

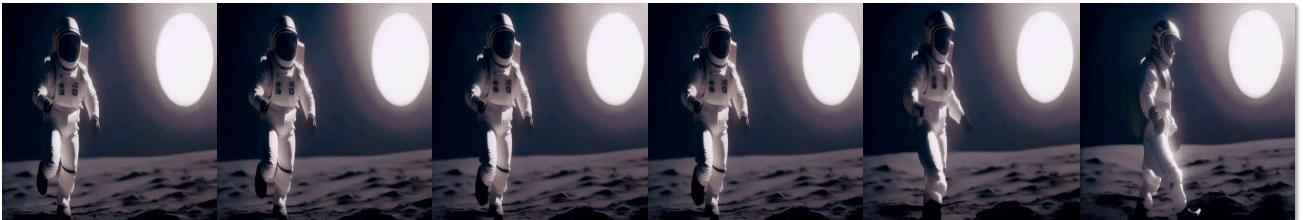
# Neutral Editing Framework for Diffusion-based Video Editing

Sunjae Yoon Gwanhyeong Koo Ji Woo Hong Chang D. Yoo

Korea Advanced Institute of Science and Technology (KAIST)

{sunjae.yoon, kookie, jiwoohong93, cd-yoo}@kaist.ac.kr

## Input Video



Target prompt: “A man jumps on the moon”



Non-rigid video editing



Target prompt: “A man gives a thumbs-up on the moon”

Fine-grained non-rigid video editing

Figure 1. Neutral editing framework enables the diffusion-based editing models to perform various text-based non-rigid editing such as motion variation of objects spanning from fine-grained variations to large dynamic variations while preserving fidelity to the input video.

## Abstract

*Text-conditioned image editing has succeeded in various types of editing based on a diffusion framework. Unfortunately, this success did not carry over to a video, which continues to be challenging. Existing video editing systems are still limited to rigid-type editing such as style transfer and object overlay. To this end, this paper proposes Neutral Editing (NeuEdit) framework to enable complex non-rigid editing by changing the motion of a person/object in a video, which has never been attempted before. NeuEdit introduces a concept of ‘neutralization’ that enhances a*

*tuning-editing process of diffusion-based editing systems in a model-agnostic manner by leveraging input video and text without any other auxiliary aids (e.g., visual masks, video captions). Extensive experiments on numerous videos demonstrate adaptability and effectiveness of the NeuEdit framework. The website of our work is available here: <https://neuedit.github.io/>*

## 1. Introduction

The recent success of generative frameworks [4, 10, 16] and large-scale models [5, 24, 25] provide surreal outputs sur-

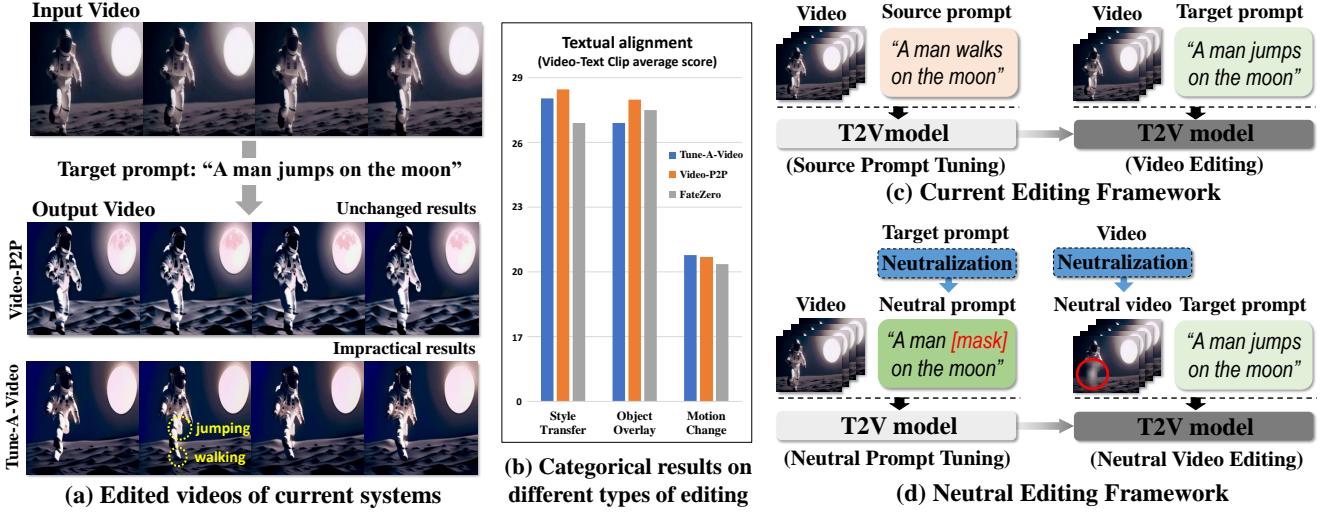


Figure 2. (a) Edited videos of current systems [19, 38] based on target prompt in terms of motion change. (b) Categorical analysis of different types of editing on videos of DAVIS [22]. Illustration of (c) current editing framework and (d) proposed neutral editing framework.

passing the boundaries of human capabilities. The diffusion models [6, 32, 34] lay the foundation for such innovative advancements, bridging a diverse range of large-scale generative models [27, 33]. To be specific, diffusion-based text-to-image (T2I) models [21, 27] synthesize natural images of high fidelity and further edit [14, 28] them by modifying specific attributes corresponding to the input text. Expanding the work in image, diffusion-based text-to-video (T2V) models [1, 38] also have been considered. Due to insufficient training resources about videos, in early work, significant technical contributions [13, 31] have been made to transfer the knowledge of pre-trained T2I models into the T2V models. Currently, researchers are striving to refine this text-based video generation into a more controlled and fine-grained approach by modifying specific attributes in a video corresponding to users' requirements from a text, ultimately performing text-based video editing.

In a formal definition of text-based video editing, as shown in Figure 2 (a), systems are given an input video and a target textual prompt describing desired modifications in the video such that they produce an edited video that conforms to the target prompt. To achieve this, editing systems largely perform two sequential processes: (1) video tuning and (2) video editing. In video tuning, the editing system is trained to generate the input video and to comprehend the contextual meaning of the video. In video editing, the system generates the variants of the input video that conform to the meaning of the target prompt. To provide necessary attributes for editing, pre-trained vision-language models [25, 27] also have to be integrated into the system.

Despite recent advancements in video editing systems, their capabilities are still restricted to rigid modifications within the realm of inpainting such as style transfer and

object overlay. To be specific, in Figure 2 (a), for a given target prompt (*e.g.* “A man jumps on the moon”) requiring non-rigid modifications by changing a motion of an object, current systems do not conform to the target prompt and return the original input video under over-fidelity. Otherwise, they often show impractical results by mixing up the original content (*i.e.* walking) and targeted content (*i.e.* jumping). In Figure 2 (b), our categorical analysis of textual alignment with video according to different types of editing (*i.e.* style transfer, object overlay, motion change) demonstrates that current systems are facing difficulties in changing a motion in a video. Therefore the results of complex non-rigid editing are still unsatisfactory.

One of the reasons for the unsatisfactory editing is rooted in a conventional tuning and editing process within diffusion-based editing frameworks. As illustrated in Figure 2 (c), current editing frameworks require additional input caption about the video (*i.e.* source prompt) in the tuning process. After tuning, the model edits the video based on a target prompt. However, employing a source prompt leads to a functionally unnecessary tuning of content (*e.g.* ‘walking’ in the video, which is unrelated to the intended editing (*e.g.* ‘jumping’) and results in suboptimal editing. Furthermore, the outcomes are vulnerable to the variants of source prompts. Therefore, frameworks employing source prompt are inadequate for effective text-based video editing.

To this end, we propose Neutral Editing (NeuEdit) framework that performs effective video editing including non-rigid editing to a video with only a target prompt. As shown in Figure 2 (d), the NeuEdit framework introduces a novel concept of *neutralization*, which enables current editing systems to conduct (1) neutral prompt tuning and (2) neutral video editing. The neutral prompt tuning refers to

tuning a model based on a neutral prompt. This prompt (*e.g.* “A man [mask] on the moon”) is a text that reduces<sup>1</sup> factors (*e.g.* “jumps”) contributing to editing from a target prompt, allowing a model to tune a video without relying on a source prompt. Furthermore, the target prompt holds effective differences from this neutral prompt by the factors related to editing. To implement a neutral prompt, we first introduce a neutralization which refers to disentangling a factor related to editing in the input. After tuning with the neutral prompt, the T2V model performs video editing. Our studies found that current models struggle with non-rigid editing, primarily due to constraints imposed by the original content in the input video (*e.g.* Figure 2 (a)). To address this, we also construct a neutral video by applying neutralization which reduces the influence of original content in a region of the video to be edited, such that it amplifies the possibility of non-rigid editing. NeuEdit can be applied to diffusion-based editing systems in a model-agnostic manner, enhancing various editing including object motion change. Extensive experiments validate its adaptability and visual effectiveness.

