VidStyleODE: Disentangled Video Editing via StyleGAN and NeuralODEs

Moayed Haji Ali * Andrew Bond *
Koç University

Tolga Birdal
Imperial College London

Duygu Ceylan
Adobe Research

Levent Karacan
Iskenderun Teknik University

Erkut Erdem
Hacettepe University

Aykut Erdem
Koç University

Figure 1. **VidStyleODE** provides a spatiotemporal video representation in which motion and content info are disentangled, making it ideal for: (a) animating images, (b) consistent video appearance manipulation based on text, (c) body part motion transfer ([blue] boxes) from a co-driving video while preserving remaining driving video dynamics ([orange] boxes) intact, (d) temporal interpolation, and (e) extrapolation.

Abstract

*We propose **VidStyleODE** , a spatiotemporally continuous disentangled **vid**eo representation based upon **Style**GAN and Neural-**ODE**s. Effective traversal of the latent space learned by Generative Adversarial Networks (GANs) has been the basis for recent breakthroughs in image editing. However, the applicability of such advancements to the video domain has been hindered by the difficulty of representing and controlling videos in the latent space of GANs. In particular, videos are composed of content (i.e., appearance) and complex motion components that require a special mechanism to disentangle and control. To achieve this, VidStyleODE encodes the video content in a pre-trained StyleGAN $\mathcal{W}_+$ space and benefits from a latent ODE component to summarize the spatiotemporal dynamics of the input video. Our novel continuous video generation process then combines the two to generate high-quality and temporally consistent videos with varying frame rates. We show that our proposed method enables a variety of applications on real videos: text-guided appearance manipulation, motion manipulation, image animation, and video interpolation and extrapolation. Project website: https://cyberiada.github.io/VidStyleODE*

Introduction

Semantic image editing is revolutionizing the visual design industry by enabling users to perform accurate edits in a fast and intuitive manner. Arguably, this is achieved by carrying out the *image manipulation* process with the guidance of a variety of inputs, including text [4,26,34,56], audio [25,27],

or scene graphs [8]. Meanwhile, the visual characteristics of real scenes are constantly changing over time due to various sources of motion, such as articulation, deformation, or movement of the observer. Hence, it is desirable to adapt the capabilities of image editing to videos. Yet, training generative models for high-res videos is challenging due to the lack of large-scale, high-res video datasets and the limited capacity of current generative models (*e.g*. GANs) to process complex domains. This is why the recent attempts [35, 58] are limited to low-res videos. Approaches that treat videos as a discrete sequence of frames and utilize image-based methods (*e.g*. [20, 49, 60]) also suffer from a lack of temporal coherency and cross-sequence generalization.

To overcome these limitations, we set out to learn **spatio-temporal** video representations suitable for both generation and manipulation with the aim of providing several desirable properties. First, representations should **express high-res** videos accurately, even when trained on low-scale low-res datasets. Second, representations should be robust to **irregular** motion patterns such as velocity variations or local differences in dynamics, *i.e*. deformations of articulated objects. Third, it should naturally allow for **control and manipulation of appearance and motion**, where manipulating one does not harm the other *e.g*. manipulating motion should not affect the face identity. We further desire to learn these representations **efficiently** on extremely sparse videos (3-5 frames) of arbitrary lengths. To this end, we introduce VidStyleODE , a principled approach that learns disentangled, spatio-temporal, and continuous motion-content representations, which possesses all the above attractive properties.

Similar to [2, 20, 49, 60], we regard an input video as a composition of a fixed appearance, often referred to as video *content*, with a motion component capturing the underlying *dynamics*. Respecting the nature of *editing*, we propose to model latent *changes* (*residuals*) required for taking the source image or video towards a target video, specified by an external *style* input *and/or* co-driving videos. For this purpose, VidStyleODE first disentangles the content and dynamics of the input video. We model content as a global code in the $\mathcal{W}_+$ space of a *pre-trained* StyleGAN generator and regard dynamics as a continuous signal encoded by a latent ordinary differential equation (ODE) [3, 7, 40], ensuring temporal smoothness in the latent space. VidStyleODE then explains all the video frames in the latent space as *offsets* from the single global code summarizing the video content. These offsets are computed by solving the latent ODE until the desired timestamp, followed by subsequent self- and cross-attention operations interacting with the dynamics, content, and style code specified by the textual guidance. To achieve effective training, we omit adversarial training, commonly used in the literature, and instead introduce a novel temporal consistency loss (Sec.) based on CLIP [36]. We show that it surpasses conventional consistency objectives and exhibits higher training stability.

Overall, our contributions are:

1. We build a novel framework, VidStyleODE , disentangling content, style, and motion representations using StyleGAN2 and latent ODEs.
2. By using latent directions with respect to a global latent code instead of per-frame codes, VidStyleODE enables external conditioning, such as text, leading to a simpler and more interpretable approach to manipulating videos.
3. We introduce a new *non-adversarial* video consistency loss that outperforms prior consistency losses, which mostly employ conv3D features at a lower training cost.
4. We demonstrate that despite being trained on low-res videos, our representation permits a wide range of applications on high-resolution videos, including appearance manipulation, motion transfer, image animation, video interpolation, and extrapolation (*cf*. Fig. 1).

Related Work

GANs. Since their introduction, GANs [14, 23] have achieved great success in synthesizing photorealistic images. Recent methods [38, 39, 46] obtain the latent codes of real images in StyleGAN's latent space and modify them to achieve guided manipulation considering the task at hand [34, 55, 56]. Despite their ability to generate high-res images, GANs are deemed challenging to train on complex distributions such as full-body images [11, 12] or videos. Earlier attempts [29, 41, 44, 47] modified GAN architecture to effectively synthesize videos based on sampled content and motion codes. Most notably, StyleGAN-V [44] recently modified StyleGAN2 to synthesize long videos while requiring a similar training cost. However, these methods are bounded by the resolution of the training data and are impractical for complex domains and motion patterns. Our work leverages the expressiveness of a pre-trained StyleGAN2 generator to encode input videos as trajectories in the latent space and extends image-based editing strategies to enable consistent text-guided video appearance manipulation.

Video generation. Recent works focused on using a pre-trained image generation as a video generation backbone. MoCoGAN-HD [45] and StyleVideoGAN [10] synthesize videos from an autoregressively sampled sequence of latent codes. InMoDeGAN [53] decomposes the latent space into semantic linear sub-spaces to form a motion dictionary. Other methods [1, 35] decompose pose from identity in the latent space of pre-trained StyleGAN3, enabling talking-head animation from a driving video. StyleHeat [61] warps intermediate pre-trained StyleGAN2 features with predicted flow fields for video/audio-driven reenactment. [43, 54] animate images based on a driving video following optical-flow-based methods in the pixel [43] or latent [54] space. Despite their success, these methods are limited to unconditional video synthesis [10, 45], are restricted to a single domain [1, 35, 61], designed for a single purpose [1, 35, 43, 54, 61],

and/or incapable to effectively generate high-res videos [44]. We present a domain-invariant framework to learn disentangled representations of content and motion, enabling a range of applications on high-res videos. In contrast to all of the aforementioned methods except MRAA [43], we also do not use adversarial training. With the motivation of handling irregularly sampled frames and continuous-time video generation, some previous works also incorporated latent ODEs [7] for unconditional video generation [30], future prediction from single frame [19], or modeling uncertainty in videos [62]. Despite being limited to low-res videos, these methods showed the potential of latent ODEs in video interpolation and extrapolation. VidStyleODE further extends them by showing the effectiveness of latent ODEs in high-res video interpolation and extrapolation.

Semantic video manipulation. Applying image-level editing to individual video frames often leads to temporal incoherence. To alleviate this problem, Latent Transformer [60] uses a shared latent mapper to the latent codes of the input frames in a pre-trained StyleGAN2 latent space. Alaluf et al. [2] propose a consistent video inversion/editing pipeline for StyleGAN3. STIT [49] fine-tunes a StyleGAN2 generator on the input video and moves along a single latent direction to realize the target edit. These methods still fail to achieve temporally consistent manipulation due to the entanglement between appearance and video dynamics in the StyleGAN space, defying their presumption of temporal independence between video frames. As a remedy, DiCoMoGAN [20] encodes video dynamics with a neural ODE [7], and learns a generator that manipulates input frames based on the learned motion dynamics and a target textual description. StyleGAN-V [44] enables video manipulation by projecting real videos onto a learned content and motion space, enabling appearance manipulation via the modification of the content code following image-based methods [34, 55]. Instead of directly modifying content code, our model achieves guided manipulation by discovering spatio-temporal latent directions conditioned on the target description and the video dynamics. This allows for greater flexibility regarding the appearance-motion entanglement of StyleGAN space. VidStyleODE also encodes video dynamics with a latent ODE that encourages a smooth latent trajectory, thus enhancing temporal consistency.

Method

We consider an input video $\mathcal{V} = \{\mathbf{X}_i \in \mathbb{R}^{M\times N\times 3}\}_{i=1}^{K}$ consisting of K RGB frames along with an associated textual description $\mathcal{D}_{\text{SRC}}$. Our goal is to explain $\mathcal{V}$ by learning an explicitly manipulable *continuous representation* conditioned on an external *style* input. As manipulation is inherently related to making *changes* [55], VidStyleODE achieves this goal via a deep neural architecture, modeling the changes through disentangled *content*[1], *style*[2] and *dynamics*[3]. To this end, VidStyleODE first uses a pre-trained spacetime encoder $f_C : \mathcal{V} \rightarrow \mathbf{z}_C$ to summarize the information content of the input video frames or individual images as a *global latent code*. Our key idea is to explain individual video frames with respect to the global code as *translations* along the latent dimensions of a pre-trained high-res image generator $G(\cdot)$:

$$\overline{\mathbf{X}}_t = G\left(\mathbf{z}_{new} = \mathbf{z}_C + \Delta_{\mathbf{z}t}\right) \tag{1}$$

To find these *latent directions* $\Delta_{\mathbf{z}t}$ that entangle dynamics and style, we (i) continuously model latent representation of dynamics $\mathbf{z}_{dt}$, which can be queried at arbitrary timesteps; (ii) learn to predict these directions by interacting with the global code $\mathbf{z}_C$ and the predicted dynamics $\mathbf{z}_{dt}$, conditioned on the target style $\mathbf{z}_S$, while preserving the content. There are multiple ways to get $\mathbf{z}_S$, but in this work, we choose to extract it based on target and source textual descriptions $(\mathcal{D}_{\text{SRC}}, \mathcal{D}_{\text{TGT}})$. We first describe the method design for each of these components, depicted in Fig. 2, followed by implementation and architectural details in **??** .

Spatiotemporal encoding f_C. To encode the entire video into a global code, we seek a *permutation-invariant* representation of the input video, factoring out the temporal information. To this end, we first project all the frames in $\mathcal{V}$ onto the $\mathcal{W}_+$ space of StyleGAN2 [23] by using an *inversion* [57] to obtain a set of *local* latent codes $\mathbf{Z} := \{\mathbf{z}_i^l \in \mathcal{W}_+\}_{i=1}^{K}$. We then apply a symmetric pooling function to obtain the order-free global video content code: $\mathbf{z}_C = \mathbb{E}\left[\mathbf{Z}\right]$.

Continuous dynamics representation. Inspired by [31, 37], to model the spatiotemporal input, *i.e.*, to compute representations for unobserved timesteps at arbitrary spacetime resolutions, we opt for learning a latent subspace $\mathbf{z}_{d0} \in \mathbb{R}^D$, that is used to initialize an autonomous latent ODE $\frac{d\mathbf{z}_{dt}}{dt} = f_\theta(\mathbf{z}_{dt})$, which can be advected in the latent space rather than physical space:

$$\mathbf{z}_{dT} = \phi_T(\mathbf{z}_{d0}) = \mathbf{z}_{d0} + \int_0^T f_\theta(\mathbf{z}_{dt}, t)\, dt \tag{2}$$

where θ denotes the learnable parameters of the model f_θ. This (1) enables *learning* a space best suited to modeling the dynamics of the observed data and (2) improves scalability due to the fixed feature size. Due to the time-independence of f_θ, advecting $\mathbf{z}_{dt=0}$ forward in time by solving this ODE until $t = T \geq 1$ yields a representation that can explain latent variations in video content. To learn the initial code $\mathbf{z}_{d0}$, we encode each frame individually by a *spatial encoder* $f_D : \mathbf{X}_i \rightarrow \mathbb{R}^{m_d\times n_d\times 64}$. Resulting tensors are fed into a ConvGRU: $\mathbb{R}^{m_d\times n_d\times 64\times K} \rightarrow \mathbb{R}^{m_{\text{ode}}\times n_{\text{ode}}\times 512}$ [3, 31] in reverse order so that the final code seen by the model corresponds to the first frame.

