objects. Results in Fig. 13B demonstrate the editing of a *silver swan* to a *small wooden boat*. In this result, the *InvEdit* mask helps in identifying regions that correspond to both *swan* and expected *boat* in order to edit the source video effectively even when they are of very different shapes and size here.
- **Style editing:** In Fig. 14, we present results of GenVideo for stylistic variation of the foreground object (in Fig. 14A) and stylistic variations of the entire frames in the video (in Fig. 14B and Fig. 14C). When editing the entire frames in the video, we skip performing the background preservation.
- **Zero-shot image editing:** In Fig. 12, we show additional results on zero-shot image editing capabilities of our approach.

References

[1] Omri Avrahami, Dani Lischinski, and Ohad Fried. Blended diffusion for text-driven editing of natural images. In *CVPR*, 2022. 1

[2] Omer Bar-Tal, Dolev Ofri-Amar, Rafail Fridman, Yoni Kasten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *ECCV*, 2022. 1, 2

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *ICCV*, 2021. 7

[4] Duygu Ceylan, Chun-Hao P. Huang, and Niloy J. Mitra. Pix2video: Video editing using image diffusion. In *ICCV*, 2023. 1, 2

[5] Wenhao Chai, Xun Guo, Gaoang Wang, and Yan Lu. Stablevideo: Text-driven consistency-aware diffusion video editing. In *ICCV*, 2023. 1, 2, 3, 6

[6] Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. Diffedit: Diffusion-based semantic image editing with mask guidance. In *ICLR*, 2023. 1, 2, 4

[7] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *NeurIPS*, 2021. 1

[8] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *ICCV*, 2023. 1, 2

[9] Michal Geyer, Omer Bar-Tal, Shai Bagon, and Tali Dekel.

Tokenflow: Consistent diffusion features for consistent video editing, 2023. 1, 2, 3, 6

[10] 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. 2023. 8

[11] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control, 2022. 1, 2

[12] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 1, 2

[13] 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, 2022. 2, 3

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

[15] Minguk Kang, Jun-Yan Zhu, Richard Zhang, Jaesik Park, Eli Shechtman, Sylvain Paris, and Taesung Park. Scaling up gans for text-to-image synthesis. *CVPR*, 2023. 2

[16] 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. *arXiv*, 2023. 2

[17] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv*, 2023. 4

[18] Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. Ablating concepts in text-to-image diffusion models. In *ICCV*, 2023. 1

[19] Yao-Chih Lee, Ji-Ze Genevieve Jang Jang, Yi-Ting Chen, Elizabeth Qiu, and Jia-Bin Huang. Shape-aware text-driven layered video editing demo. *arXiv preprint arXiv:2301.13173*, 2023. 3

[20] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv*, 2023. 4

[21] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control, 2023. 1, 2

[22] Zhengxiong Luo, Dayou Chen, Yingya Zhang, Yan Huang, Liang Wang, Yujun Shen, Deli Zhao, Jinren Zhou, and Tieniu Tan. Videofusion: Decomposed diffusion models for high-quality video generation. In *CVPR*, 2023. 2

[23] Chenlin Meng, Yang Song, Jiaming Song, Jiajun Wu, Jun-Yan Zhu, and Stefano Ermon. Sdedit: Image synthesis and editing with stochastic differential equations. In *ICLR*, 2022. 2

[24] Eyal Molad, Eliahu Horwitz, Dani Valevski, Alex Rav Acha, Yossi Matias, Yael Pritch, Yaniv Leviathan, and Yedid Hoshen. Dreamix: Video diffusion models are general video editors, 2023. 1, 2

[25] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *ICML*, 2021. 1

[26] Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. Glide: Towards photorealistic image generation and editing with text-guided diffusion models. In *ICML*, 2022. 1

[27] Aaron Van Den Oord et al. The frobnicatable foo filter, 2017. Neural discrete representation learning. In NeurIPS. 2

[28] Gaurav Parmar, Krishna Kumar Singh, Richard Zhang, Yijun Li, Jingwan Lu, and Jun-Yan Zhu. Zero-shot image-to-image translation. In *SIGGRAPH*, 2023. 1

[29] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. Styleclip: Text-driven manipulation of stylegan imagery. In *ICCV*, 2021. 2

[30] 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*, 2023. 1

[31] Chenyang Qi, Xiaodong Cun, Yong Zhang, Chenyang Lei, Xintao Wang, Ying Shan, and Qifeng Chen. Fatezero: Fusing attentions for zero-shot text-based video editing. In *ICCV*, 2023. 1, 2, 4, 6, 8

[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 *ICML*, 2021. 2, 7

[33] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents, 2022. 1, 2, 3, 4, 5

[34] A. Revanur, D. Basu, S. Agrawal, D. Agarwal, and D. Pai. Coralstyleclip: Co-optimized region and layer selection for image editing. In *CVPR*, 2023. 2

[35] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 1, 2, 3

[36] 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. In *NeurIPS*, 2022. 1, 2

[37] 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, 2022. 1, 2

[38] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *ICLR*, 2023. 2, 4

[39] Yizhi Song, Zhifei Zhang, Zhe Lin, Scott Cohen, Brian Price, Jianming Zhang, Soo Ye Kim, and Daniel Aliaga. Objectstitch: Object compositing with diffusion model. In *CVPR*, 2023. 5

[40] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *ECCV*, 2020. 5

[41] Narek Tumanyan, Michal Geyer, Shai Bagon, and Tali Dekel. Plug-and-play diffusion features for text-driven image-to-image translation. In *CVPR*, 2023. 1

[42] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*, 2018. 2

[43] 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 *ICCV*, 2023. 1, 2, 3, 5, 6, 8

[44] Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *arXiv*, 2022. 2

[45] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *ICCV*, 2023. 2

[46] Yuyang Zhao, Enze Xie, Lanqing Hong, Zhenguo Li, and Gim Hee Lee. Make-a-protagonist: Generic video editing with an ensemble of experts, 2023. 1, 2, 3, 4, 6

[47] Daquan Zhou, Weimin Wang, Hanshu Yan, Weiwei Lv, Yizhe Zhu, and Jiashi Feng. Magicvideo: Efficient video generation with latent diffusion models. *arXiv*, 2022. 1