## 2. Related Works

### 2.1. Diffusion-based generative models

Deep diffusion models [10, 33] exhibit significant capability by outperforming the prior best qualities of generative adversarial networks [8]. Applying diffusion to a text-to-image (T2I) generation, significant advancements have been observed in image generation, where diffusion-based T2I models [26, 29] produce high-fidelity images from textual inputs. Recently, the T2I models have expanded their visual generative capabilities into the domain of videos to perform text-to-video (T2V) generation. Earlier studies [12, 13, 37] in T2V generation modified the T2I model by introducing a temporal axis for video data, thus transferring pre-trained knowledge from the T2I model. To enhance temporal consistency in generated frames, temporal attentions [11, 31] are also designed. Recently, these diffusion-based models have been successful in various generative works including inpainting and super-resolution [20, 30]. Among them, visual editing emerges as a new challenge to perform controlling and reasoning about selective synthesizing, which is discussed in detail below.

### 2.2. Image and video editing

Text-based image editing aims to edit a given image based on text descriptions. To perform this, DiffusionCLIP [15] first proposes a tuning-editing framework of diffusion model based on CLIP embedding and Prompt-To-Prompt [9] proposes weight blending to perform effective rigid editing in this framework. For efficient editing, InstructPix2Pix

<sup>1</sup>Masking is an intuitive approach for reducing the factors. See also other approaches in Method.

[2] design zero-shot edits without fine-tuning. Similar to work in the image, video editing also has expanded. Especially to keep temporal consistency, several technical solutions are introduced such as layered editing [3] and attention control [19]. However, current video editing is limited to rigid types of editing and still challenging to dynamic motion change. Thus, NeuEdit first performs complex non-rigid editing in a video based on a text.

## 3. Preliminaries

### 3.1. Denoising diffusion probabilistic models

Denoising diffusion probabilistic models (DDPMs) [10] are parameterized Markov chains to reconstruct a sequence of data  $\{x_1, \dots, x_T\}$ . Given raw data  $x_0$ , the Markov transition gradually adds Gaussian noise upto  $x_T$  using  $q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{\alpha_t}x_{t-1}, (1 - \alpha_t)I)$  under pre-defined schedule  $\alpha_t$  following  $t = 1, \dots, T$ . This process is referred to as a *forward process* of the diffusion model. In the *reverse process*, the diffusion model approximates the  $q(x_{t-1}|x_t)$  using trainable Gaussian transitions  $p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta(x_t, t))$  starting at normal distribution  $p(x_T) = \mathcal{N}(x_T; 0, I)$ . The training objective is to maximize log-likelihood  $\log(p_\theta(x_0))$ , where we can also apply variational inference by maximizing the variational lower bound of this. This makes a closed-form of KL divergence<sup>2</sup> between the distributions of  $p_\theta$  and  $q$  while optimizing the parameter  $\theta$ . The beauty of DDPM is that this process can be summarized as denoising network  $\epsilon_\theta(x_t, t)$  for predicting noise  $\epsilon \sim \mathcal{N}(0, I)$  as given below:

$$\mathbb{E}_{x, \epsilon \sim \mathcal{N}(0, 1), t \sim \mathcal{U}\{1, T\}}[\|\epsilon - \epsilon_\theta(x_t, t)\|_2^2]. \quad (1)$$

To keep the robustness in all steps,  $t$  is sampled from the discrete uniform distribution  $\mathcal{U}\{1, T\}$ . Based on trained  $\epsilon_\theta$ , denoising is performed, where denoising diffusion implicit model (DDIM) [33] has been a popular choice for the denoising by a small number of sampling steps as below:

$$x_{t-1} = \sqrt{\frac{\alpha_{t-1}}{\alpha_t}}x_t + \left( \sqrt{\frac{1 - \alpha_{t-1}}{\alpha_{t-1}}} - \sqrt{\frac{1 - \alpha_t}{\alpha_t}} \right) \cdot \epsilon_\theta. \quad (2)$$

### 3.2. Text-guided diffusion model

The text-guided diffusion model is a DDPM that restores the output data  $x_0$  from random noise with a guided condition of a text prompt  $\mathcal{T}$ . Thus, the training objective is also formulated with this condition under latent space to interact with textual modality as  $\mathbb{E}_{z, \epsilon, t}[\|\epsilon - \epsilon_\theta(z_t, t, \mathbf{c})\|_2^2]$ , where  $z_t = E(x_t)$  is a latent noise encoding (*e.g.* VQ-VAE [36]) and  $\mathbf{c} = \psi(\mathcal{T})$  is conditional textual embedding (*e.g.* CLIP [25]). In video editing,  $z_t$  is a latent encoding of video data, and  $\epsilon_\theta$  can be pre-trained video diffusion networks.

<sup>2</sup>See the detailed proof in Appendix D.

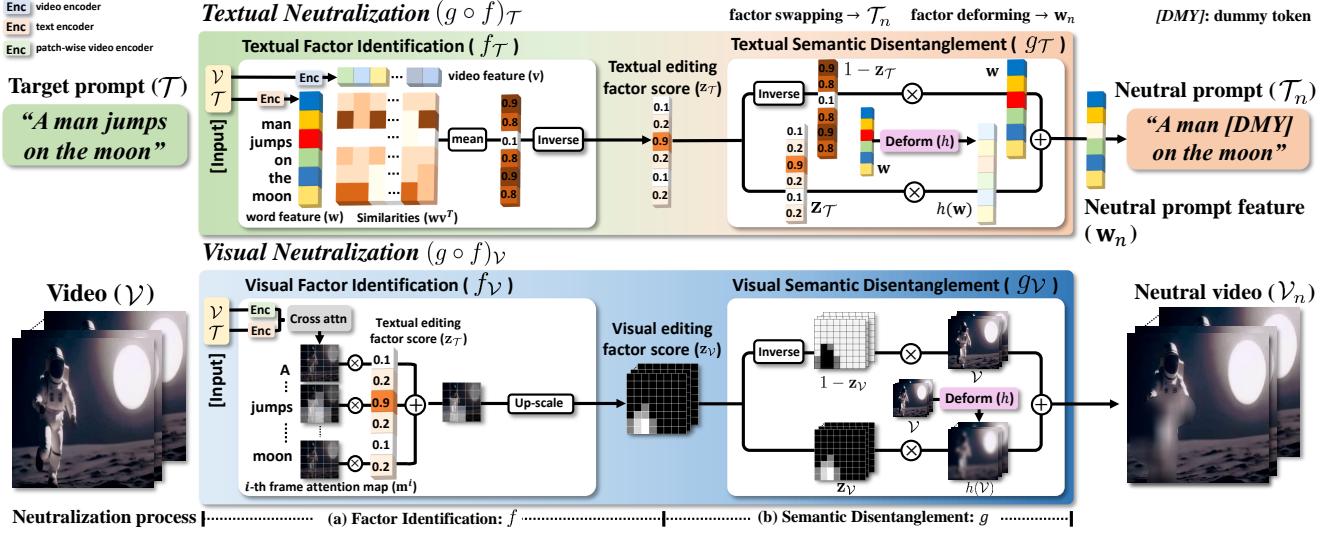


Figure 3. Illustration of neutralization composed of (a) factor identification  $f$  and (b) semantic disentanglement  $g$ . The  $f$  localizes editing factors in video and text and the  $g$  produces neutralized video and text via semantically reducing the editing factors in the video and text. The textual neutralization  $(g \circ f)_T$  and visual neutralization  $(g \circ f)_V$  are the applications of  $f$  and  $g$  for performing the neutralization.

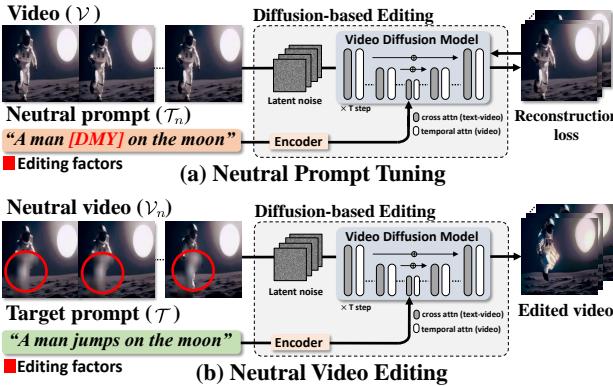


Figure 4. Illustration of Neutral Editing framework composed of (a) neutral prompt tuning and (b) neutral video editing. The neutral prompt is a text that removes factors contributing to editing from the target prompt. The neutral video is a video that reduces factors related to editing enabling the systems to perform effective editing including non-rigid editing without auxiliary inputs.