[1] set of attributes fixed along the temporal dimension [20, 45, 47]
[2] attributes of interest subject to change
[3] an intrinsic force producing change

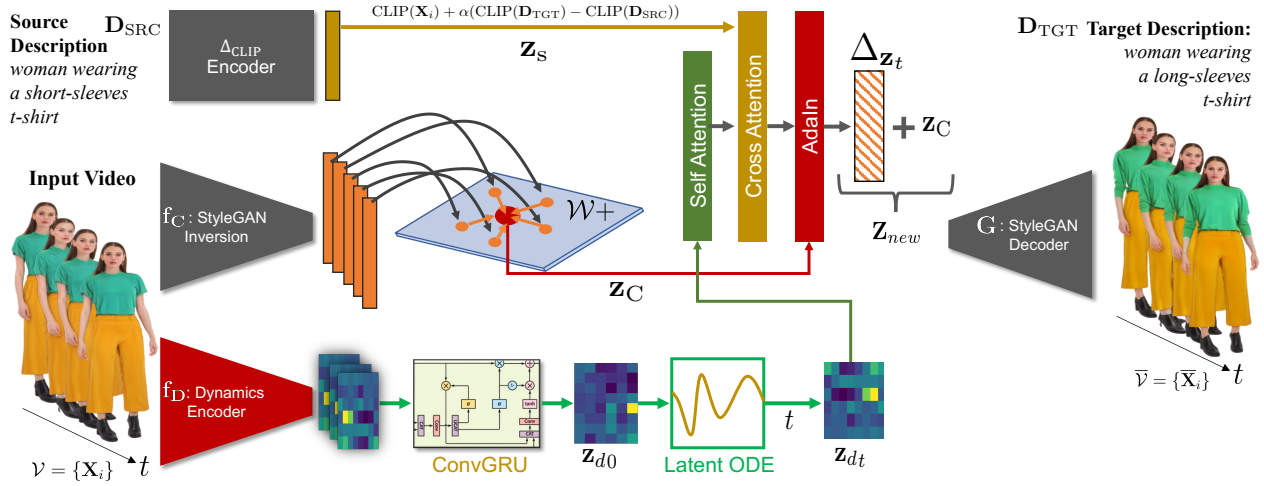

Figure 2. **VidStyleODE overview**. We encode video dynamics and process them using a ConvGRU layer to obtain a dynamic latent representation $\mathbf{Z}_{d0}$ used to initialize a latent ODE of the motion (bottom). We also encode the video in $\mathcal{W}_+$ space to obtain a global latent code Z_C (middle). We combine the two with an external style cue through an attention mechanism to condition the AdaIN layer that predicts the directions to the latent codes of the frames in the target video (top). Modules in **gray** are *pre-trained* and *frozen* during training.

Conditional generative model f_G. To synthesize high-quality video frames that adhere to the target style $\mathbf{z}_S$, VidStyleODE generatively models the desired output at time t as an explicit function of content, dynamics and style:

$$\overline{\mathbf{X}}_t = G(\mathbf{z}_t), \quad \mathbf{z}_t = f_G(\mathbf{z}_c, \mathbf{z}_d \,|\, \mathbf{z}_S) = \mathbf{z}_C + \Delta_{\mathbf{z}t}, \quad (3)$$

where the *latent direction* $\Delta_{\mathbf{z}t}$ depicts the residual required to realize the desired edits and is computed by a series of self-attention (SA) [51], cross-attention (CA) [51] and adaptive instance normalization (AdaIN) [16] operators:

$$\Delta_{\mathbf{z}t} = \text{AdaIn}\left(\text{CA}(\text{SA}(\mathbf{z}_{dt}), \mathbf{z}_S), \mathbf{z}_C\right) \quad (4)$$

Modeling the *change* in this manner rather than the target latents themselves is significantly less complex and allows for manipulating the given video in relation to its global code. As such, and as we demonstrate experimentally, it offers significant advantages of fidelity and manipulation-ability. We implement $G(\cdot)$ as a pre-trained StyleGAN2 generator.

Obtaining the text-driven style $\mathbf{z}_S$. We model the *change* in source and target descriptions as a *style direction* $\Delta_{\mathbf{z}}^{\text{Style}} = \text{CLIP}(\mathcal{D}_{\text{TGT}}) - \text{CLIP}(\mathcal{D}_{\text{SRC}})$ in the CLIP latent space [34, 36]. We then move towards this direction in the CLIP space to obtain the text conditioning code:

$$\mathbf{z}_S = \text{CLIP}(\mathbf{X}_i) + \alpha \Delta_{\mathbf{z}}^{\text{Style}} \quad (5)$$

where α is a user-controllable parameter determining the scale of the manipulation.

Training and Network Architectures

We train VidStyleODE by minimizing a multi-task loss $\mathcal{L}$ over the text-video pairs to find the best parameters of dynamics encoder f_D as well as f_G while keeping the content encoder and the image generator frozen:

$$\mathcal{L} = \lambda_C \mathcal{L}_C + \lambda_A \mathcal{L}_A + \lambda_S \mathcal{L}_S + \lambda_D \mathcal{L}_D + \lambda_L \mathcal{L}_L \quad (6)$$

where λ_* depicts the corresponding regularization coefficients. We next detail each of these terms, which are consistency, appearance reconstruction, structure reconstruction, CLIP directional loss, and latent direction regularization.

CLIP consistency loss. DietNeRF [18] shows that the CLIP [36] image similarity score is more sensitive to changes in appearance, compared to those caused by varying viewpoints. This led the authors to propose a new consistency loss as the pair-wise CLIP dissimilarity between images rendered from different viewpoints in order to guide the reconstruction of 3D NeRF representation. We observe that CLIP is also more sensitive to changes in appearance than to changes in dynamics Thus, we propose to replace the expensive temporal discriminator used in the literature [45, 47, 52], with a CLIP consistency loss along the temporal dimension. Specifically, we sample N_C frames from the generated video and minimize the pair-wise dissimilarity between them.

$$\mathcal{L}_C(\mathcal{V}) = \sum_{i=1}^{N_C} \sum_{j \geq i}^{N_C} 1 - (\text{CLIP}_I(\overline{\mathbf{X}}_i)^T \text{CLIP}_I(\overline{\mathbf{X}}_j)) \quad (7)$$

where $\overline{\mathbf{X}}_i$ is the i_{th} sampled frame from the generated video, and CLIP_I is the CLIP image encoder.

Appearance and structure reconstruction loss. To learn the video dynamics, previous work [1, 20, 35, 61] commonly used a VGG perceptual loss and L2 loss, which reconstructs both the structure and appearance of the input video. This inherently requires the image generator to be fine-tuned on the

input video dataset. Considering that most available video datasets are of a low resolution and low diversity, fine-tuning the image generator on these datasets would greatly affect the model's capability to generate diverse and high-quality videos. Therefore, we propose to use a disentangled structure/appearance reconstruction loss to guide learning the dynamic representation. In particular, we use the Splicing-ViT [48] appearance loss to encourage the appearance of the generated video to match the appearance represented in the global code $\mathbf{z}_C$. Additionally, as motion dynamics are closely related to the change in structure [59], we use Splicing-ViT structural loss to encourage the dynamics of the generated video to follow the dynamics of the input video.

$$\mathcal{L}_A = \sum_{i=1}^{N} ||ViT_A(G(\mathbf{z}_C)) - ViT_A(G(\mathbf{z}_{t_i}))|| \quad (8)$$

$$\mathcal{L}_S = \sum_{i=1}^{N} ||ViT_S(\mathbf{X}_i) - ViT_S(G(\mathbf{z}_{t_i}))|| \quad (9)$$

where ViT_A, and ViT_S are the latent features in DINO-ViT [5] corresponding to appearance and structure, respectively, as described in [48]. This way, we can disentangle learning appearance and dynamic representation completely, enabling diverse high-res video generation via low-res video datasets.

CLIP video directional loss. Given source and target descriptions, and a reference image, [13] proposes to guide the appearance manipulation in the generated image by encouraging the change of the images in the CLIP space to be in the same direction as the change in descriptions. We adapted this loss to the video domain using:

$$\Delta_T = \text{CLIP}_T(T_{desc}) - \text{CLIP}_T(S_{desc}) \quad (10)$$

$$\Delta_V = \frac{\sum_1^N \text{CLIP}_I(\overline{\mathbf{X}}_i) - \text{CLIP}_I(G(\mathbf{z}_{t_i}))}{N}$$

$$\mathcal{L}_D = 1 - \Delta_V \Delta_T \,/\, |\Delta_V||\Delta_T|$$

where CLIP_T, and CLIP_I correspond to the CLIP text and image encoder, respectively, and N refers to the number of sampled frames from the generated video. During training, we sample three frames per video.

Latent direction loss. We regularize the norm of the latent directions $\Delta_{\mathbf{z}}$ to prevent the model from following directions with large magnitudes: $\mathcal{L}_L = \mathbb{E}[||\Delta_{\mathbf{z}t_i}||]_i$. We observed that this loss also helped in making the model converge faster.

Network architectures. We used a ResNet architecture adapted from [33] as our dynamic encoder f_D. Additionally, we used Vid-ODE ConvGRU network [32] to obtain the dynamic representation $\mathbf{z}_d$ before utilizing the Dopri5 [6] method to solve the first-order ODE. We apply self-attention and cross-attention over $\mathbf{z}_d$ by dividing the input tensor into patches and treating them as separate tokens, following [9]. Additionally, we used a pSp encoder to obtain $\mathbf{z}_i$, and a StyleGAN2 generator [23] for $G(\cdot)$, pre-trained on Stylish-Humans-HQ Dataset [12] for fashion video experiments, and on FFHQ [21] for face video experiments.

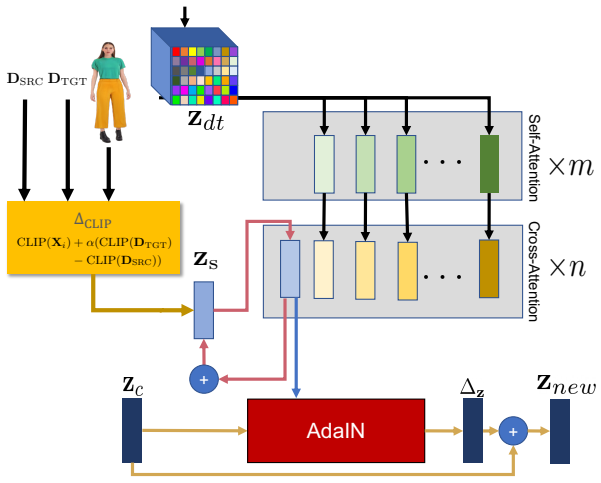

Figure 3. **Proposed attention scheme utilized in VidStyleODE.**

Training details. Thanks to our choice of modeling dynamics as a latent ODE, we are able to train on irregularly sampled frames. Specifically, for every training step, we sample k different frames from each input video and a target description from other videos in the batch. We use those to compute the aforementioned losses. Details about hyperparameters can be found in the supp. materials.

Experimental Analysis

Datasets and prepossessing.. We evaluated our method mainly on the recent dataset of Fashion Videos [20] composed of 3178 videos of fashion models and RAVDESS dataset [28], containing 2,452 videos of 24 different actors speaking with different facial expressions. We split each dataset randomly into 80% train and 20% test data. Moreover, we aligned their video frames following [12, 23], and downsampled the input videos during training to 128×96 for Fashion 128×128 for RAVDESS. Additionally, we annotated each actor in RAVDESS according to gender, hairstyle, hair color, and eye color, and procedurally generated target descriptions based on these attributes.

Evaluation metrics. To assess the performance of the models, we use the following metrics. *Frechet Video Distance* (FVD) [50] measures the difference in the distribution between ground truth (GT) videos and generated ones. *Inception Score* (IS) [42] and *Frechet Inception Distance* (FID) [15] measures the diversity and perceptual quality of the generated frames. *Manipulation Accuracy* quantifies the agreement of the edited video with the target text, relative to a GT video description. *Warping error* [24] measures the temporal appearance consistency. *Average key-point distance* (AKD) assesses the structural similarity between the generated and driving videos. *Average Euclidean distance* (AED) evaluates identity preservation in reconstructed videos.

Baselines. We compare our method against SOTA text-guided video manipulation and image animation approaches: Latent Transformer (LT) [60], DiCoMoGAN [20], STIT [49],

Figure 4. **Text-guided editing results**. VidStyleODE lets the users manipulate a frame based on a text prompt, and transfer manipulated attributes to other videos in a consistent way. Source frames are shown at the top left corner along with the target texts.

StyleGAN-V [44], and MRAA [43]. As LT requires separate training for each target attribute, we trained it to manipulate only the sleeve length on Fashion Videos and averaged its performance for RAVDESS on gender, hair, and eye color. Additionally, we trained DiCoMoGAN and StyleGAN-V on the face and fashion datasets using the same alignment process in our method. STIT fine-tunes the generator using PTI [39] for each input video, taking 10 minutes for a 1-minute video on NVIDIA RTX 2080, and further uses image-based manipulation methods. We employed StyleCLIP global directions. StyleGAN-V achieves text-guided manipulation by performing test-time optimization of projected latent codes with CLIP. We also considered HairCLIP [55] and StyleCLIP [34] as baselines for frame-by-frame manipulation of the video. Lastly, we train MRAA [43] and adapt StyleGAN-V code to evaluate same-identity and cross-identity image animation. (*cf*. supplementary materials).

Results

Semantic video editing. Our method allows for text-guided video editing by conditioning the prediction of the latent direction on the manipulation direction specified by the target and source descriptions. Fig. 4 shows that our method accurately manipulates the color, clothing style, and sleeve length in a temporally-consistent way on several sample video frames. VidStyleODE can also handle target descriptions that consider either single or multiple attributes without introducing artifacts. Fig. 13 compares our method against the state-of-the-art. As seen, LT [60] and the frame-level HairCLIP [55] fail to preserve temporal consistency, especially with respect to the identity. DiCoMoGAN [20] and STIT [49] perform poorly in applying meaningful and consistent manipulations. In particular, DiCoMoGAN fails to perform the necessary manipulations in the text-relevant parts such as the sleeves, and produces artifacts in the text-irrelevant parts. STIT applies the same latent direction to all of the video frames in the StyleGAN2 $\mathcal{W}_+$ space. We show that this is prohibitive, as the relative edits of the manipulated parts, such as the sleeves' length, change as the body moves.

Figure 5. **Qualitative comparison against the state-of-the-art**. VidStyleODE produces more realistic results than existing semantic video methods when changing sleeve length from short to long, with improved visual quality and manipulation accuracy. HairCLIP, a frame-level method, lacks temporal coherence.

