VidStyleODE: Disentangled Video Editing via StyleGAN and NeuralODEs Supplemental Material

Moayed Haji Ali* Andrew Bond*
Koç University
{mali18, abond19}@ku.edu.tr

Tolga Birdal
Imperial College London
tbirdal@imperial.ac.uk

Duygu Ceylan
Adobe Research
ceylan@adobe.com

Levent Karacan
Iskenderun Technical University
levent.karacan@iste.edu.tr

Erkut Erdem
Hacettepe University
erkut@cs.hacettepe.edu.tr

Aykut Erdem
Koç University
aerdem@ku.edu.tr

In this supplementary document, we 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

1. Discussions

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

On inversion vs. quality. HairCLIP [19] 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 [18] 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

*Equal contribution

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 [6] 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 [17] 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 [16] 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. 4, 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. 5, 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.

2. 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 [14] pre-trained on StylishHumans-HQ Dataset [3] for fashion video experiments, and on FFHQ [7] 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 [11] 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 [10] 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

$$\mathbf{z}_{dT} = \phi_T(\mathbf{z}_{d0}) = \mathbf{z}_{d0} + \int_0^T f_\theta(\mathbf{z}_{dt}, t)\, dt, \tag{1}$$

Where $z_{d0} = z_d$. We parameterize f_θ as a convolutional network obtained from [10]. 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 [13] 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 $G(\cdot)$.

$$\mathbf{z}_{\text{Style}} = \text{CLIP}_I(G(\mathbf{z}_C)) + \alpha(\text{CLIP}_T(\mathcal{D}_{\text{TGT}}) - \text{CLIP}_T(\mathcal{D}_{\text{SRC}})) \tag{2}$$

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:

$$\mathbf{z}_t = \mathbf{z}_c + \Delta\mathbf{z} \tag{3}$$

The output frame at time t is then generated as

$$\mathbf{X}_t = G(\mathbf{z}_t) \tag{4}$$

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.

3. Details on the Datasets & Evaluations

Datasets. All our results were evaluated on the Fashion dataset [6] and the RAVDESS dataset [9]. 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.

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)* [15] 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)* [4] 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 [5]. *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 [13] as a zero-shot classifier for this task.

Baselines. We trained HairCLIP [19] 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 [18] on the Fashion dataset, we used the StyleGAN-Human [3] pre-trained generator. For RAVDESS, we used the FFHQ pre-trained generator for all 3 models. For DiCoMoGAN [6], we trained the official code until convergence.

For MRAA [16] 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.

4. Further Quantitative Results

4.1. 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, 2, 12] 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. 1 the performances of VidStyleODE on Fashion Videos, where a generator and an inversion network pre-trained on Stylish-Humans-HQ

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 1. **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.

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 2. 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.

Dataset [3] 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 [8] and fine-tuned on RAVDESS [9] 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.

4.2. Further Ablation Studies

In Sec. 4.1 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 of the first frame, a random frame, or the mean latent code. Tab. 2 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).

5. 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.

5.1. Latent motion representation

Our model is able to learn a meaningful latent space for motion, which enables multiple applications. As seen in Fig. 1, 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. 2 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 [16] are found in Figs. 4 and 5.

5.2. Fashion Videos dataset

- Fig. 1 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. 2, 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. 3 provides qualitative comparisons for our VidStyleODE model to the existing methods.
- Fig. 4 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. 5 provides a second comparison of image animation methods.
- Fig. 6 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. 7 and 8 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. 9 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. 10, 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.

5.3. RAVDESS dataset

- In Fig. 11, 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. 12 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. 13, we show that our VidStyleODE method can also animate still frames via extrapolation.
- Finally, in Fig. 14, 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.

References

[1] Rameen Abdal, Peihao Zhu, Niloy J. Mitra, and Peter Wonka. Video2stylegan: Disentangling local and global variations in a video, 2022. 3

[2] Gereon Fox, Ayush Tewari, Mohamed Elgharib, and Christian Theobalt. Stylevideogan: A temporal generative model using a pretrained stylegan, 2021. 3

[3] 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, 3, 4

[4] 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. 3

[5] 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. 3

[6] 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

[7] 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. 2

[8] 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. 4

[9] 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. 3, 4

[10] 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. 2

Figure 1. 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.