## 4. Neutral Editing Framework

The Neutral Editing (NeuEdit) framework aims to enhance existing diffusion-based video editing systems, enabling more effective non-rigid modifications in a model-agnostic approach using only an input video and a target prompt. To achieve this, as shown in Figure 4 (a), NeuEdit introduces the novel concept of neutralization which enables current editing systems to conduct (1) neutral prompt tuning and (2) neutral video editing. The neutral prompt tuning refers to model tuning based on a neutral prompt. This neutral prompt is a text that reduces factors (*e.g.* specific words

or features) contributing to editing from a target prompt, effectively resolving the issue of spurious reliance on additional source prompt of current editing systems. Henceforth, the target prompt keeps effective differences from the neutral prompt by the factors related to editing. After tuning, the systems edit a video based on the target prompt. However, our studies found that original content within the editing region in the video imposes constraints on non-rigid edits. Thus we construct a neutral video that sensibly reduces the influence of original content in a region of video to be edited, amplifying the possibilities of non-rigid editing. The neutral prompt and neutral video are constructed by following our proposed neutralization operation.

### 4.1. Neutralization

Neutralization aims to disentangle factors contributing to editing in the input modality (*i.e.* video, text). To perform this, as shown in Figure 3, it comprises two sequential processes: (1) factor identification  $f$  and (2) semantic disentanglement  $g$ . To provide a formal definition of the neutralization, it takes inputs of target prompt  $T$  and video  $V$  and produces neutral prompt  $T_n$  and neutral video  $V_n$  as below:

$$T_n, V_n = (g \circ f)(T, V), \quad (3)$$

where  $f$  is factor identification which localizes the editing factors within each modality, and  $g$  is semantic disentanglement that produces the modality that semantically reduces the meaning of the identified editing factors. The implementations of  $f$  and  $g$  are specified depending on the target (*i.e.* text or video) of neutralization, referred to as textual neutralization and visual neutralization in the following.

**Textual Neutralization** Textual neutralization  $(g \circ f)_{\mathcal{T}}$  at the top of Figure 3 is an application of the neutralization to the target prompt, which obtains neutral prompt  $\mathcal{T}_n$  from target prompt  $\mathcal{T}$  and video  $\mathcal{V}$ . To implement the factor identification  $f_{\mathcal{T}}$  and the semantic disentanglement  $g_{\mathcal{T}}$  in the textual neutralization, we first define the editing factors in the target prompt as ‘words’ contributing to editing. Thus, the  $f_{\mathcal{T}}$  aims to localize these words in the target prompt  $\mathcal{T}$ . Since the target prompt is a description of desired modifications in the current video, the prompt and the video exhibit semantic misalignment due to the words associated with the modifications (*i.e.* editing factors). Intrigued by this observation, we measure all words in  $\mathcal{T}$  based on their cosine similarities with  $\mathcal{V}$  to localize the words exhibiting low similarities as the editing factors. Therefore, we define a textual factor identification as  $f_{\mathcal{T}}(\mathcal{T}, \mathcal{V})$  to produce scores about localization of editing factor in the target prompt based on their similarity scores as given below:

$$f_{\mathcal{T}}(\mathcal{T}, \mathcal{V}) = 1 - \text{mean}(\mathbf{w}\mathbf{v}^T) \in \mathbb{R}^M, \quad (4)$$

where the  $\mathbf{w} = \psi_T(\mathcal{T}) \in \mathbb{R}^{M \times d}$  is  $d$ -dimensional normalized word features of target prompt. The  $M$  is the number of words and  $\psi_T$  is the CLIP [25] text encoder. The  $\mathbf{v} = \psi_I(\mathcal{V}) \in \mathbb{R}^{L \times d}$  is  $d$ -dimensional video features with frame length  $L$ , where  $\psi_I$  is the CLIP image encoder.  $\text{mean}(\cdot)$  is a mean-pooling along frame axis. As editing factors have low similarity scores with video, we inverse the scores by subtracting them from one, finally producing the textual editing factor score denoted as  $\mathbf{z}_{\mathcal{T}} = f_{\mathcal{T}}(\mathcal{T}, \mathcal{V})$ . This score  $\mathbf{z}_{\mathcal{T}}$  is utilized for the following textual semantic disentanglement.

The textual semantic disentanglement  $g_{\mathcal{T}}$  aims to build a neutral prompt  $\mathcal{T}_n$  from a target prompt  $\mathcal{T}$  by semantically reducing editing factors in the  $\mathcal{T}$ . To perform this,  $g_{\mathcal{T}}$  employs the editing factor score  $\mathbf{z}_{\mathcal{T}}$  to identify the editing factors and reduce their meaning by disentangling them. To be specific, we present two technical contributions about the disentangling methods: (1) factor swapping and (2) factor deforming. The factor swapping is to swap the identified editing factors with other words. To the scores above a specific value  $s$  (*e.g.* 0.7) on  $\mathbf{z}_{\mathcal{T}}$  (*e.g.* [0.1, 0.9,  $\dots$ , 0.2]), their corresponding words are decided as editing factors, denoting them as  $W_{\mathbf{z}_{\mathcal{T}} > s}$  in word space, where  $\mathbf{z}_{\mathcal{T}} > s$  is the indices of words in target prompt  $\mathcal{T}$  of a higher score than  $s$ . Thus, the  $W_{\mathbf{z}_{\mathcal{T}} > s}$  are swapped with other word tokens, where to mitigate a semantic intervention by swapping tokens, we define a dummy token as  $\langle \text{DMY} \rangle$  for the swapping. As a result, the textual semantic disentanglement with factor swapping ultimately produces a neutral prompt  $g_{\mathcal{T}}(\mathbf{z}_{\mathcal{T}}) = \mathcal{T}_n = [W_1, \dots, W_i, \dots, W_M]$  including  $W_{\mathbf{z}_{\mathcal{T}} > s} = \langle \text{DMY} \rangle$ , where  $W_i$  denotes corresponding  $i$ -th word in the target prompt  $\mathcal{T}$ . Although the  $\mathcal{T}_n$  with factor swapping maintains a distinct difference<sup>3</sup>

<sup>3</sup>Experimental studies in Appendix F provide optimal region of the  $s$ .

with the  $\mathcal{T}$  by the editing factors  $W_{\mathbf{z}_{\mathcal{T}} > s}$ , it relies on a heuristic manner in selecting the editing factor and is difficult to distinguish the difference among the factors. To this end, we further devised a feature-level disentanglement referred to as factor deforming. To be specific, it first disentangles the target prompt features  $\mathbf{w} \in \mathbb{R}^{M \times d}$  into a format of linear combination using factor score  $\mathbf{z}_{\mathcal{T}} \in \mathbb{R}^M$  as  $\mathbf{w} = \mathbf{z}_{\mathcal{T}} \circ \mathbf{w} + (1 - \mathbf{z}_{\mathcal{T}}) \circ \mathbf{w}$ , where  $\circ$ <sup>4</sup> is element-wise multiplication with broadcasting. After that, we deform the features attended by  $\mathbf{z}_{\mathcal{T}}$  using deformable operation  $h(\cdot)$  as:

$$\mathbf{w}_n = \mathbf{z}_{\mathcal{T}} \circ h(\mathbf{w}) + (1 - \mathbf{z}_{\mathcal{T}}) \circ \mathbf{w}. \quad (5)$$

This format selectively deforms the text features concerning the editing factor while keeping disparities among features of editing factors. We apply feature down-scaling for the deforming<sup>5</sup> as  $h(\mathbf{w}) = \alpha \times \mathbf{w}$  with scaler  $0 \leq \alpha < 1$ . For  $\alpha = 0$ , all deformed features become identical, working similarly to the factor swapping. Finally, we define a neutral prompt features  $g_{\mathcal{T}}(\mathbf{z}_{\mathcal{T}}) = \mathbf{w}_n \in \mathbb{R}^{M \times d}$ , which is utilized in tuning process of NeuEdit framework.