These observations are also reflected in the results reported in Tab. 1. As LT cannot jointly manipulate multiple attributes with the same model, we consider a relatively simple setup where we only manipulate the length of the sleeves of the source garments for a fair comparison. STIT, which performs instance-level optimization, gives the best FVD scores, yet its manipulation accuracy is significantly inferior to ours. Although HairCLIP achieves the best accuracy metric, its performance is the worst in terms of (temporal) video quality as measured by FVD. Our VidStyleODE method achieves an FVD score close to STIT, and a manipulation accuracy close to HairCLIP. In general, it is the only method that produce smooth and temporally-consistent videos with high fidelity to the target attributes. It also preserves the identity of the person while making the target garment edits.

Fig. 6 shows further manipulation results on the RAVDESS dataset. We observe that existing models exhibit similar limitations observed in the Fashion Videos dataset but at a lower degree. We hypothesize that this is mainly due to StyleGAN2 learning a more disentangled and expressive

Method	Fashion Videos					RAVDESS				
	FVD ↓	IS ↑	FID ↓	Acc. ↑	W_{error} ↓	FVD ↓	IS ↑	FID ↓	Acc. ↑	W_{error} ↓
HairCLIP [55]	548.09	2.56	65.57	**0.92**	0.0152	218.70	1.33	31.47	0.83	0.01360
STIT [49]	**126.04**	3.08	33.24	0.72	0.0089	226.31	1.33	32.89	0.71	0.0088
LT [60]	262.17	3.08	39.06	0.24	0.0095	339.48	1.35	37.05	0.43	0.0192
DiCoMoGAN [20]	324.30	2.50	103.62	0.51	0.0151	**121.92**	**1.40**	**16.38**	0.38	0.0086
StyleGAN-V [44]	988.96	2.30	135.49	0.71	0.0384	487.91	1.28	66.89	**0.87**	0.0307
Ours	157.48	**3.25**	**26.28**	0.87	**0.0075**	273.10	1.33	34.92	0.83	**0.0076**

Table 1. **Quantitative comparison on the Fashion and RAVDESS datasets**. We report the performances using metrics for evaluating photorealism (FVD, IS, and FID), manipulation accuracy (Acc.), and temporal coherency (W_{error}). While the scores in **bold** highlight the best performance, the underlined ones show the second best. Overall, our VidStyleODE method is the only approach that gives photorealistic and temporally consistent results with accurate edits of the garment attributes.

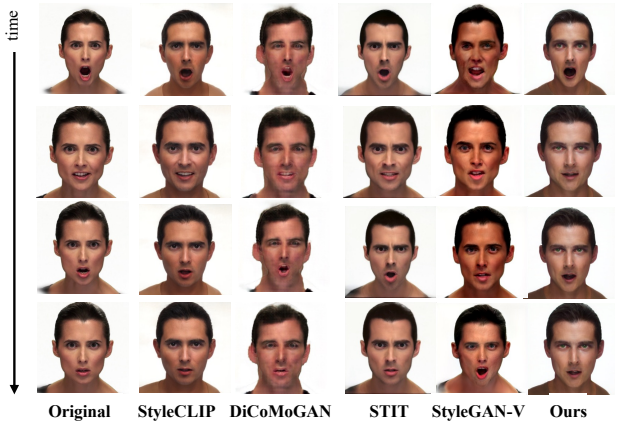

Figure 6. **Facial attribute manipulation**. Target Description: a photo of a man with *green eyes*. VidStyleODE gives a temporally consistent output when manipulating source face video, unlike other methods which show inconsistencies in hairline, nose, or identity, or fails to make the proper edits.

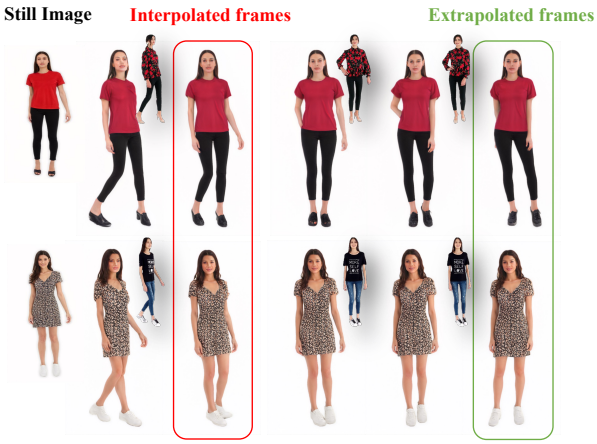

Figure 7. **Animating a still image**. Our method animates input images using motion dynamics from a driving video. With a learned continuous representation of motion dynamics via a latent ODE, it can also generate realistic frames via interpolation or extrapolation.

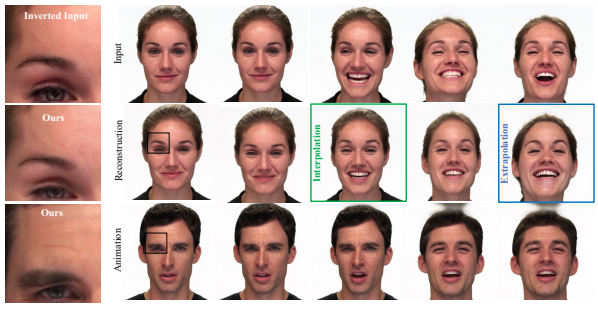

Figure 8. **High-res results on RAVDESS.** VidStyleODE maintains the perceptual quality of the pre-trained and frozen StyleGAN2 Generator (col. 1), while enabling temporal interpolation (col. 4) and extrapolation (col. 6), and image animation (last row).

latent space on a simple dataset containing face images.

In summary, we conclude that auto-encoder-based approaches such as [20] are able to faithfully reconstruct the text-irrelevant parts such as the face identity but lack the capability of performing meaningful manipulations, resulting in artifacts and unnatural-looking videos. StyleGAN2-based approaches [49, 55] achieve good semantic manipulation but lack the ability to keep a consistent appearance in the generated video. VidStyleODE benefits from a pre-trained StyleGAN2 generator to perform meaningful semantic manipulations while producing smooth and consistent videos.

Image animation and video interpolation/extrapolation. Our model is able to learn a disentangled representation of content and motion, allowing for animating the content extracted from a still image using the motion dynamics coming from a driving video. In Fig. 7 and Fig. 8, we show some sample results of this process. Since our framework is equipped with a latent ODE, we can use our method to perform interpolation between selected video frames. Moreover, we are able to extrapolate the motion dynamics to future timesteps not seen in the original driving video. Fig. 9 further shows the ability of our method in controlling the motion dynamics in a disentangled manner. As seen, we can obtain diverse animations of a given source image by transferring motion from different driving videos. Our method generates a consistent appearance for the person across different videos (Table 2).

Figure 9. **Diverse animation results achieved by VidStyleODE** . Each example shows a separate driving video (top-left corner) and the corresponding animations. Our method provides disentangled motion control while keeping the source content information intact.

Method	Fashion Videos			RAVDESS		
	$AKD_C \downarrow$	$AKD_S \downarrow$	$AED_S \downarrow$	$AKD_C \downarrow$	$AKD_S \downarrow$	$AED_S \downarrow$
StyleGAN-V [44]	12.76	10.24	0.29	3.36	2.17	0.16
MRAA [43]	10.67	**2.46**	0.25	**2.65**	**1.08**	**0.12**
Ours	**6.15**	5.46	**0.22**	2.86	2.12	0.16

Table 2. Quantitative comparison on cross-identity (C) and same-identity (S) image animation. Our method achieves competitive results to SOTA image animation approaches as a byproduct of encoding video dynamics with Latent-ODEs.

Controlling local motion dynamics.. We observed a local correspondence between VidStyleODE dynamic latent representation and video motion dynamics, allowing for transferring local motion of body parts between different videos. In particular, given $\mathbf{z}_{d_A} \in \mathbb{R}^{8\times8}$ and $\mathbf{z}_{d_B} \in \mathbb{R}^{8\times8}$ corresponding to videos A and B respectively, we follow a blending operation to obtain a new dynamic latent code $\mathbf{z}_{d_{new}}$ as $\mathbf{z}_{d_{new}} = m\mathbf{z}_{d_A} + (1-m)\mathbf{z}_{d_B}$ where $m \in \{0,1\}^{8\times8}$ is a spatial mask. In Fig. 10, we show an example of transferring different body part movements (right hand or left leg) from different videos. To the best of our knowledge, we are the first that manage to control local motion dynamics. Additional results can be found in our supp. materials.

Ablation study. Tab. 3 shows the impact of each component of our model on the final FVD score. First, omitting CLIP consistency loss $\mathcal{L}_C$ or replacing it with MoCoGAN-HD temporal discriminator or StyleGAN-V discriminator affects both the training stability and temporal coherence of generated videos. Second, predicting the final $\mathbf{z}_t$ vector directly without any residuals drops the FVD score by 31 points. Additionally, replacing $\mathcal{L}_A$ and $\mathcal{L}_S$ with VGG-based perceptual loss significantly affects temporal coherence. Finally, we analyze the effect of further removing our cross-attention generator by concatenating all the vectors before AdaIN.

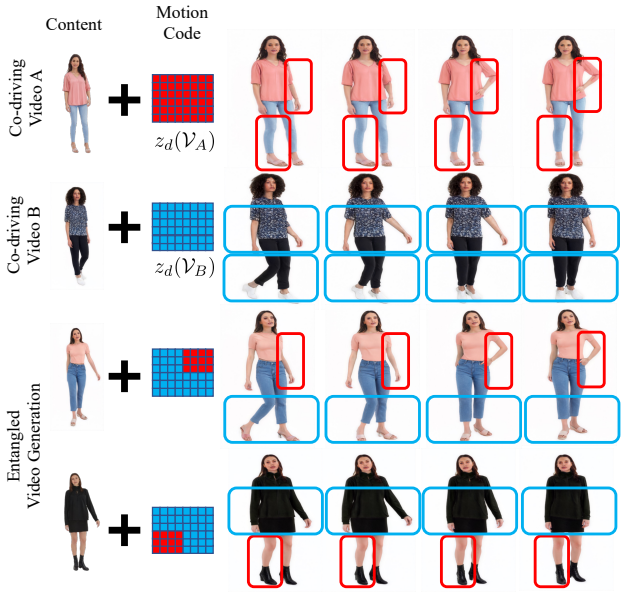

Figure 10. **Local motion dynamics control**. VidStyleODE can blend motion from two co-driving videos A and B, whose dynamics are depicted in first two rows. The last two rows show VidStyleODE 's ability to transfer dynamics from these driving videos in a local manner. The [red] and the [blue] boxes encode spatial regions where the motion dynamics are extracted and transferred.

Model Details	FVD $\downarrow$	$W_{error} \downarrow$
VidStyleODE	157.48	0.0075
w/o $\mathcal{L}_C$	191.08	0.0095
w/o $\mathcal{L}_C$, w/ SD	229.87	0.0084
w/o $\mathcal{L}_C$, w/ MD	245.04	0.0115
w/o $\mathcal{L}_C$, directions	222.76	0.0097
w/o $\mathcal{L}_A$, $\mathcal{L}_S$, and $\mathcal{L}_C$	244.49	0.0125
w/o $G(\cdot)$, $\mathcal{L}_A$, $\mathcal{L}_S$, and $\mathcal{L}_C$	265.90	0.0145

Table 3. **Ablation study on the Fashion Dataset**. MD refers to the temporal discriminator introduced in Mocogan-HD [45] and SD refers to the temporal discriminator from StyleGAN-V [44].

Conclusion

We have presented VidStyleODE , a novel method to disentangle the content and motion of a video by modeling *changes* in the StyleGAN latent space. As far as we are aware, it is the first method using a Neural ODE to represent motion in conjunction with StyleGAN, leading to a well-formed latent space for dynamics. By modifying content-dynamics combinations in different ways, we enable various applications. We have also introduced a novel consistency loss using CLIP that improves the temporal consistency without requiring adversarial training.

Limitations & future work. Our generation quality is limite by the pre-trained StyleGAN generator. Thus a test-time fine-tuning can help improving the results. Future work will involve 2^{nd}-order ODEs to enhance our dynamics representation and further explorations on text-guided editing of local dynamics.

Acknowledgements. Tolga Birdal wants to thank Google for their gifts.

References

[1] Rameen Abdal, Peihao Zhu, Niloy J. Mitra, and Peter Wonka. Video2stylegan: Disentangling local and global variations in a video, 2022. 2, 4, 14

[2] Yuval Alaluf, Or Patashnik, Zongze Wu, Asif Zamir, Eli Shechtman, Dani Lischinski, and Daniel Cohen-Or. Third time's the charm? image and video editing with stylegan3. In *Advances in Image Manipulation Workshop (AIM 2022) – in conjunction with ECCV 2022*, 2022. 2, 3

[3] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. *arXiv preprint arXiv:1511.06432*, 2015. 2, 3

[4] David Bau, Alex Andonian, Audrey Cui, YeonHwan Park, Ali Jahanian, Aude Oliva, and Antonio Torralba. Paint by word. *CoRR*, abs/2103.10951, 2021. 1

[5] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers, 2021. 5

[6] Ricky T. Q. Chen. torchdiffeq, 2018. 5

[7] Tian Qi Chen, Yulia Rubanova, Jesse Bettencourt, and David Kristjanson Duvenaud. Neural ordinary differential equations. In *NeurIPS*, 2018. 2, 3

[8] Helisa Dhamo, Azade Farshad, Iro Laina, Nassir Navab, Gregory D. Hager, Federico Tombari, and Christian Rupprecht. Semantic image manipulation using scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 2

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. 5

[10] Gereon Fox, Ayush Tewari, Mohamed Elgharib, and Christian Theobalt. Stylevideogan: A temporal generative model using a pretrained stylegan, 2021. 2, 14

[11] Anna Frühstück, Krishna Kumar Singh, Eli Shechtman, Niloy J. Mitra, Peter Wonka, and Jingwan Lu. Insetgan for full-body image generation, 2022. 2

[12] Jianglin Fu, Shikai Li, Yuming Jiang, Kwan-Yee Lin, Chen Qian, Chen-Change Loy, Wayne Wu, and Ziwei Liu. Stylegan-human: A data-centric odyssey of human generation. *arXiv preprint*, arXiv:2204.11823, 2022. 2, 5, 13, 14

[13] Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik, and Daniel Cohen-Or. Stylegan-nada: Clip-guided domain adaptation of image generators, 2021. 5

[14] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 2

[15] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *NIPS*, 2017. 5, 14

[16] Xun Huang and Serge J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1510–1519, 2017. 4

[17] Zhaoyang Huang, Xiaoyu Shi, Chao Zhang, Qiang Wang, Ka Chun Cheung, Hongwei Qin, Jifeng Dai, and Hongsheng Li. Flowformer: A transformer architecture for optical flow. *ArXiv*, abs/2203.16194, 2022. 14

[18] Ajay Jain, Matthew Tancik, and P. Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5865–5874, 2021. 4

[19] David Kanaa, Vikram Voleti, Samira Ebrahimi Kahou, and Christopher Pal. Simple video generation using neural odes. 2021. 3

[20] Levent Karacan, Tolga Kerimoğlu, İsmail Ata İnan, Tolga Birdal, Erkut Erdem, and Aykut Erdem. "disentangling content and motion for text-based neural video manipulation". In *Proceedings of the British Machine Vision Conference (BMVC)*, November 2022. 2, 3, 4, 5, 6, 7, 12, 13, 14

[21] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 5, 13

[22] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4396–4405, 2019. 14

[23] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 2, 3, 5

[24] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency, 2018. 5

[25] Seung Hyun Lee, Wonseok Roh, Wonmin Byeon, Sang Ho Yoon, Chanyoung Kim, Jinkyu Kim, and Sangpil Kim. Sound-guided semantic image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3377–3386, June 2022. 1

[26] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip H.S. Torr. Manigan: Text-guided image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1

[27] Tingle Li, Yichen Liu, Andrew Owens, and Hang Zhao. Learning visual styles from audio-visual associations. In *ECCV*, 2022. 1

[28] Steven R. Livingstone and Frank A. Russo. The ryerson audio-visual database of emotional speech and song (ravdess): A dynamic, multimodal set of facial and vocal expressions in north american english. *PLoS ONE*, 13, 2018. 5, 13, 14

[29] Andres Munoz, Mohammadreza Zolfaghari, Max Argus, and Thomas Brox. Temporal shift gan for large scale video generation, 2020. 2

[30] Sunghyun Park, Kangyeol Kim, Junsoo Lee, Jaegul Choo, Joonseok Lee, Sookyung Kim, and Edward Choi. Vid-ode: Continuous-time video generation with neural ordinary differential equation, 2020. 3

[31] Sunghyun Park, Kangyeol Kim, Junsoo Lee, Jaegul Choo, Joonseok Lee, Sookyung Kim, and Edward Choi. Vid-ode: Continuous-time video generation with neural ordinary differential equation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 2412–2422, 2021. 3

[32] Sunghyun Park, Kangyeol Kim, Junsoo Lee, Jaegul Choo, Joonseok Lee, Sookyung Kim, and Edward Choi. Vid-ode: Continuous-time video generation with neural ordinary differential equation. *arXiv preprint arXiv:2010.08188*, page online, 2021. 5, 13

[33] Taesung Park, Jun-Yan Zhu, Oliver Wang, Jingwan Lu, Eli Shechtman, Alexei A. Efros, and Richard Zhang. Swapping autoencoder for deep image manipulation. In *Advances in Neural Information Processing Systems*, 2020. 5, 13

[34] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. StyleCLIP: Text-driven manipulation of stylegan imagery. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2085–2094, 2021. 1, 2, 3, 4, 6

[35] Haonan Qiu, Yuming Jiang, Hang Zhou, Wayne Wu, and Ziwei Liu. Stylefacev: Face video generation via decomposing and recomposing pretrained stylegan3. *ArXiv*, abs/2208.07862, 2022. 2, 4, 14

[36] 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. 2, 4, 13, 14

[37] Davis Rempe, Tolga Birdal, Yongheng Zhao, Zan Gojcic, Srinath Sridhar, and Leonidas J Guibas. Caspr: Learning canonical spatiotemporal point cloud representations. *NIPS*, 33:13688–13701, 2020. 3

[38] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2287–2296, 2021. 2, 13

[39] Daniel Roich, Ron Mokady, Amit H. Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. *ACM Transactions on Graphics (TOG)*, 2022. 2, 6

[40] Yulia Rubanova, Ricky T. Q. Chen, and David K Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. 2

[41] Masaki Saito, Shunta Saito, Masanori Koyama, and Sosuke Kobayashi. Train sparsely, generate densely: Memory-efficient unsupervised training of high-resolution temporal GAN. *International Journal of Computer Vision*, 128(10-11):2586–2606, may 2020. 2

[42] Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. *ArXiv*, abs/1606.03498, 2016. 5, 14

[43] Aliaksandr Siarohin, Oliver J. Woodford, Jian Ren, Menglei Chai, and Sergey Tulyakov. Motion representations for articulated animation. 2021. 2, 3, 6, 8, 12, 14, 15

[44] Ivan Skorokhodov, S. Tulyakov, and Mohamed Elhoseiny. Stylegan-v: A continuous video generator with the price, image quality and perks of stylegan2. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3616–3626, 2022. 2, 3, 6, 7, 8, 12

[45] Yu Tian, Jian Ren, Menglei Chai, Kyle Olszewski, Xi Peng, Dimitris N Metaxas, and Sergey Tulyakov. A good image generator is what you need for high-resolution video synthesis. *arXiv preprint arXiv:2104.15069*, 2021. 2, 3, 4, 8

[46] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for StyleGAN image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021. 2

[47] Sergey Tulyakov, Ming-Yu Liu, Xiaodong Yang, and Jan Kautz. MoCoGAN: Decomposing motion and content for video generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 2, 3, 4

[48] Narek Tumanyan, Omer Bar-Tal, Shai Bagon, and Tali Dekel. Splicing vit features for semantic appearance transfer. *arXiv preprint arXiv:2201.00424*, 2022. 5

[49] Rotem Tzaban, Ron Mokady, Rinon Gal, Amit H. Bermano, and Daniel Cohen-Or. Stitch it in time: GAN-based facial editing of real videos, 2022. 2, 3, 5, 6, 7, 12, 14

[50] Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *ArXiv*, abs/1812.01717, 2018. 5

[51] Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *ArXiv*, abs/1706.03762, 2017. 4

[52] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *NeurIPS*, 2018. 4

[53] Yaohui Wang, François Brémond, and Antitza Dantcheva. Inmodegan: Interpretable motion decomposition generative adversarial network for video generation. *ArXiv*, abs/2101.03049, 2021. 2

[54] Yaohui Wang, Di Yang, Francois Bremond, and Antitza Dantcheva. Latent image animator: Learning to animate images via latent space navigation. *arXiv preprint arXiv:2203.09043*, 2022. 2

[55] Tianyi Wei, Dongdong Chen, Wenbo Zhou, Jing Liao, Zhentao Tan, Lu Yuan, Weiming Zhang, and Nenghai Yu. Hairclip: Design your hair by text and reference image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18072–18081, 2022. 2, 3, 6, 7, 12, 14

[56] Weihao Xia, Yujiu Yang, Jing-Hao Xue, and Baoyuan Wu. Tedigan: Text-guided diverse face image generation and manipulation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2256–2265, 2021. 1, 2

[57] Weihao Xia, Yulun Zhang, Yujiu Yang, Jing-Hao Xue, Bolei Zhou, and Ming-Hsuan Yang. Gan inversion: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 3

[58] Wilson Yan, Yunzhi Zhang, P. Abbeel, and A. Srinivas. Videogpt: Video generation using vq-vae and transformers. *ArXiv*, abs/2104.10157, 2021. 2

[59] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. Describing videos by exploiting temporal structure, 2015. 5

[60] Xu Yao, Alasdair Newson, Yann Gousseau, and Pierre Hellier. A latent transformer for disentangled face editing in images and videos. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 13789–13798, 2021. 2, 3, 5, 6, 7

[61] Fei Yin, Yong Zhang, Xiaodong Cun, Mingdeng Cao, Yanbo Fan, Xuan Wang, Qingyan Bai, Baoyuan Wu, Jue Wang, and Yujiu Yang. Styleheat: One-shot high-resolution editable talking face generation via pre-trained stylegan. In *ECCV*, 2022. 2, 4

[62] Çağatay Yıldız, Markus Heinonen, and Harri Lähdesmäki. Ode^{2}vae: Deep generative second order odes with bayesian neural networks, 2019. 3

Appendices

We now discuss several design choices, introduce ablation studies, and implementation details. We also provide additional qualitative and quantitative results both on Fashion Videos and RAVDESS datasets. To view a comprehensive collection of videos from all our different applications, you can access the website https://cyberiada.github.io/VidStyleODE

Contents

A. Discussions

We now discuss some interesting trends, comparisons, and trade-offs between models, supported by quantitative and qualitative results.

On inversion vs. quality. HairCLIP [55] manages to consistently obtain high results on manipulation accuracy across both datasets. However, it is highly dependent on high-quality inversions to obtain good results. On the Fashion dataset, where the base inversion quality is not good enough, we see very bad results across the other metrics. However, on RAVDESS, the inversion quality is much higher, due to taking advantage of the models trained on FFHQ. Therefore, we see very good results across all metrics.

Perceptual quality vs. manipulation capability. STIT [49] performs quite well on consistency and perceptual quality metrics, primarily due to the fine-tuning of the generator. However, by ensuring high-quality results, it reduces the ability to manipulate the videos effectively, and so is consistently behind multiple other models. Additionally, high-frequency details of the videos (such as shoes, hair, and complex color patterns) are lost due to the focus on reducing distortion.

DiCoMoGAN [20] primarily acts as an autoencoder, with additional steps for manipulation. On RAVDESS, where the videos are not too complicated to learn, this allows DiCoMoGAN to obtain very high results on all the perceptual quality metrics. However, this auto-encoding property also restricts the ability to manipulate accurately, leading to poor results for that metric.

Complexity of dynamics vs. generation quality. StyleGAN-V [44] had a lot of challenges learning the correct motion of the Fashion dataset and frequently suffered from mode collapse. This led to very poor perceptual quality results, as well as a very high warping error. On the RAVDESS dataset, there were fewer training issues, which contributed to relatively better results. However, all the perceptual quality metrics are still very poor.

The primary issue with MRAA [43] is the inability to distinguish between what motion should be transferred and what should not, as well as retaining key structural details of the reference frame. As seen in Fig. 14, MRAA transforms the dress into pants, following the style of the driving video. Additionally, because the sleeve of the right arm is not visible in the reference frame, it attempts to copy the sleeve style of the driving video, leading to inconsistencies between the two sleeve lengths. In Fig. 15, it also attempts to transfer sleeve length, even when the right arm is visible in the reference frame. Not but not least, MRAA also removes a lot of the fine detail of the clothing. Therefore, despite being able to properly capture the motion of the people, in both cases, it is unable to create a complete and consistent video.

On trade-offs. Notably, most models were unable to handle all tasks effectively: **generation**, **disentanglement**, and **manipulation**. While some are very good at manipulation, others obtained high perceptual quality. However, VidStyleODE hits a

sweet spot. On the Fashion dataset, it consistently achieves very good results across all metrics, including being the best in many of them. On RAVDESS, VidStyleODE is able to achieve very good consistency and manipulation accuracy, while still reporting competitive perceptual quality metrics. Therefore, it does not suffer from the same trade-offs between manipulation, consistency, and perceptual quality as the other models.

B. Architectural Details

Spatiotemporal encoder f_C. We use a pre-trained StyleGAN2 inversion network to obtain the K input frames' latent representation in the $\mathcal{W}_+$ space $\mathbf{Z} := \{\mathbf{z}_i^l \in \mathcal{W}_+\}_{i=1}^K$. We freeze the inversion network's weights during training. Then, we take the expectation of $\mathbf{Z}$ to obtain the video's global latent code $\mathbf{z}_C$. During inference, the global latent code can be sampled or obtained from a single frame. In our experiments, we used pSp inversion network [38] pre-trained on StylishHumans-HQ Dataset [12] for fashion video experiments, and on FFHQ [21] for face video experiments.

Dynamic representation network f_D. We first process the K video frames $X_i \in \mathbb{R}^{M \times N \times 3}$ independently using a $2D$ ResNet encoder architecture based on the implementation of [33] to extract K feature maps $\mathbf{z}_r \in \mathbb{R}^{m_d \times n_d \times d_{sp}}$. In our experiments with the fashion videos dataset, we used $M = 128$, $N = 96$, $m_d = 8$, $n_d = 6$, and $d_{sp} = 64$. Additionally, for face videos experiments, we used $M = 128$, $N = 1128$, $m_d = 8$, $n_d = 8$, and $d_{sp} = 64$. Subsequently, we adapt ConvGRU from [32] to extract dynamic latent representation $\mathbf{z}_d \in \mathbb{R}^{m_{ode} \times n_{ode} \times 512}$ from $\mathbf{z}_R = \{\mathbf{z}_{r_i}\}_{i=1}^K$. For all of our experiments, we set $m_{ode} = m_d$ and $n_{ode} = n_d$. We use the dynamic representation to initialize an autonomous latent ODE

$$\mathbf{z}_{dT} = \phi_T(\mathbf{z}_{d0}) = \mathbf{z}_{d0} + \int_0^T f_\theta(\mathbf{z}_{dt}, t)\, dt, \tag{11}$$

Where $z_{d0} = z_d$. We parameterize f_θ as a convolutional network obtained from [32]. For every training batch, we sample n frames from each video and solve the ODE at their corresponding timestamps to obtain their spatiotemporal feature representation $\mathbf{z}_{dT} = \{\mathbf{z}_{d_{t_i}}\}_{i=1}^n$.