**Visual Neutralization** Visual neutralization  $(g \circ f)_{\mathcal{V}}$  at the bottom of Figure 3 is another application of neutralization to the input video, which produces neutral video  $\mathcal{V}_n$  from target prompt  $\mathcal{T}$  and video  $\mathcal{V}$ . Neutral video improves the effectiveness of editing by reducing the influence of the original content in a region to be edited. To construct a factor identification  $f_{\mathcal{V}}$  and semantic disentanglement  $g_{\mathcal{V}}$  in the visual neutralization, we also define the editing factors in the video as ‘pixels’ contributing to editing. Thus the  $f_{\mathcal{V}}$  aims to localize these pixels in the input video  $\mathcal{V}$ . As editing models’ multi-modal interaction modules (*i.e.* cross-attention)<sup>6</sup> contain information about the interactions between texts and frames, we employ this information to build the  $f_{\mathcal{V}}$ . We first embed each  $i$ -th frame of video into patch-wise features as  $\mathbf{p}^i \in \mathbb{R}^{(W_p \times H_p) \times d}$  and perform cross attention with a target prompt features  $\mathbf{w} \in \mathbb{R}^{M \times d}$  to get cross attention maps as  $\mathbf{m}^i \in \mathbb{R}^{(W_p \times H_p) \times M}$ , where  $(W_p \times H_p)$  is the number patches in the frame and  $M$  is the number of words in the target prompt. Among the attention maps  $\mathbf{m}^i$ , we highlight the maps related to the editing using textual editing factor score  $\mathbf{z}_{\mathcal{T}} \in \mathbb{R}^{M \times 1}$  as (Please see also illustration in the bottom of Figure 3 for clear understanding):

$$\mathbf{z}_{\mathcal{V}}^i = \mathbf{m}^i \mathbf{z}_{\mathcal{T}} \in \mathbb{R}^{W_p \times H_p}, \quad (6)$$

where  $\mathbf{z}_{\mathcal{V}}^i$  is the  $i$ -th frame visual editing factor score. After restoring  $\mathbf{z}_{\mathcal{V}}^i$  up to the original frame scale ( $W \times H$ ) and aggregating all frames, we finally define the visual editing

<sup>4</sup>Here,  $\circ$  is different from composite operation in  $f \circ g$

<sup>5</sup>Appendix F also provides other factor deforming methods

<sup>6</sup>Video editing is based on pre-trained knowledge (*e.g.* Stable Diffusion [27]) with multi-modal attention to provide required modifications.

identification as  $f_{\mathcal{V}}(\mathcal{T}, \mathcal{V}) = \mathbf{z}_{\mathcal{V}} \in \mathbb{R}^{L \times (W \times H)}$ , where  $L$  is the number of video frames. The visual factor score  $\mathbf{z}_{\mathcal{V}}$  is utilized for the following visual semantic disentanglement.

The visual semantic disentanglement  $g_{\mathcal{V}}$  aims to build neutral video  $\mathcal{V}_n$  from input video  $\mathcal{V}$  by reducing the meaning of editing factors in the video. To this, based on visual editing factor score  $\mathbf{z}_{\mathcal{V}}$ , the  $g_{\mathcal{V}}$  identifies factors contributing to editing at a pixel level and semantically reduces them. Similar to textual semantic disentanglement  $g_{\mathcal{T}}$ , we apply a factor deforming by separating video pixels into two groups of pixels and deforming a group related to editing as below:

$$\mathcal{V}_n = \mathbf{z}_{\mathcal{V}} \circ h(\mathcal{V}) + (1 - \mathbf{z}_{\mathcal{V}}) \circ \mathcal{V}, \quad (7)$$

where we applied Gaussian blurring to deform the video concerned about editing factors as  $h(\mathcal{V}) = \mathcal{V} * G$  with Gaussian kernel  $G(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$ . Therefore the visual semantic disentanglement is summarized as  $g_{\mathcal{V}}(\mathbf{z}_{\mathcal{V}}) = \mathcal{V}_n$ . The  $\mathcal{V}_n$  is used for editing instead of the  $\mathcal{V}$ , alleviating restrictions imposed by the original content and facilitating dynamic variations within the editing areas.

## 4.2. Plug-and-play NeuEdit framework

We integrate the neutral prompt  $\mathcal{T}_n$  and neutral video  $\mathcal{V}_n$  into diffusion-based video editing system. The editing system includes two processes: (1) video tuning and (2) video editing, where the neutral prompt is introduced in the tuning process as a guided condition features  $\mathbf{w}_n = \psi_T(\mathcal{T}_n)$  for the training objective of a text-guided diffusion model (*i.e.* refer details in Section 3.1) as given below:

$$\mathbb{E}_{z, \epsilon, t} [\|\epsilon - \epsilon_{\theta}(z_t, t, \mathbf{w}_n)\|_2^2], \quad (8)$$

After tuning with neutral prompt, the model performs editing by denoising an initial latent noise with the input condition of target prompt  $\mathcal{T}$ , producing edited video  $\mathcal{V}_{\text{edit}}$  as:

$$\mathcal{V}_{\text{edit}} = \text{Denoise}(f_{\text{init}}(\mathcal{V}_n), \mathcal{T}), \quad (9)$$

where  $\text{Denoise}(\cdot, \cdot)$  is the reverse process of diffusion model by gradual denoising under the sequential process using Equation 2 and  $f_{\text{init}}$  is initial latent noise encoding such as DDIM inversion<sup>7</sup> for enhanced reconstruction based on input video, where we provide neutral video  $\mathcal{V}_n$  instead of  $\mathcal{V}$  to improve dynamic modifications in editing region.

## 5. Experiment

### 5.1. Experimental Settings

**Implementation Details.** For textual factor identification  $f_{\mathcal{T}}$ , CLIP model (ViT-L/14) [25] is used for text and image features. The attention map for visual neutralization is based on the cross attention module of video diffusion model [38] (*i.e.* Stable Diffusion v1.5). The experimental settings are  $W = H = 512$ ,  $W_p = H_p = 16$  on NVIDIA A100 GPU.

<sup>7</sup>Appendix E gives details of DDIM inversion and gradual denoising.

**Dataset and Baselines.** We validate videos on DAVIS [22] and LOVEU-TGVE [39], which are video editing challenge dataset<sup>8</sup> comprising 32 to 128 frames of each. NeuEdit framework is validated about non-rigid/rigid editing on recent editing systems including Tune-A-video, Video-P2P, FateZero [23] on their public codes.

### 5.2. Evaluation Metric

We validate editing results based on four assessments: (1) textual alignment, (2) fidelity to input video, (3) frame consistency, and (4) human preference. The textual alignment measures the semantic alignment between a target prompt and an edited video using the CLIP score and PickScore [18]. The PickScore approximates human preferences by a large-scale trained model. The fidelity measures the preservation of original content in the unedited region<sup>9</sup> using peak signal-to-noise ratio (PSNR), learned perceptual image patch similarity (LPIPS), and structural similarity index measure (SSIM). The frame consistency measures image CLIP scores between sequential frames and measures Fréchet video distance (FVD) to evaluate the naturalness of videos. For the human evaluation, we investigate the preferences of edited videos according to the target prompt between the editing models and the models with NeuEdit.

### 5.3. Experimental Results

**Qualitative Comparisons.** Figure 5 shows the qualitative results of recent editing systems [19, 38] with the proposed neutral editing framework. (See also qualitative results in Appendix G). To validate the qualitative effectiveness of neutralization, we perform case studies in terms of two types of editing: (a) non-rigid editing and (b) rigid editing. In the case of non-rigid editing, current editing systems' results are not aligned with the target prompt, generating original input videos or incorrectly synthesizing original content (*e.g.* trees) and required variations (*e.g.* wings). However, these models with the NeuEdit framework demonstrate effective non-rigid editing on various targets including human and object. It is also notable that motion editing about thumbs-up is conditionally performed according to the visibility of the skier's hand, this is because the visual neutralization is sensibly applied to visible editing factors (*i.e.* hand). We provide further analysis of this in Section 5.4. In the case of rigid editing (*i.e.* top: style transfer, bottom: object overlay), the current models and the models with NeuEdit provide qualitatively proper modifications. But, in detail, only the models under NeuEdit framework maintain a finer fidelity to the unedited region (*i.e.* yellow box) in the video. It is considered that feature commonality between neutral prompt and target prompt, excluding the editing factor, improves selective fidelity in the model. At

<sup>8</sup><https://sites.google.com/view/loveucvpr23/track4>

<sup>9</sup>Detailed explanations of capturing unedited region are in Appendix C.

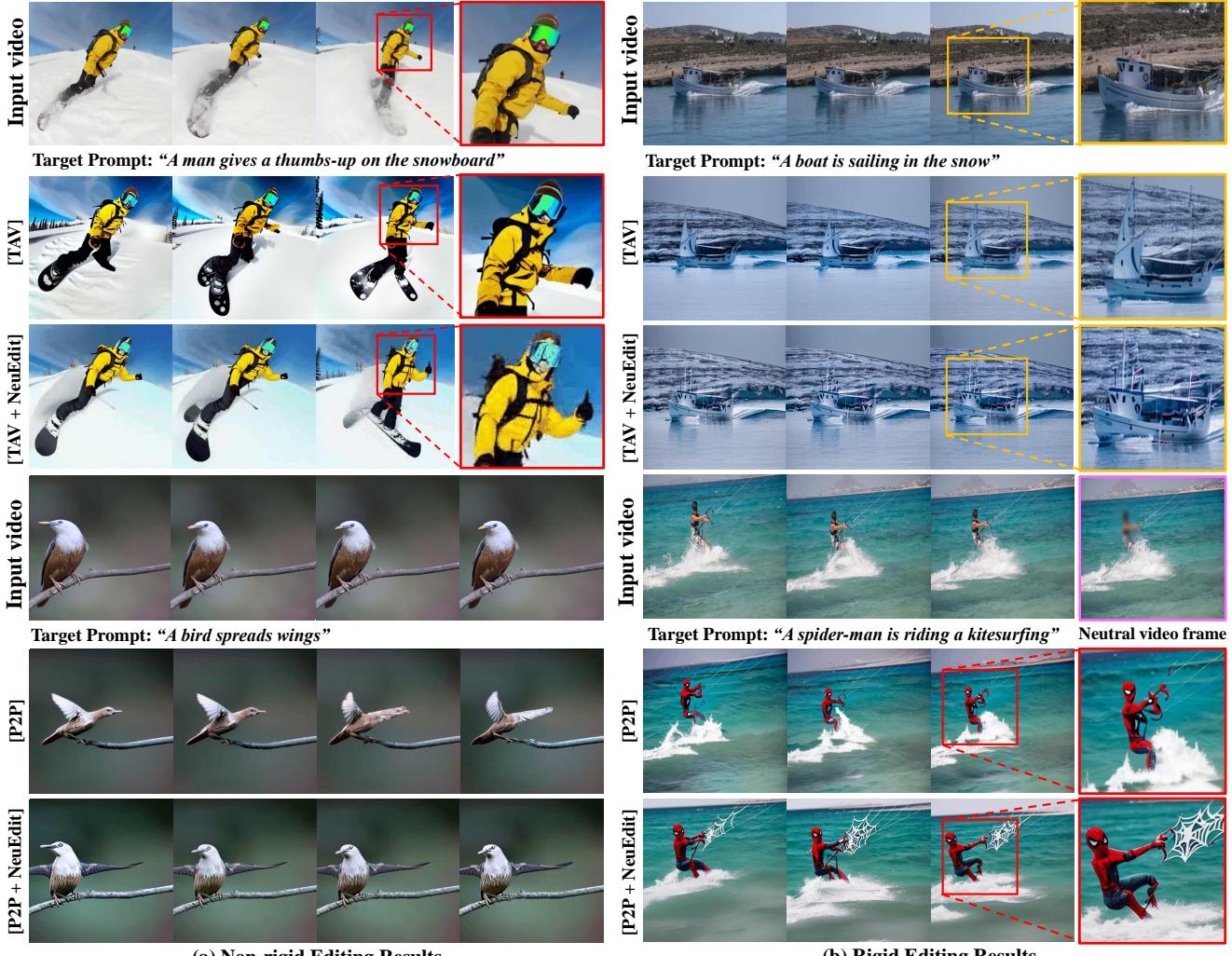


Figure 5. Qualitative result about applying NeuEdit framework on recent editing systems according to (a) non-rigid editing (*i.e.* motion variation) and (b) rigid editing (*i.e.* style transfer, object overlay). (TAV: Tune-A-Video [38], P2P: Video-P2P [19]).

the bottom, we also edited a man in kite-surfing to resemble Spider-Man. Interestingly, the model with NeuEdit also modifies the action of catching a kite as catching it with a spider web. We consider the neutral video shown in the pink box contributes to effective editing, mitigating the restriction by the original content in the area to be edited.

**Quantitative Results.** Table 1 presents evaluations of the non-rigid/rigid editing on videos of DAVIS and TGVE<sup>10</sup> of recent editing systems with the NeuEdit about four assessments (*i.e.* alignment, fidelity, consistency, human evaluation). The effectiveness of NeuEdit is confirmed in all models. Especially in non-rigid editing, textual alignment is significantly improved. Fidelity evaluates the preservation of unedited areas in video, such that we measure fidelity after

masking identical regions related to editing. The fidelity is effectively enhanced in the tuning-based models (*i.e.* TAV, Video-P2P) than tuning-free model (*i.e.* FateZero), which tells that neutral prompt contributes to improving fidelity.

#### 5.4. Ablation Study

Figure 6 presents ablation studies about neutral video and neutral prompt in terms of motion editing. In Figure 6 (a), the current model is ineffective in motion editing (*i.e.* thumbs-up), resulting in a video closely resembling the input video. We applied neutralization to this model, implementing its process step by step. In (b) and (c), we show results from neutral prompt tuning applied to the editing model. The results (b) from neutral prompt with factor swapping, while (c) stems from neutral prompt feature with factor deforming. As shown in yellow circles in (b) and (c),

<sup>10</sup>Appendix C provides further results on UCF101[35]

	Textual Alignment		Fidelity to Input Video			Frame Consistency		Human
	CLIP* $\uparrow$	PickScore $\uparrow$	PSNR $\uparrow$	LPIPS $\downarrow$	SSIM $\uparrow$	CLIP $^\dagger$ $\uparrow$	FVD $\downarrow$	Preference $\uparrow$
TAV [38]	22.6 / 27.1	19.5 / 20.2	13.1 / 14.1	0.1813 / 0.1934	0.621 / 0.653	0.921 / 0.952	3481 / 3392	0.14
TAV + NeuEdit	27.6 / 28.5	20.6 / 20.9	19.2 / 18.9	0.1438 / 0.1411	0.706 / 0.711	0.962 / 0.971	3270 / 3151	0.86
FateZero [23]	21.2 / 26.1	19.4 / 20.1	14.1 / 13.6	0.1653 / 0.1731	0.636 / 0.643	0.958 / 0.960	3319 / 3106	0.34
FateZero + NeuEdit	27.3 / 28.7	20.1 / 21.2	16.8 / 17.3	0.1621 / 0.1724	0.637 / 0.657	0.969 / 0.968	3209 / 3071	0.66
Video-P2P [19]	22.5 / 27.2	19.6 / 20.0	14.7 / 15.5	0.1738 / 0.1814	0.645 / 0.677	0.961 / 0.958	3231 / 3095	0.38
Video-P2P + NeuEdit	<b>27.9 / 29.6</b>	<b>20.9 / 21.3</b>	<b>19.3 / 19.8</b>	<b>0.1298 / 0.1388</b>	<b>0.727 / 0.733</b>	<b>0.966 / 0.973</b>	<b>3135 / 2953</b>	0.62

Table 1. Evaluations about edited videos based on DAVIS and TGVE in terms of non-rigid/rigid type editing corresponding to textual alignment, fidelity to input video, frame consistency, and human preference. CLIP\*: text-video clip score, CLIP $^\dagger$ : image-image clip score.

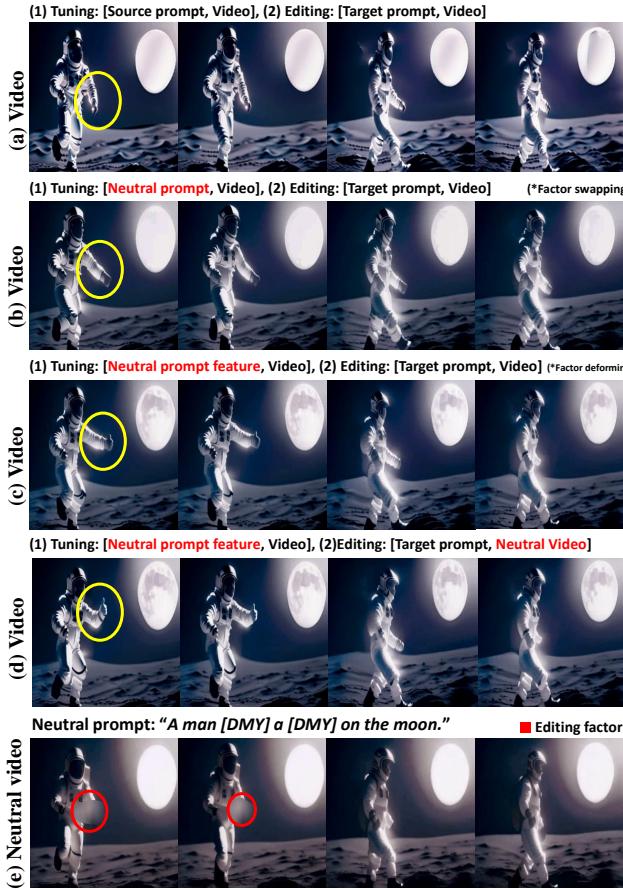


Figure 6. Ablation studies about neutral video and neutral prompt. The input video and target are shown in Figure 1.

they show the astronaut slightly extends his arm to give a thumbs-up. The action in (c) seems more effective than (b). We consider that neutral prompt features identify differences among editing factors to be emphasized. Nevertheless, unnatural motion editing persists due to constraints posed by the original arm motion. Video (d) is edited results with neutral video (e), where it showcases dynamic