Obtaining style code.. To guide the manipulation, we condition the video reconstruction on an external style code $\mathbf{z}_{\text{Style}}$. We represent this style code in the CLIP [36] embedding space by encoding the content frame X_c, source description $\mathcal{D}_{\text{SRC}}$ of the appearance of the video, and a target description $\mathcal{D}_{\text{TGT}}$. To obtain the content frame, we decode the latent global code using a pre-trained StyleGAN2 generator $G(\cdot)$.

$$\mathbf{z}_{\text{Style}} = \text{CLIP}_I(G(\mathbf{z}_C)) + \alpha(\text{CLIP}_T(\mathcal{D}_{\text{TGT}}) - \text{CLIP}_T(\mathcal{D}_{\text{SRC}})) \tag{12}$$

where CLIP_I and CLIP_T are the CLIP image and text encoder, respectively. α is a user-defined parameter that controls the level of manipulation during inference time. For all of our quantitative experiments, we used $\alpha = 1$.

Conditional generator model. Once the video global code $\mathbf{z}_c$, the frames dynamic representation z_d, and the video style z_{Style} have been collected, we apply N layers of self-attention onto the different spatial components of z_d. Then, we perform cross-attention between the outputs of the self-attention and the style vector z_{Style}. At each layer of cross-attention, we predict and apply an offset to the style code in the CLIP space. We then take the final output style vector and modulate it over the global code z_c. This produces our direction, which is then added to the original code:

$$\mathbf{z}_t = \mathbf{z}_c + \Delta\mathbf{z} \tag{13}$$

The output frame at time t is then generated as

$$\mathbf{X}_t = G\left(\mathbf{z}_t\right) \tag{14}$$

Hyper-parameters. The appearance and structural losses both have $\lambda_S = 10, \lambda_A = 10$. The latent loss has $\lambda_L = 1.0$. For the directional clip loss, we have $\lambda_D = 2.0$. For the consistency loss, we use a scheduler to go from 0.01 to 1 over 40000 steps. For the trade-off between the structural and appearance loss, we use $\lambda = 0.5$, so that both are equally important.

In our self-attention network, we use 12 layers, each with 8 heads, as well as a hidden dimension size of 512. Both the coarse and medium layers receive the dynamics, while the fine layers do not.

C. Details on the Datasets & Evaluations

Datasets. All our results were evaluated on the Fashion dataset [20] and the RAVDESS dataset [28]. The Fashion dataset contains descriptions already, which we used for our manipulations. On RAVDESS, we hand-crafted descriptions for each of the 24 actors, which we used during training and testing for manipulation.

Method	Fashion Videos					RAVDESS				
	FVD ↓	IS ↑	FID ↓	Acc. ↑	W_{error} ↓	FVD ↓	IS ↑	FID ↓	Acc. ↑	W_{error} ↓
Ours	157.48	3.25	**26.28**	0.87	**0.0075**	273.10	**1.33**	**34.92**	**0.83**	0.0076
Ours w/ FT	**139.69**	**3.27**	31.69	0.87	0.0096	**160.90**	1.32	36.48	0.79	**0.0049**

Table 4. **Effect of fine-tuning the pre-trained generator and inversion network.** Fine-tuning StyleGAN-2 image generator and inversion network (Ours w/ FT) significantly improves the FVD score, with a minimum effect on the perceptual quality and manipulation capabilities of our model.

Evaluation metrics. We evaluate our model in terms of perception, temporal smoothness, and editing consistency of the generated videos as well as the accuracy of the applied manipulation. The *Frechet Video Distance (FVD)* [**?**] score measures the difference in the distribution between ground truth (GT) videos and generated ones, evaluating both the motion and visual quality of the video. To compute the metric, we used 12 frames sampled at 10 frames per second. *Inception Score (IS)* [42] measures the diversity and perceptual quality of the generated frames. To eliminate any gain in IS from the diversity resulting in inconsistency in the video frames, we use only a single frame from each generated video. *Frechet Inception Distance (FID)* [15] measures the difference in distribution between GT and generated videos. Similar to IS, we use only a single frame from each generated video to calculate FID. *Warping Error* predicts subsequent frames of a video using an optical flow network, and compares this with the generated frames, to measure consistency. The network we used is [17]. *Manipulation Accuracy* measures the accuracy of the manipulation in the generated video according to the target textual description, and relative to the GT description of the video. We used CLIP [36] as a zero-shot classifier for this task.

Baselines. We trained HairCLIP [55] on the Fashion Videos dataset by omitting the attribute preservation losses concerning face images. For Latent Transformer, we followed the authors' instructions and trained the classifier for 20 epochs and the models for 10 epochs each. For HairCLIP, Latent Transformer, and STIT [49] on the Fashion dataset, we used the StyleGAN-Human [12] pre-trained generator. For RAVDESS, we used the FFHQ pre-trained generator for all 3 models. For DiCoMoGAN [20], we trained the official code until convergence.

For MRAA [43] on the Fashion dataset, we followed the training procedure provided by the authors for the tai-chi dataset. On the RAVDESS dataset, we used the training procedure provided for the VoxCeleb dataset.

We trained StyleGAN-V 3 times on each dataset for 1 week, using 2 V100 gpus. We picked the best model according to FVD (fvd2048_16f), and used this for all metric calculations and figures. On both datasets, we noticed that later iterations suffered from significant mode collapse. Therefore, we also picked the epoch with the best FVD. For manipulation, we projected real videos using 1000 iterations.

D. Further Quantitative Results

D.1. Fine-tuning pre-trained networks

A key motivation for this work is to develop a method that can generate and manipulate high-resolution videos (e.g. 1024×512) even when trained on low-resolution ones (e.g. 128×96 for Fashion Videos). This impacted our choices for the architecture design and training objectives. For instance, fine-tuning the pre-trained image generator on the low-resolution training video dataset defies our original motivation to generate high-resolution videos. Additionally, while reconstruction loss between the generated and ground truth frames has been used in prior work [1, 10, 35] to reconstruct local dynamics, it often trades the lower distortion with the worse perceptual quality. However, in certain scenarios where a high-resolution training dataset is available, fine-tuning the generator and inversion networks is possible. We report in Tab. 4 the performances of VidStyleODE on Fashion Videos, where a generator and an inversion network pre-trained on Stylish-Humans-HQ Dataset [12] were fine-tuned on Fashion Videos at a resolution of 1024×512 for $200k$ iterations. We also present results of VidStyleODE with a pre-trained generator and inversion networks trained on FFHQ [22] and fine-tuned on RAVDESS [28] at a resolution of 1024×1024 for $250k$ iterations. For both experiments, we trained VidStyleODE at a low resolution, i.e. 128×96 for Fashion Videos and 128×128 for RAVDESS.

D.2. Further Ablation Studies

In **??** of the main paper, we analyzed the contribution of each component of our model to the final FVD and W_{error} scores on the Fashion Videos, showing the superiority of our proposed CLIP temporal consistency loss $\mathcal{L}_C$ over the MoCoGAN-HD temporal discriminator or the StyleGAN-V discriminator, as well as the validity of our architecture choices. We further analyze the effect of different strategies for obtaining the video global latent code. Specifically, we consider using the $\mathcal{W}_+$ latent code

Inference \ Training	First Frame	Mean Frame
First Frame	169.62	181.64
Random Frame	206.98	182.39
Mean Frame	229.92	150.59

Table 5. A comparison of different methods to obtain the global content representation for the Fashion dataset, in terms of FVD over same-identity image animation. Each row represents a different method of training, while each column represents inference using the stated global representation. Encoding video content with the mean $\mathcal{W}_+$ latent code of the input frames during training provides a better FVD score with less sensitivity to the content frame position during inference.

of the first frame, a random frame, or the mean latent code. Tab. 5 shows that encoding the video content as the mean $\mathcal{W}_+$ latent code (*i.e.* $\mathbf{z}_C = \mathbb{E}[\mathbf{Z}]$) provides an overall better FVD, with less sensitivity to the frame order during inference (*e.g.* First vs Last Frame).

E. Further Qualitative Results

In this section, we provide additional qualitative results obtained with our proposed VidStyleODE and further comparisons against the state-of-the-art.

E.1. Latent motion representation

Our model is able to learn a meaningful latent space for motion, which enables multiple applications. As seen in Fig. 11, interpolating between two motion representations produces a smooth combination of the two motions. Additionally, our motion representation contains spatial dimensions as well as a time dimension. By having access to local representations as well as a global representation for motion, we are able to manipulate only certain spatial parts of a video, or optionally the entire video. Fig. 12 shows the swapping of the upper right quarter of the motion representation while keeping the rest of it untouched. This is able to affect only the upper right quarter of the video, corresponding to the arm movement. Meanwhile, the other parts of the body follow the original motion path. We are also able to swap the global motion representation between two videos, resulting directly in the swap of the dynamics of the videos. This immediately allows for image animation, when combined with the global content representation, which can be obtained from a single frame if necessary. This is done by using the global motion representation from the driving video while extracting the global content representation from the source frame. Examples and comparisons with a popular image animation model [43] are found in Figs. 14 and 15.

E.2. Fashion Videos dataset

- Fig. 11 shows the results of interpolating between two dynamic representations. Note especially the decrease in right arm movement, which happens smoothly as λ moves from 0.0 to 1.0. Also, the legs spread out less at $t = 25$ as we increase λ.
- In Fig. 12, we provide an additional example of controlling local motion dynamics of a figure. In particular, we show that our model is able to transfer the right arm movements of another figure to the target figure.
- Fig. 13 provides qualitative comparisons for our VidStyleODE model to the existing methods.
- Fig. 14 shows a comparison of our VidStyleODE model to multiple image animation methods. It can be seen that our method is the only one to transfer motion while maintaining the structure and perceptual quality of the reference frame.
- Fig. 15 provides a second comparison of image animation methods.
- Fig. 16 supports the quantitative ablations provided in the main paper with qualitative results, showing that the quality of the video does indeed improve as we add more components to the model.
- Figs. 17 and 18 show text-guided editing examples on two sample videos from the Fashion Video dataset for two distinct target texts for each source video. As clearly seen, our method, VidStyleODE , performs the necessary edits as suggested by the target descriptions successfully.
- Fig. 19 shows the ability of our method in generating realistic and consistent frames via interpolation. Our method accurately estimates the latent codes of the frames with the missing timestamps and generates the frames in an accurate manner.
- In Fig. 20, we demonstrate that our VidStyleODE method animates still frames via extrapolation. As seen, it generates a video depicting visually plausible and temporally consistent movements, showing the effectiveness of our method.

Figure 11. Obtaining the dynamic representation from two videos, we interpolate between them with values $\lambda = 0.0$, $\lambda = 0.5$, and $\lambda = 1.0$, and show that the dynamics do change smoothly as we interpolate. We interpolate over 25 frames.

E.3. RAVDESS dataset

- In Fig. 21, we present sample text-guided editing examples on a sample video from the RAVDESS dataset for three different target texts. Our proposed VidStyleODE method accurately manipulates the provided source videos in a temporally-consistent way according to the provided target text. It successfully changes the eye color, the hair color, and the gender of the person of interest.
- Fig. 22 provides example frame interpolation results over the provided face frames. As seen, our VidStyleODE method accurately predicts what the frames from the missing timestamps look like.
- In Fig. 23, we show that our VidStyleODE method can also animate still frames via extrapolation.
- Finally, in Fig. 24, we give qualitative comparisons against state-of-the-art editing techniques on a sample video having the source description "*A woman with blond hair, and green eyes*" and with the target description being specified as "*A woman with brown hair and blue eyes*". As seen, compared to the state-of-the-art methods, VidStyleODE generates a temporally coherent output depicting all the proper edits done on the source video.

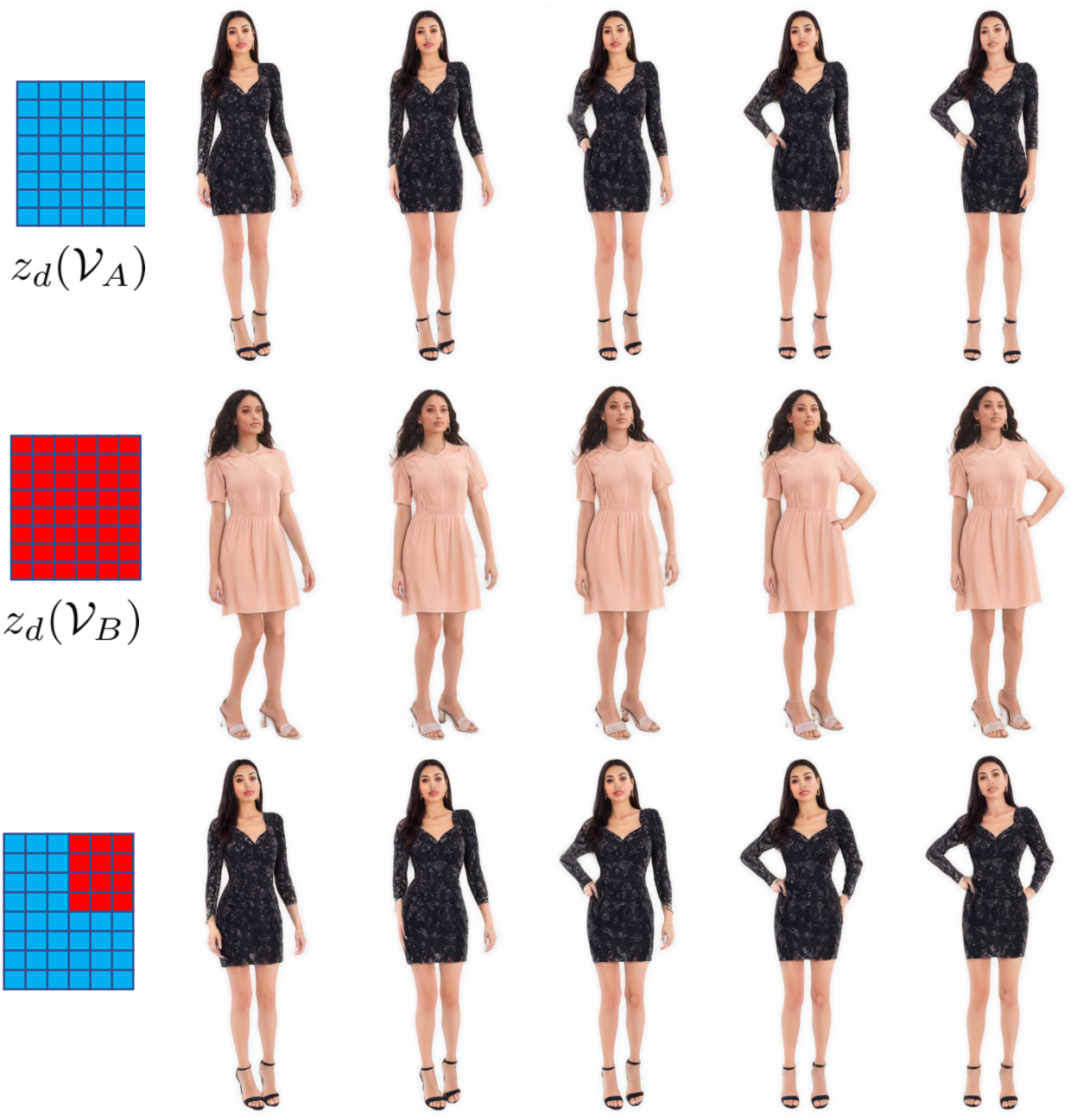

Figure 12. We transfer the upper right dynamics from $\mathcal{V}_B$ to $\mathcal{V}_A$, while keeping the rest of the dynamics from $\mathcal{V}_A$. This results in the right arm moving upwards, while the rest of the dynamics are unchanged.

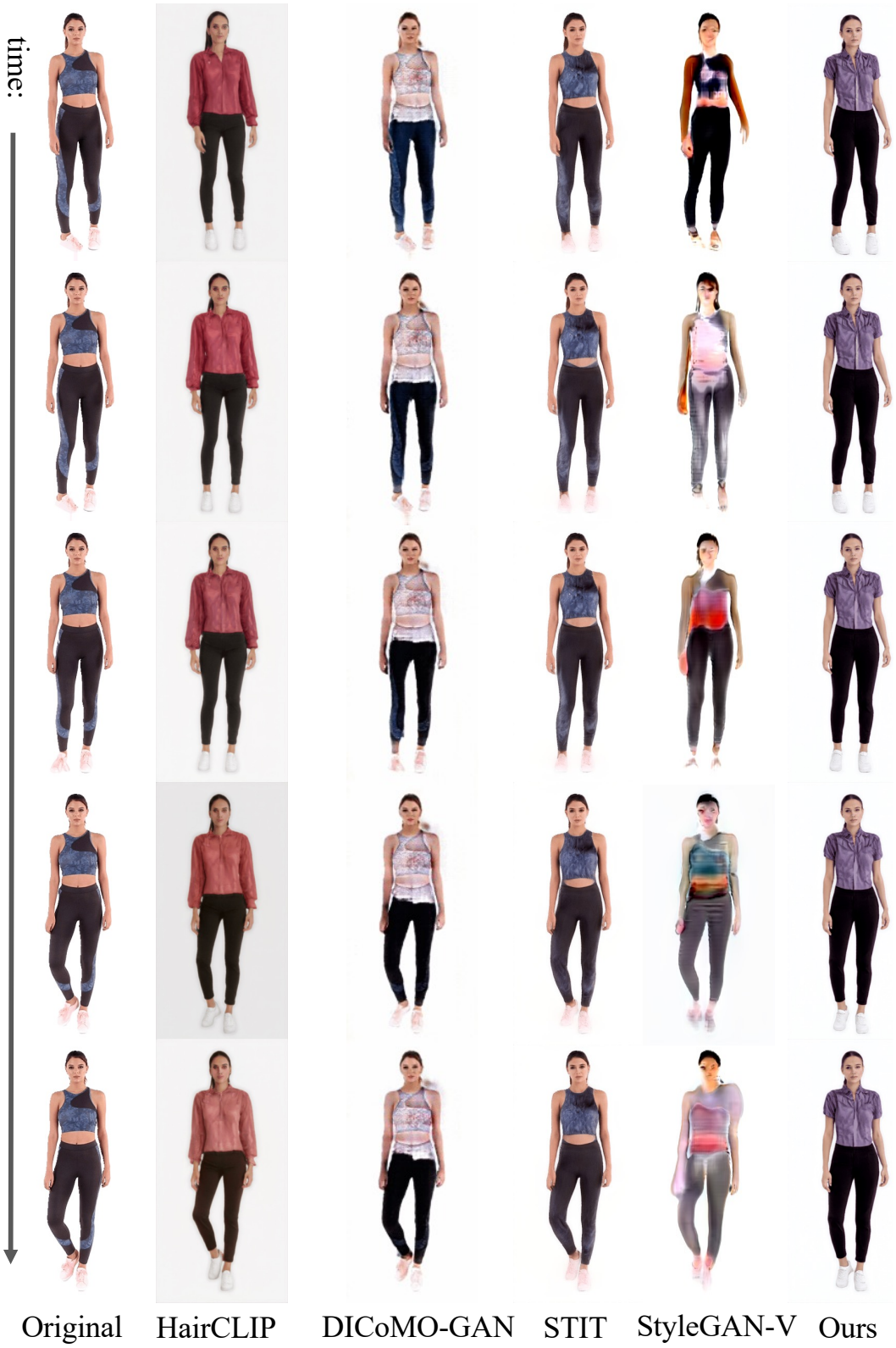

Figure 13. Here, we perform additional manipulations using the baselines. We exclude the latent transformer results since it is unable to perform complex manipulations without multiple steps. The source text is "*a photo of a woman wearing a crop top*", and the target text is "*a photo of a woman wearing a* ***blouse***".

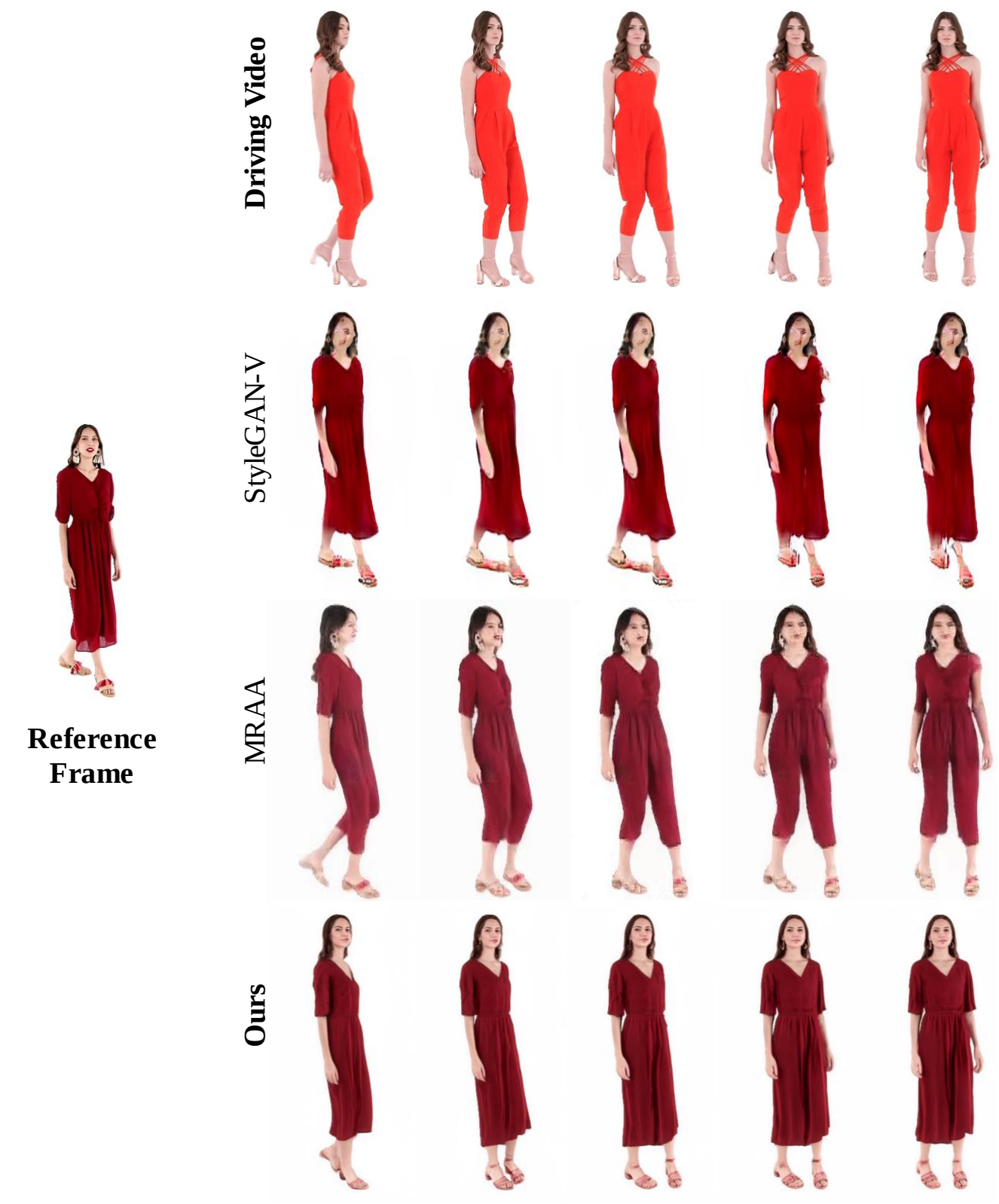

Figure 14. We perform image animation using our model and multiple baselines. We obtain the dynamics from the driving video in the first row and apply it to the reference frame to generate the videos. Our results obtain the highest perceptual quality, while also matching the dynamics of the driving video, and structure of the reference frame. MRAA modifies the structure according to the driving video, instead of just transferring motion, while StyleGAN-V has some motion transferred, but has a very low perceptual quality.

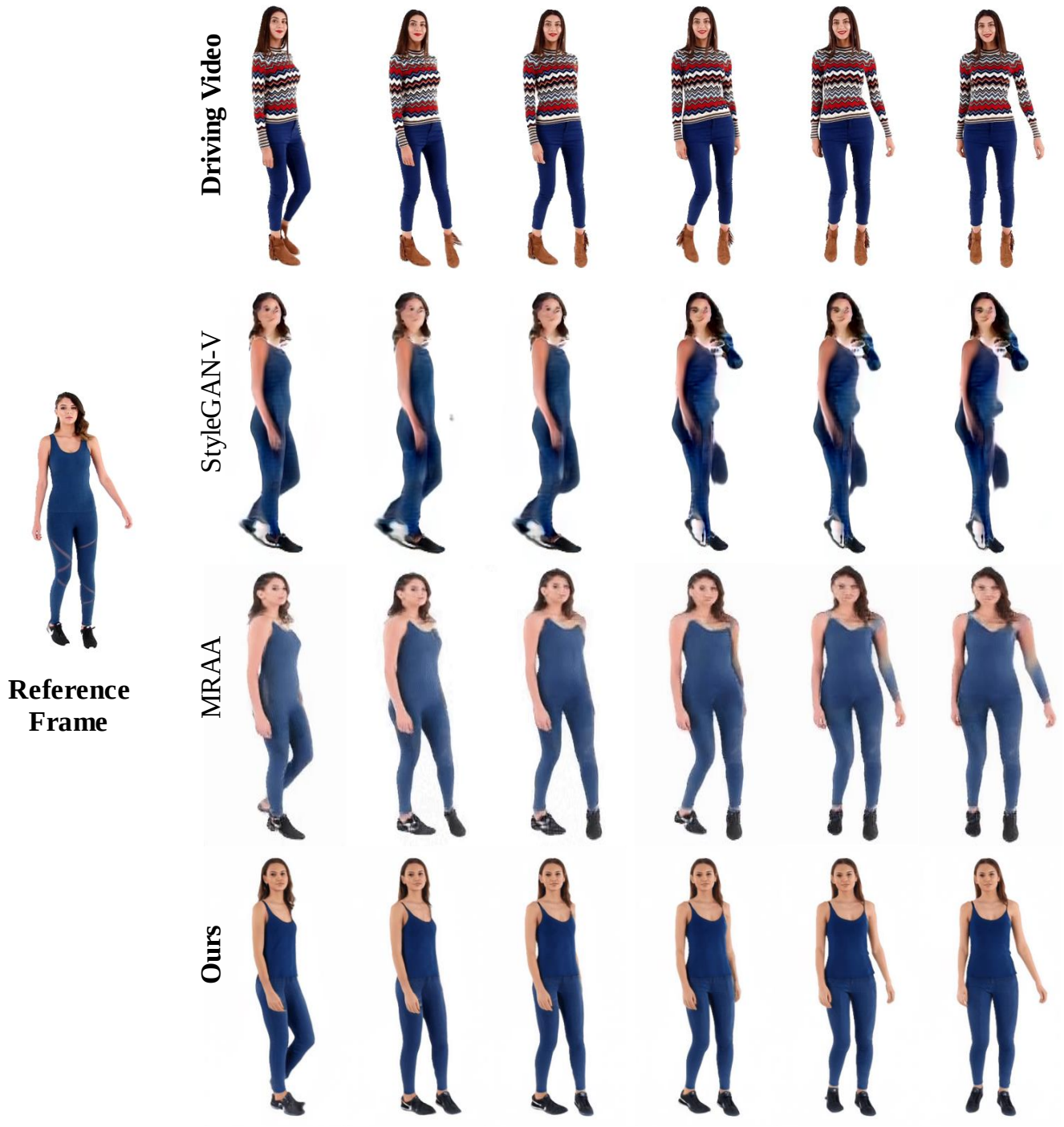

Figure 15. Another example on image animation. Our model produces the result with the highest perceptual quality while being consistent with the driving video and reference frame. MRAA has some inconsistencies in the arm, and StyleGAN-V is unable to capture the motion in any meaningful way.