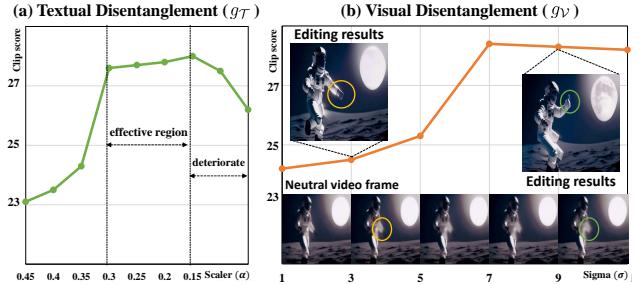


Figure 7. Sensitivity analysis of textual and visual semantic disentanglement. The  $g_T$  employs feature down-scaling with scaler  $\alpha$  and  $g_V$  employs Gaussian blurring with  $\sigma$ . Single frame of neutral video according to the blurring by  $\sigma$  is shown below the curve (b).

motion edits, such as bending arms for a thumbs-up. Notably, neutralization is sensibly applied to editing factors. When the hand disappears behind the leg, the neutralization effect (*i.e.* red circle) vanishes, facilitating natural motion recovery akin to temporal motion change in the result. Figure 7 shows sensitivity analysis of factor deforming in textual and visual semantic disentanglement. The textual disentanglement  $g_T$  is modulated by feature down-scaler  $\alpha$ . Neutral prompt features are effective for values below 0.3, but below 0.15, it shows deterioration, damaging the distinguishability among editing factors. For the disentanglement  $g_V$ , the Gaussian blurring is controlled by the  $\sigma$ . Effective motion editing occurs for  $\sigma > 3$ , aligning with ambiguity caused by blurring (*i.e.* yellow to green in Figure 7 (b)). In lower values, it is restricted by the original motion of video.

## 6. Conclusion

This paper proposes diffusion-based video editing framework referred to as Neutral Editing (NeuEdit), which enables complex non-rigid editing of a person/object in a video. NeuEdit introduces a ‘neutralization’ concept to enhance the current tuning-editing process of diffusion-based editing systems in a model-agnostic manner. Extensive experiments validate its editability and visual effectiveness.

## References

- [1] Omer Bar-Tal, Dolev Ofri-Amar, Rafaail Fridman, Yoni Kassten, and Tali Dekel. Text2live: Text-driven layered image and video editing. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XV*, pages 707–723. Springer, 2022. 2
- [2] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 3
- [3] Wenhao Chai, Xun Guo, Gaoang Wang, and Yan Lu. Stable-video: Text-driven consistency-aware diffusion video editing. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 23040–23050, 2023. 3
- [4] Antonia Creswell, Tom White, Vincent Dumoulin, Kai Arulkumaran, Biswa Sengupta, and Anil A Bharath. Generative adversarial networks: An overview. *IEEE signal processing magazine*, 35(1):53–65, 2018. 1
- [5] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 1
- [6] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Advances in Neural Information Processing Systems*, 34:8780–8794, 2021. 2
- [7] Patrick Esser, Johnathan Chiu, Parmida Atighehchian, Jonathan Granskog, and Anastasis Germanidis. Structure and content-guided video synthesis with diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7346–7356, 2023. 4
- [8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020. 3
- [9] Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. Prompt-to-prompt image editing with cross attention control. *arXiv preprint arXiv:2208.01626*, 2022. 3
- [10] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 1, 3
- [11] 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. 3
- [12] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *arXiv preprint arXiv:2204.03458*, 2022. 3
- [13] 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, 3
- [14] Bahjat Kawar, Shiran Zada, Oran Lang, Omer Tov, Huiwen Chang, Tali Dekel, Inbar Mosseri, and Michal Irani. Imagic: Text-based real image editing with diffusion models. *arXiv preprint arXiv:2210.09276*, 2022. 2
- [15] Gwanghyun Kim, Taesung Kwon, and Jong Chul Ye. Diffusionclip: Text-guided diffusion models for robust image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2426–2435, 2022. 3
- [16] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013. 1
- [17] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023. 1
- [18] Yuval Kirstain, Adam Polyak, Uriel Singer, Shahbuland Matiana, Joe Penna, and Omer Levy. Pick-a-pic: An open dataset of user preferences for text-to-image generation. *arXiv preprint arXiv:2305.01569*, 2023. 6
- [19] Shaoteng Liu, Yuechen Zhang, Wenbo Li, Zhe Lin, and Jiaya Jia. Video-p2p: Video editing with cross-attention control. *arXiv preprint arXiv:2303.04761*, 2023. 2, 3, 6, 7, 8
- [20] Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, and Luc Van Gool. Repaint: Inpainting using denoising diffusion probabilistic models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11461–11471, 2022. 3
- [21] Alex 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. *arXiv preprint arXiv:2112.10741*, 2021. 2
- [22] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbelaez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017. 2, 6, 4
- [23] 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. *arXiv preprint arXiv:2303.09535*, 2023. 6, 8, 3
- [24] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018. 1
- [25] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 1, 2, 3, 5, 6
- [26] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022. 3
- [27] 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. 2, 5, 1
- [28] 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. *arXiv preprint arXiv:2208.12242*, 2022. 2

- [29] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasempour, 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. [3](#)
- [30] Chitwan Saharia, Jonathan Ho, William Chan, Tim Salimans, David J Fleet, and Mohammad Norouzi. Image super-resolution via iterative refinement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. [3](#)
- [31] 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](#), [3](#)
- [32] 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](#)
- [33] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020. [2](#), [3](#)
- [34] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020. [2](#)
- [35] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012. [7](#), [1](#), [4](#)
- [36] Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017. [3](#), [1](#)
- [37] Chenfei Wu, Jian Liang, Lei Ji, Fan Yang, Yuejian Fang, Dixin Jiang, and Nan Duan. Nüwa: Visual synthesis pre-training for neural visual world creation. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVI*, pages 720–736. Springer, 2022. [3](#)
- [38] Jay Zhangjie Wu, Yixiao Ge, Xintao Wang, Weixian Lei, Yuchao Gu, 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. *arXiv preprint arXiv:2212.11565*, 2022. [2](#), [6](#), [7](#), [8](#), [3](#), [4](#)
- [39] Jay Zhangjie Wu, Xiuyu Li, Difei Gao, Zhen Dong, Jinbin Bai, Aishani Singh, Xiaoyu Xiang, Youzeng Li, Zuwei Huang, Yuanxi Sun, et al. Cvpr 2023 text guided video editing competition. *arXiv preprint arXiv:2310.16003*, 2023. [6](#)

# Neutral Editing Framework for Diffusion-based Video Editing

## Supplementary Material

### A. Broader Impacts and Ethic Statements

Visual generative models raise several ethical concerns such as illegal counterfeit content, potential invasion of privacy, and fairness issues. Our work also relies on the underlying framework of these generative models, making it vulnerable to these concerns. Therefore, effectively addressing these concerns is required, where various regulations should be prepared including technical safeguards. Crucially, researchers should take responsibility for these concerns and actively make an effort to build technical safeguards. Therefore, to mitigate potential concerns and hold transparency, we will release our source code including specifications of models and data that we employed under a license encouraging ethical and legal usage. We also consider introducing further regulations such as learning-based digital forensics and digital watermarking. Collectively, these measures aim to navigate the ethical landscape of visual generative models, fostering their responsible and beneficial use.

### B. Limitation and Future work

The editing systems seem to be susceptible to unintended bias in modifying required attributes. For instance, Figure 14 shows the failure case of our method. When modifying specific attributes (e.g., motion of riding snowboard) of an object in a video, the scene in the video is also changed into a context (e.g. snow) primarily associated with the desired attributes. We define this issue as editing bias and our future work is to mitigate the editing bias. Although NeuEdit is also successfully applicable in image non-rigid editing including still-pose editing (*i.e.* Figure 10), in a video domain, it was also challenging to edit a motion of a moving object to be still. As the temporal attention in the video diffusion model performs to preserve temporal consistency, this consistency is reflected in the object to be edited, such that a still pose is made, but follows the movements of the object.

### C. Further details and more evaluations

We present further details of our experiments including implementations, evaluations, and results.

#### C.1. Implementation details

For video encoding, we utilize VQ-VAE [36], which provides patch-wise features of each frame, and for text encoding the CLIP model (ViT-L/14) [25] is employed. In the visual neutralizing, we applied bicubic interpolation to restore the original scale of frames from each  $i$ -th frame editing factor score  $\mathbf{z}_V^i$ . In the case of the cross-attention module, the

cross-attention weights of the first up-block attention layer in Stable Diffusion U-Net [27] is utilized, where the attention map with the size of  $16 \times 16$  is constructed. Empirically, other mid-block and up-block layers can also properly work for designing the visual editing factor. However, the down-block layers were not effective due to insufficient early multimodal interactions between text and image.

For the details of neutral video, we set visual editing factor scores below 0.2 uniformly to zero. This approach improves the effectiveness of editing in the targeted region by establishing a clear boundary between the visual regions associated with the editing factor and those that are not.