Figure 16. We provide examples to support our ablation study. The first row is the model without the conditional generative model f_G, structural loss, appearance loss, or consistency loss. The second row is without structural loss, appearance loss, or consistency loss. The third row is without consistency loss, and without using directions. The fourth row is just without consistency loss. Finally, the fifth row is our best model, with everything.

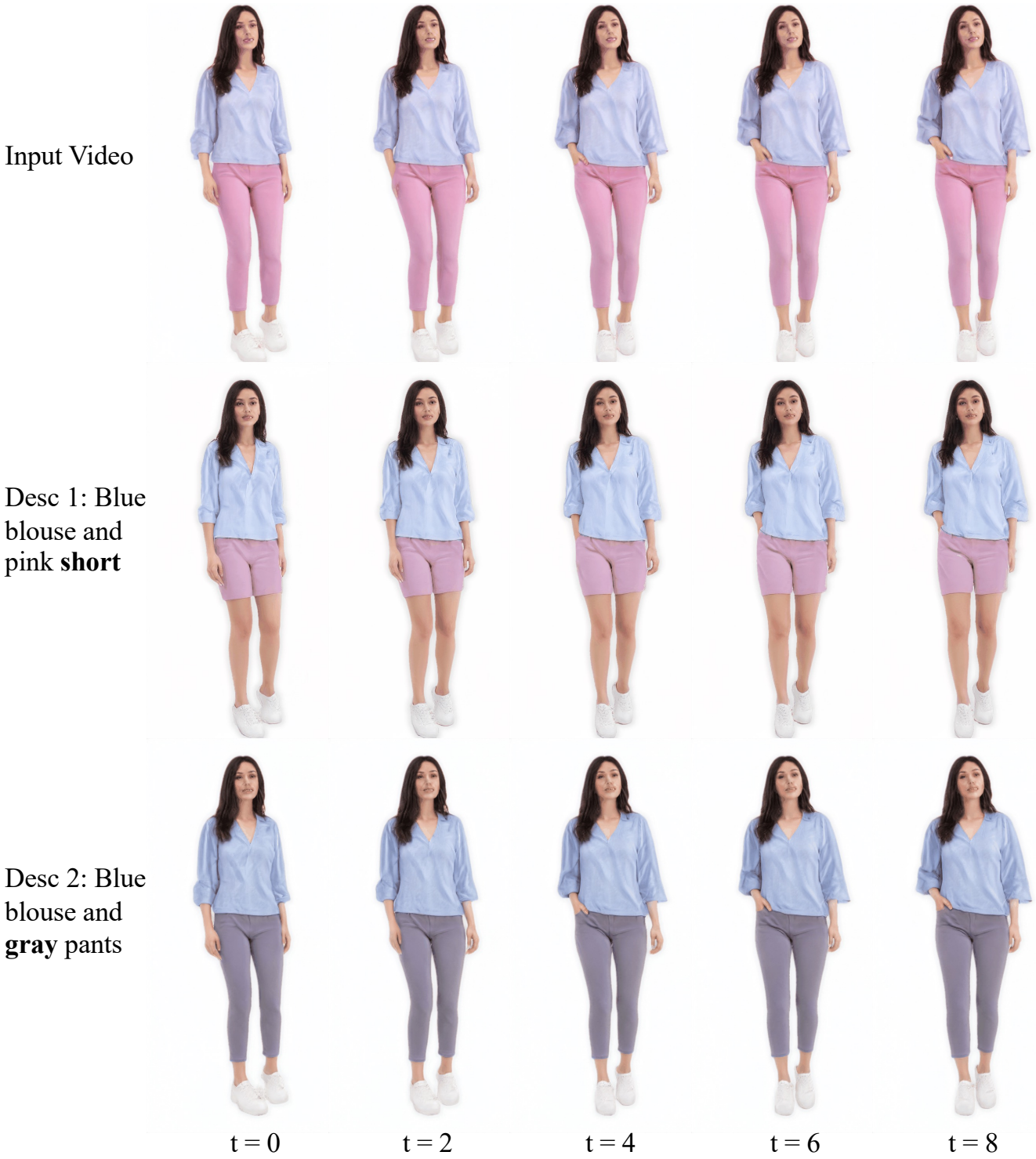

Figure 17. We perform two different manipulations to a sample video (the source video) from the Fashion Videos dataset and display the corresponding results here. Target 1 uses the target text "*A photo of a woman wearing blue blouse and pink **short***". Target 2 uses the target text "*A photo of a woman wearing blue blouse and **gray** pants*".

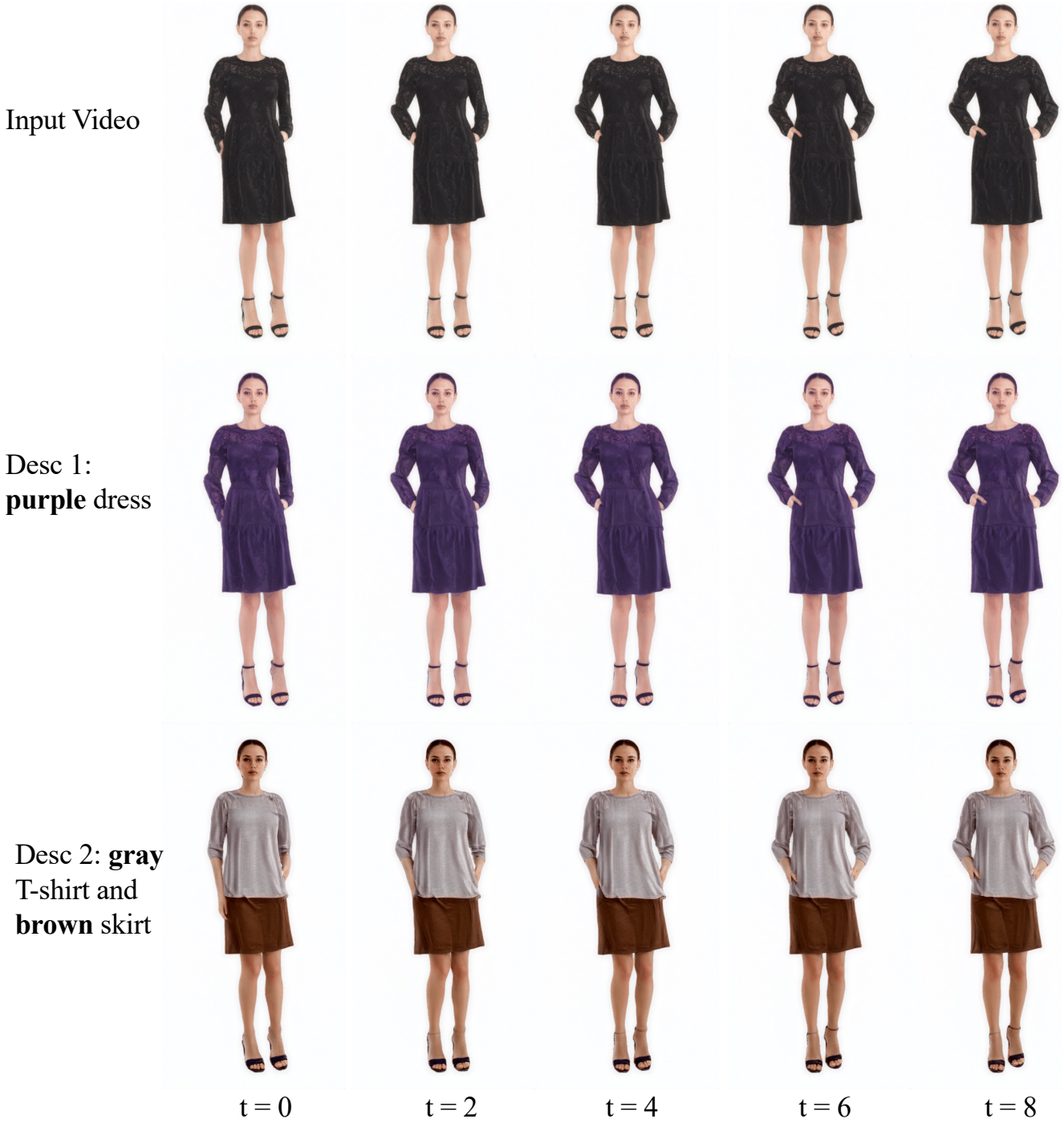

Figure 18. We perform two different manipulations to a sample video (the source video) from the Fashion Videos dataset and display the corresponding results here. Target 1 uses the target text "*A photo of a woman wearing a **purple** dress*". Target 2 uses the target text "*A photo of a woman wearing **gray** T-shirt and **brown** skirt.*".

Figure 19. To perform interpolation, we provide the first and last frames (shown in blue) to the model and then generate the whole video. We display 3 evenly-spaced interpolated frames for each video (shown in red).

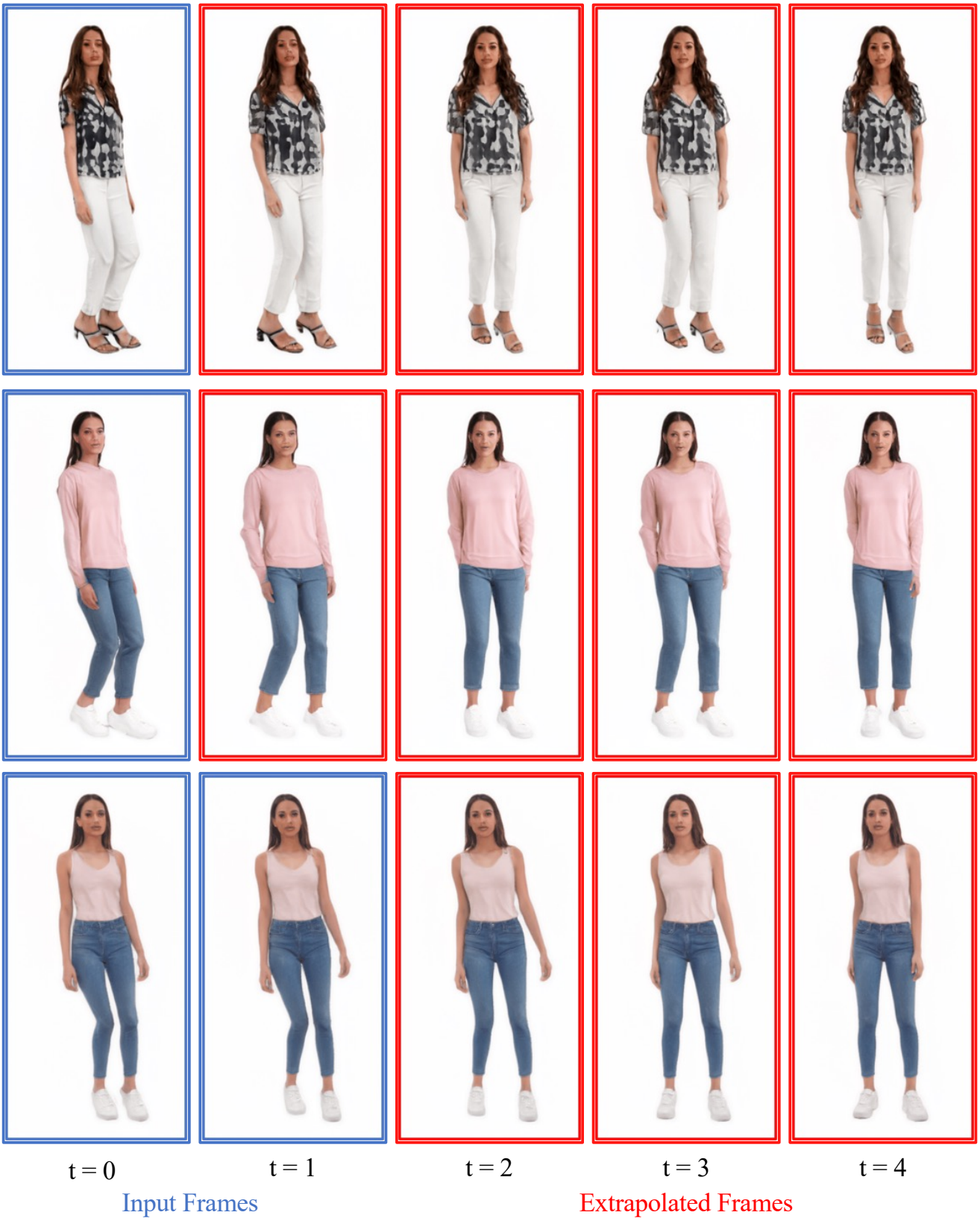

Figure 20. To perform extrapolation from a single frame, we provide just the initial frame (shown in blue), and then generate the next 4 frames (shown in red).

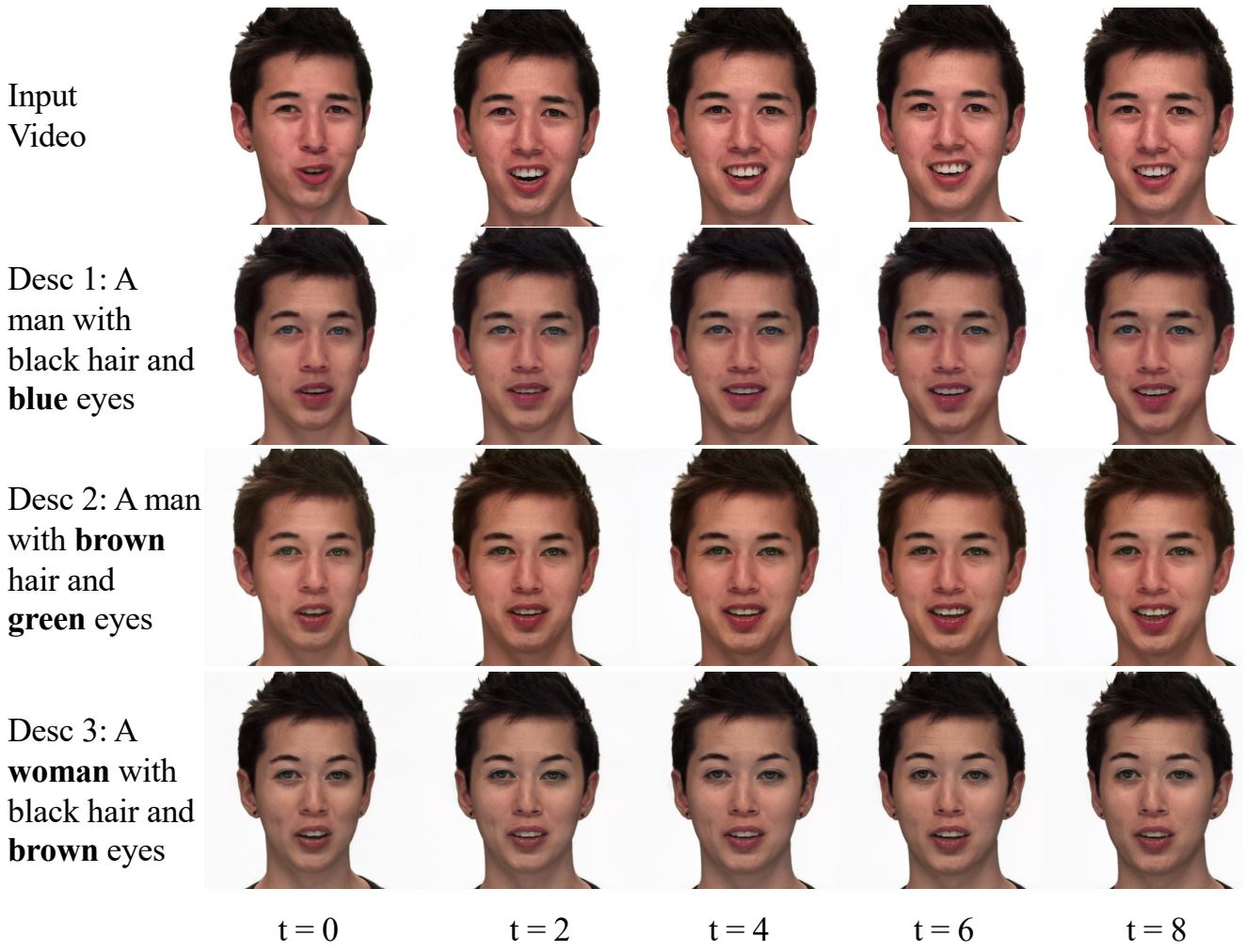

Figure 21. We perform three different manipulations to a sample video (the source video) from the RAVDESS dataset, and display the corresponding results here. Target 1 uses the target text "*A man with black hair and **blue** eyes*". Target 2 employs the target text "*A man with **brown** hair and **green** eyes.*". Target 3 uses the target text "*A **woman** with black hair and brown eyes.*".

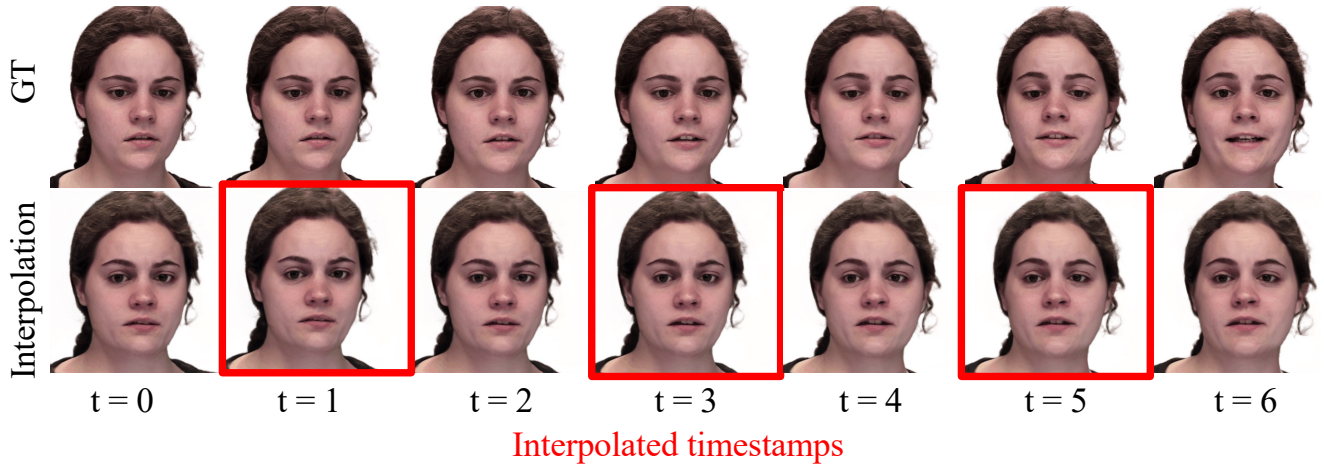

Figure 22. To perform interpolation, we provide four distinct frames with different timestamps ($t = 0$, $t = 2$, $t = 4$, and $t = 6$) (shown in blue) to the model, and then generate the unobserved frames for timestamps $t = 1$, $t = 3$, and $t = 5$ (shown in red).

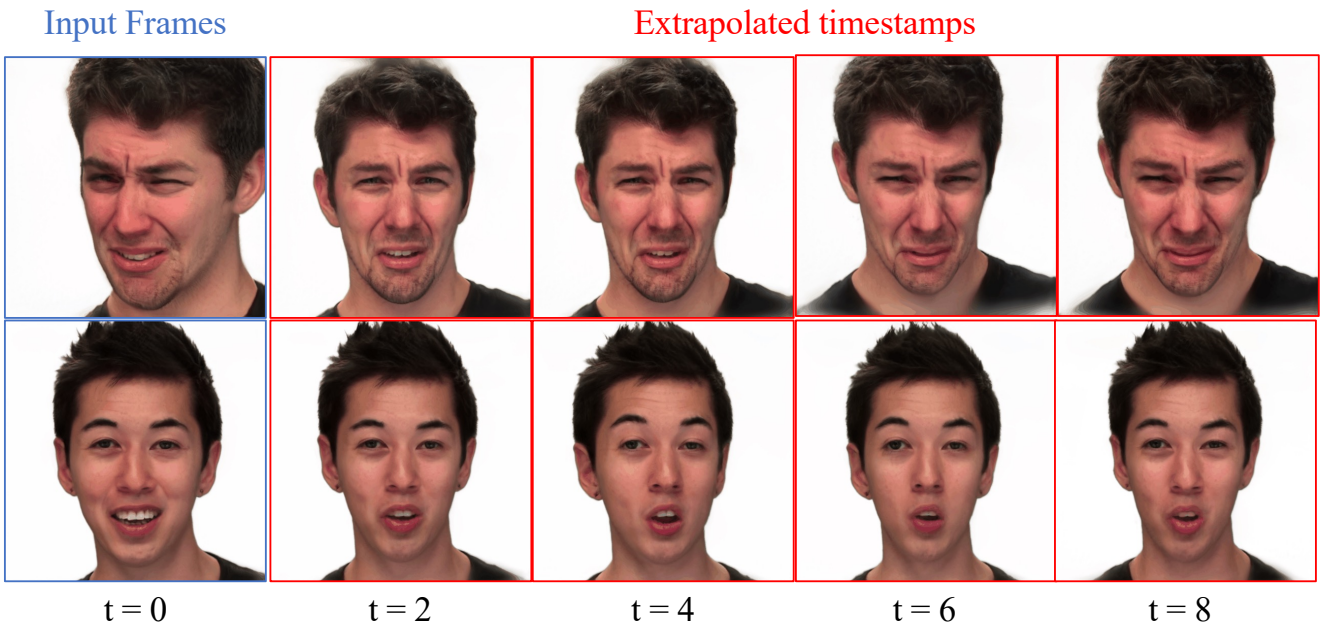

Figure 23. Extrapolation from a single frame: we provide just the initial frame (shown in blue), and then generate the next 4 frames (shown in red).

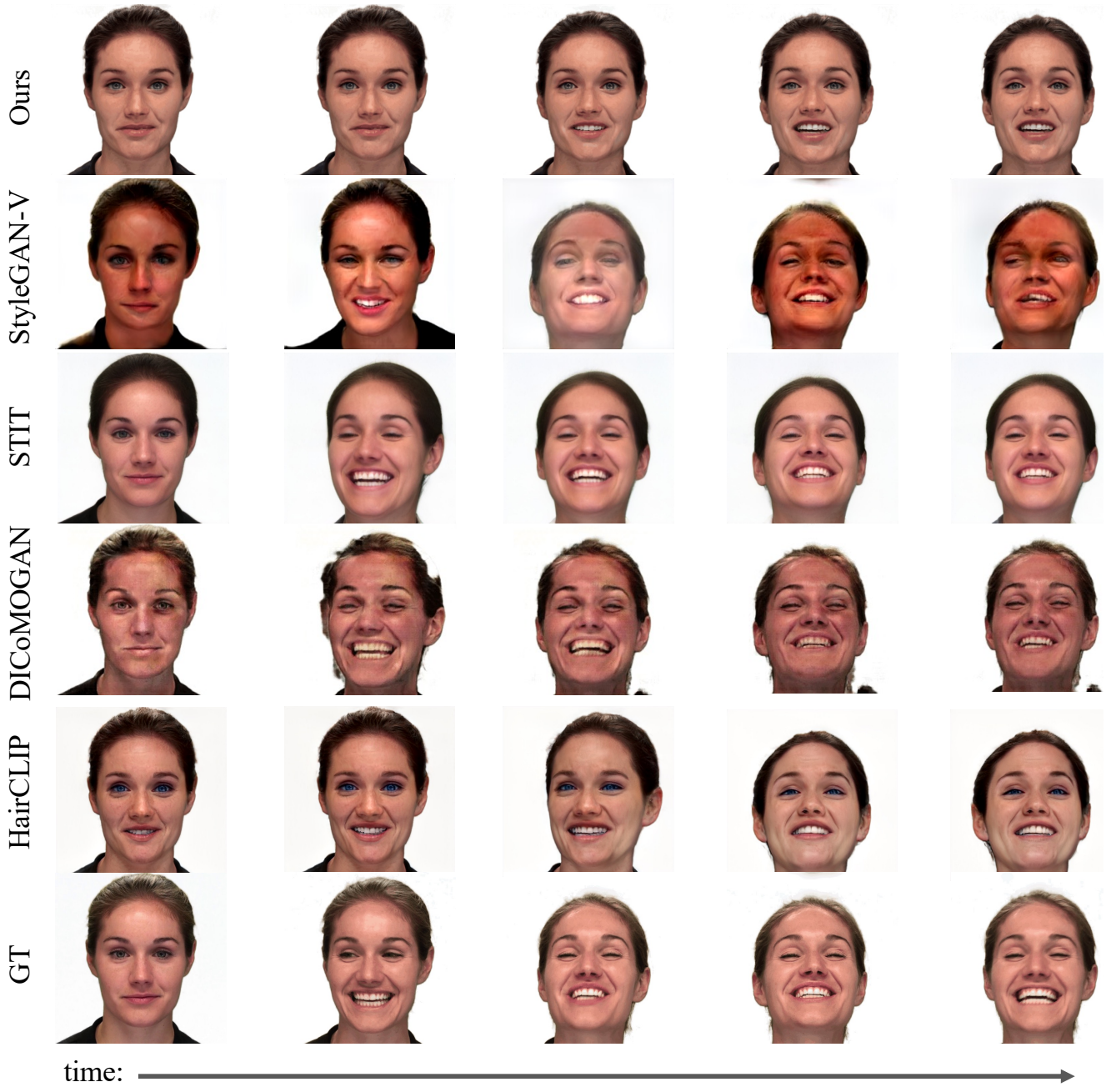

Figure 24. Qualitative results of our approach and the competing editing methods. The description of the source image is "*A woman with blond hair, and green eyes*", while the target description is specified as "*A woman with **brown** hair and **blue** eyes*".