#### C.2. Evaluation details

**Fidelity evaluation.** In order to measure fidelity to input video, as shown in Figure 8, we applied the identical zero mask to the edited area in the input video and output video. Applying a mask to the area to be edited allows us to measure the similarity and commonality between the input video and the output video in terms of preservation of unedited content, producing the score of PSNR, LPIPS, and SSIM. In the early study, we attempted several automatic detectors such as segmentation-based detectors [17] to identify areas to be edited in input and output video, but the specifications by humans were the most accurate.

**Human evaluation.** Human evaluation is performed to measure the preference for edited results according to a given target prompt. We conducted a survey about discrete selection between the outcomes of current editing systems and those generated using the NeuEdit framework. A survey involving 36 participants was conducted, incorporating diverse academic backgrounds (e.g. engineering, literature, art) and including those who speak English as their native language and those who do not.

#### C.3. Evaluations on videos of different domains

**UCF101** Collected from YouTube, UCF101 [35] dataset is designed for action recognition, comprising 101 action categories. To measure the video editing performances according to the editing model with NeuEdit framework and without the framework, We selected 83 videos from the dataset. Overall, performances of all models (*i.e.* TAV: Tune-A-Video, FateZero, and Video-P2P) using NeuEdit framework are enhanced in terms of textual alignment, fidelity, and consistency. Furthermore, the tuning-editing models (*i.e.* TAV, Video-P2P) of NeuEdit framework exhibited a significant enhancement in fidelity, similar to

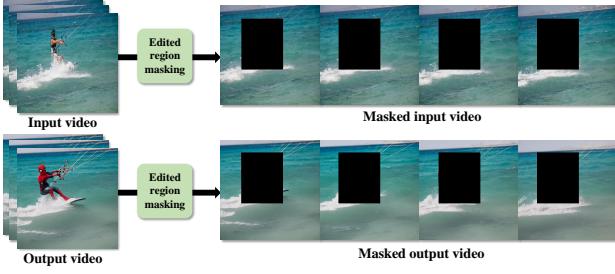


Figure 8. Illustration of masked input and output video for measuring fidelity metrics. Identical masking is applied between the input video and the output edited video.

the DAVIS dataset, demonstrating general effectiveness in videos.

## D. Proof for the closed form of KL divergence in reverse diffusion process

The reverse process of denoising diffusion probabilistic models is to approximate  $q(x_{t-1}|x_t)$  using parameterized Gaussian transitions  $p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_\theta(x_t, t))$ . Considering whole  $T$  step parameterized transitions, these are sequentially constructed as given below:

$$p_\theta(X) = p_\theta(x_T) \prod_{t=1}^T p_\theta(x_{t-1}|x_t), \quad (10)$$

where we take  $X = x_{0:T}$  and it starts at normal distribution  $p(x_T) = \mathcal{N}(x_T; 0, I)$ . To optimize the  $p_\theta(X)$ , training objective is to maximize log-likelihood  $\log(p_\theta(X))$ , where we can also apply variational inference by maximizing the variational lower bound  $-L_{VLB}$  as given below:

$$-L_{VLB} = \log p_\theta(X) - D_{KL}(q(Z|X)||p_\theta(Z|X)) \leq \log p_\theta(X), \quad (11)$$

where  $D_{KL}$  is the Kullback-Leibler divergence (KL divergence) and the  $Z$  is latent variable by reparametrization trick used in the variational auto-encoder. The  $q$  can be any distribution that we can address with ease. We leverage this inequality condition as  $-\log p_\theta(X) \leq L_{VLB}$ . The  $L_{VLB}$  can be expanded out as  $L_{VLB} = L_T + L_{T-1} + \dots + L_0$ , where they are defined with  $1 \leq t \leq T$  as given below:

$$\begin{aligned} L_T &= D_{KL}(q(x_T|x_0)||p_\theta(x_T)), \\ L_t &= D_{KL}(q(x_t|x_{t+1}, x_0)||p_\theta(x_t|x_{t+1})), \\ L_0 &= -\log p_\theta(x_0|x_1). \end{aligned} \quad (12)$$

Therefore the terms about  $L_t$  make the closed form of KL divergence under step  $t$  with a range of  $0 \leq t \leq T$ .

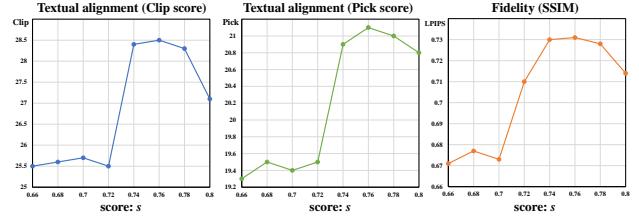


Figure 9. Sensitivity analysis of the score  $s$  for selecting editing factor for factor swapping in terms of textual alignment (i.e. CLIP score, Pick score) and fidelity (i.e. SSIM).

## E. DDIM sampling and DDIM inversion

To accelerate the reverse process of DDPM, denoising diffusion implicit model (DDIM) [33] is proposed, it samples latent features with a small number of denoising steps as:

$$z_{t-1} = \sqrt{\frac{\alpha_{t-1}}{\alpha_t}} z_t + \left( \sqrt{\frac{1}{\alpha_{t-1}} - 1} - \sqrt{\frac{1}{\alpha_t} - 1} \right) \epsilon \quad (13)$$

We can also reverse this process to make latent noise again, which gives corresponding latent features as below:

$$z_{t+1} = \sqrt{\frac{\alpha_{t+1}}{\alpha_t}} z_t + \left( \sqrt{\frac{1}{\alpha_{t+1}} - 1} - \sqrt{\frac{1}{\alpha_t} - 1} \right) \epsilon, \quad (14)$$

where it is referred to as DDIM inversion process, which maintains higher fidelity to the input than just initially starting from Gaussian noise.

## F. Studies of textual semantic disentanglement

### F.1. Sensitivity analysis of factor swapping

Factor swapping involves the binary deletion of a word according to a decision on the editing factor. Hence, the proper selection of editing factors contributes to an efficient neutral prompt. To explore this aspect further, we aim to investigate the optimal operational range of factor swapping by leveraging the threshold score  $s$  for deciding the editing factor. Figure 9 depicts the editing performance variations with changes in the score  $s$ , where the  $0.74 < s < 0.78$  presents the optimal range for effectively working as a neutral prompt with factor swapping.

### F.2. Ablation studies about factor deforming

We introduce further empirical methods performed for textual semantic disentanglement. In the main paper, we employ factor deforming  $\mathbf{w}_n = \mathbf{z}_T \circ h(\mathbf{w}) + (1 - \mathbf{z}_T) \circ \mathbf{w}$  with the deformable operation of down-scaling as below:

$$h(\mathbf{w}) = \alpha \mathbf{w} \in \mathbb{R}^{M \times d}, \quad (15)$$

	Textual Alignment		Fidelity to Input Video		Frame Consistency		Human	
	CLIP* $\uparrow$	PickScore $\uparrow$	PSNR $\uparrow$	LPIPS $\downarrow$	SSIM $\uparrow$	CLIP $^\dagger$ $\uparrow$	FVD $\downarrow$	Preference $\uparrow$
TAV [38]	22.3 / 26.7	19.2 / 19.7	12.7 / 13.8	0.1911 / 0.2031	0.573 / 0.609	0.913 / 0.937	3537 / 3441	0.15
TAV + NeuEdit	26.8 / 27.9	20.1 / 20.6	18.2 / 17.4	0.1532 / 0.1509	0.658 / 0.669	0.946 / 0.952	3321 / 3202	0.85
FateZero [23]	21.0 / 25.7	19.1 / 19.6	13.1 / 13.2	0.1749 / 0.1836	0.583 / 0.598	0.936 / 0.941	3372 / 3151	0.30
FateZero + NeuEdit	26.7 / 27.9	20.0 / 20.8	15.9 / 16.9	0.1723 / 0.1811	0.589 / 0.601	0.947 / 0.946	3252 / 3123	0.70
Video-P2P [19]	22.3 / 26.8	19.3 / 19.6	14.2 / 14.9	0.1821 / 0.1923	0.598 / 0.623	0.942 / 0.939	3282 / 3142	0.31
Video-P2P + NeuEdit	<b>27.1 / 28.4</b>	<b>20.5 / 21.1</b>	<b>18.4 / 18.9</b>	<b>0.1312 / 0.1413</b>	<b>0.676 / 0.681</b>	<b>0.948 / 0.954</b>	<b>3182 / 3001</b>	0.69

Table 2. Evaluations about edited videos based on UCF101 in terms of non-rigid/rigid type editing corresponding to textual alignment, fidelity to input video, frame consistency, and human preference. CLIP\*: text-video clip score, CLIP $^\dagger$ : image-image clip score.

where the  $0 \leq \alpha < 1$  is the down-scaler. Our studies also consider other deformable operations defined as (1) deformable swapping and (2) factor blurring. The detailed explanations are presented in the following.

**Deformable swapping** The deformable swapping is an extended version of factor swapping. Factor swapping involves transforming all words identified as editing factors into a unified token as a dummy token <DMY>. The limitation of this approach is that all modified dummy tokens become indistinguishable from one another. Therefore, we integrated the format of factor deforming with the factor swapping and adaptively changed the magnitudes of the dummy token feature according to the editing factor score as given below:

$$\mathbf{w}_n = \mathbf{z}_{\mathcal{T}} \circ \mathbf{w}_n^{\text{swp}} + (1 - \mathbf{z}_{\mathcal{T}}) \circ \mathbf{w}, \quad (16)$$

where  $\mathbf{w}_n^{\text{swp}}$  is text feature obtained by factor swapping. This format is available to impart distinguishable influence to dummy token features based on the editing factor score.

**Factor blurring** Another attempt is factor blurring. Similar to the factor deforming in the visual textual semantic disentanglement, we apply to blur the features related to editing factors. Hence, rather than employing feature downscaling, we adopt an alternative approach by introducing an additional  $d$ -dimensional noise feature to deform the target prompt features as given below

$$\mathbf{w}_n = \mathbf{z}_{\mathcal{T}} \circ (\mathbf{w} + \epsilon) + (1 - \mathbf{z}_{\mathcal{T}}) \circ \mathbf{w}, \quad (17)$$

where  $\epsilon \sim \mathcal{N}(0, 1)$  is the noise features added to target prompt features. It deforms the prompt feature corresponding to editing factors.

Table 3 presents ablation studies in terms of factor deforming used for textual semantic disentangling. The first section is the original factor deforming with feature-down scaling in the main paper. The second section is the factor blurring method, which was less effective

	Alignment	Fidelity	Consistency
	CLIP* $\uparrow$	SSIM $\uparrow$	CLIP $^\dagger$ $\uparrow$
factor deforming (M)	28.6	0.730	0.969
factor blurring (A)	27.7	0.697	0.948
factor swapping (M)	27.5	0.707	0.951
deformable swapping (A)	28.1	0.711	0.961

Table 3. Ablation study of different methods of factor deforming for textual semantic disentanglement. M: method in Main paper, A: method is Appendix

than other methods. We consider that although the blurring method is effective in visual semantic disentanglement, in the case of words, it seems difficult to obtain meaningful information from the blurred features that the network understands due to words' discrete characteristics. The third and fourth sections are factor swapping. Building distinguishable features among dummy token features enhances the editing performances in deformable factor swapping. This demonstrates that the editing factor scores properly contain information about how effective each factor is in performing the editing.

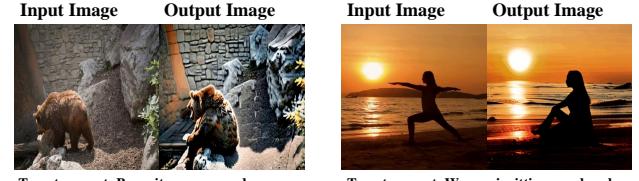


Figure 10. Application of NeuEdit into image editing about non-rigid editing (e.g. pose variations).

## G. Further qualitative results

**Application to image editing** The NeuEdit framework can be structurally applied to any diffusion-based editing system, such that we extend our work in image editing.

Figure 10 shows image editing results under the NeuEdit framework. It is notable that the non-rigid editing is also successfully applied based on the input image.

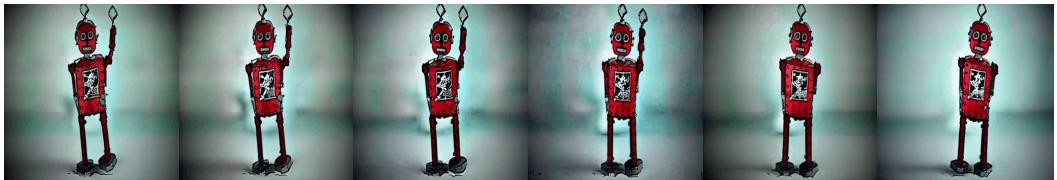
**Further qualitative results** In the following, we showcase our qualitative editing results in terms of non-rigid and rigid editing. All the videos in our experiments are sourced from publicly available sources in [7, 22, 35, 38].

**Input video**



Target prompt: "*A toy raise its hand and down*s"

[Motion change]



**Input video**



Target Text: "*A black swan is walking on the road*"

[Motion change]



**Input video**



Target Text: "*A woman dances ballet while walking on the road*"

[Motion change]



Figure 11. Illustration non-rigid editing including motion change.

**Input video**



Target Text: “*LEGO astronaut is walking on the moon*”

[Object overlay]



Target Text: “*Cookie astronaut is walking on the moon*”

[Object overlay]



**Input video**



Target Text: “*A tiger in the sunlight*”

[Object overlay]



**Input video**



Target Text: “*A woman does kettlebell exercises in front of an orange sky*”

[Object overlay]



Figure 12. Illustration of rigid editing about object overlay.

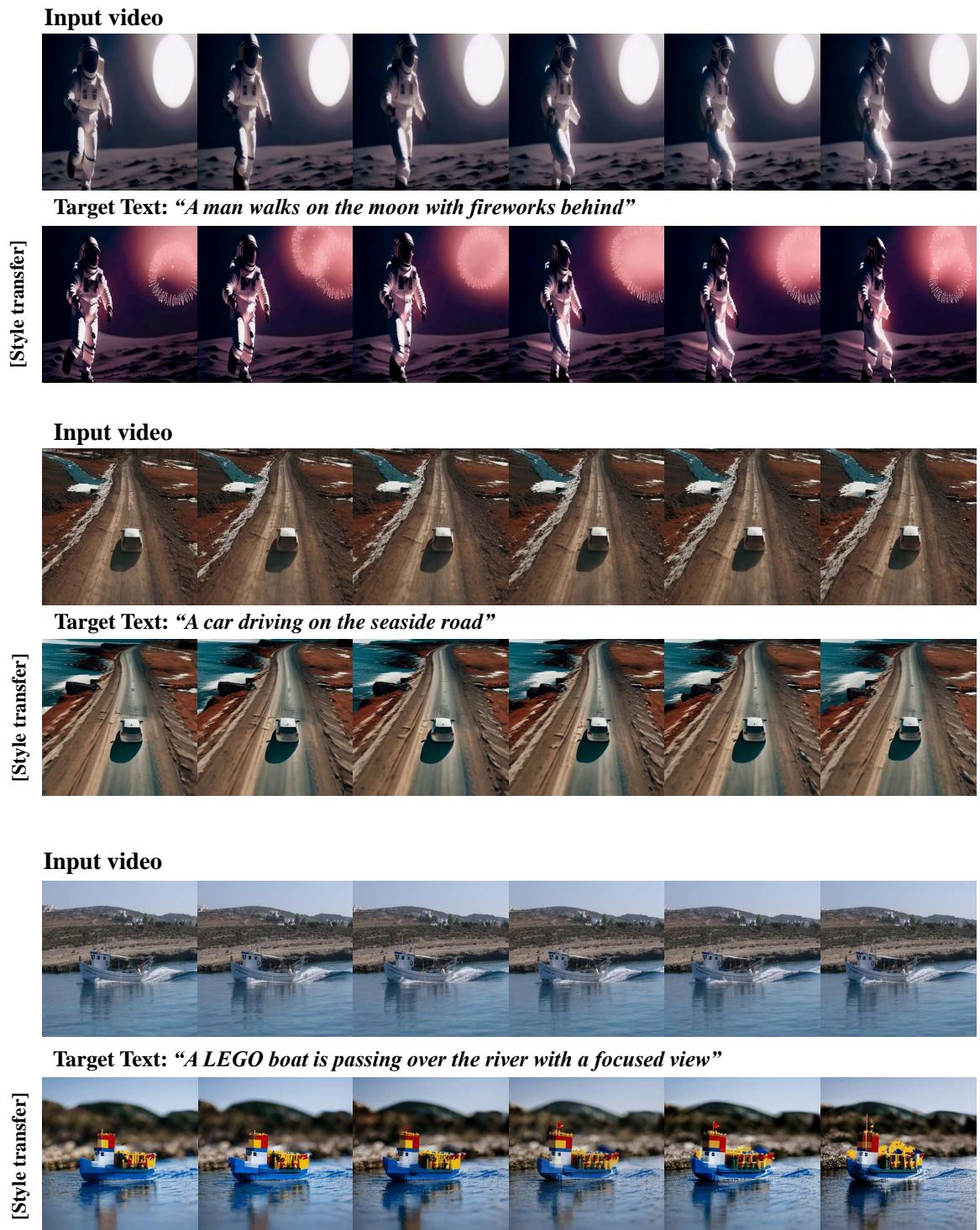


Figure 13. Illustration of rigid editing about style transfer.

**Input video**



**Target prompt: "Astronaut is riding a snowboard while raising hands"**



Figure 14. Failure case by editing bias. The scene is unintentionally changed into a snow background correlated with riding a snowboard.