[11] 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. 2

[12] 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. 3

[13] 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, 3

[14] 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

[15] Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. *ArXiv*, abs/1606.03498, 2016. 3

[16] Aliaksandr Siarohin, Oliver J. Woodford, Jian Ren, Menglei Chai, and Sergey Tulyakov. Motion representations for articulated animation. 2021. 2, 3, 4

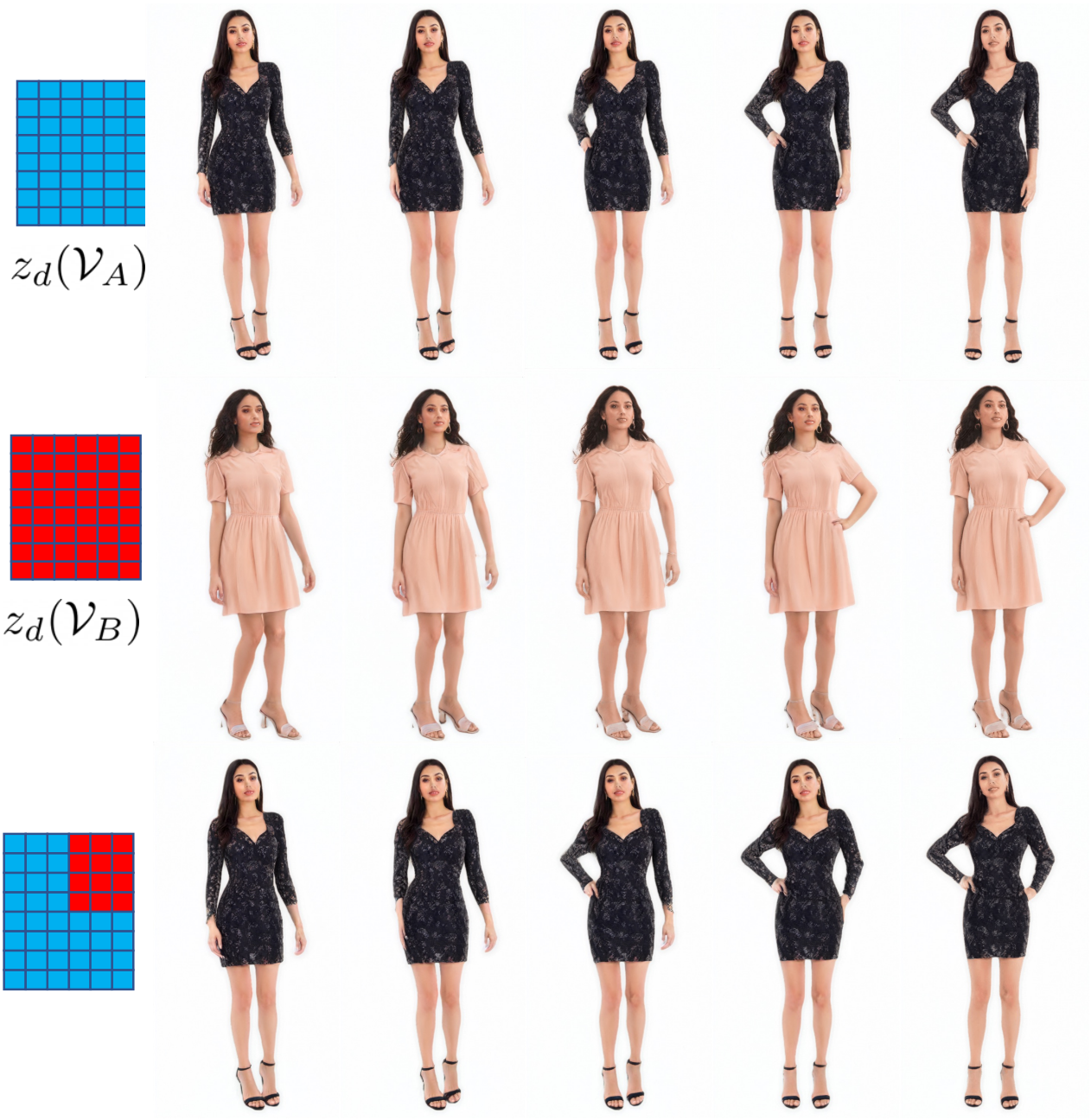

Figure 2. We transfer the upper right dynamics from $\mathcal{V}_B$ to $\mathcal{V}_A$, while keeping the rest of the dynamics from $\mathcal{V}_A$. This results in the right arm moving upwards, while the rest of the dynamics are unchanged.

[17] 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

[18] 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. 1, 3

[19] 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. 1, 3

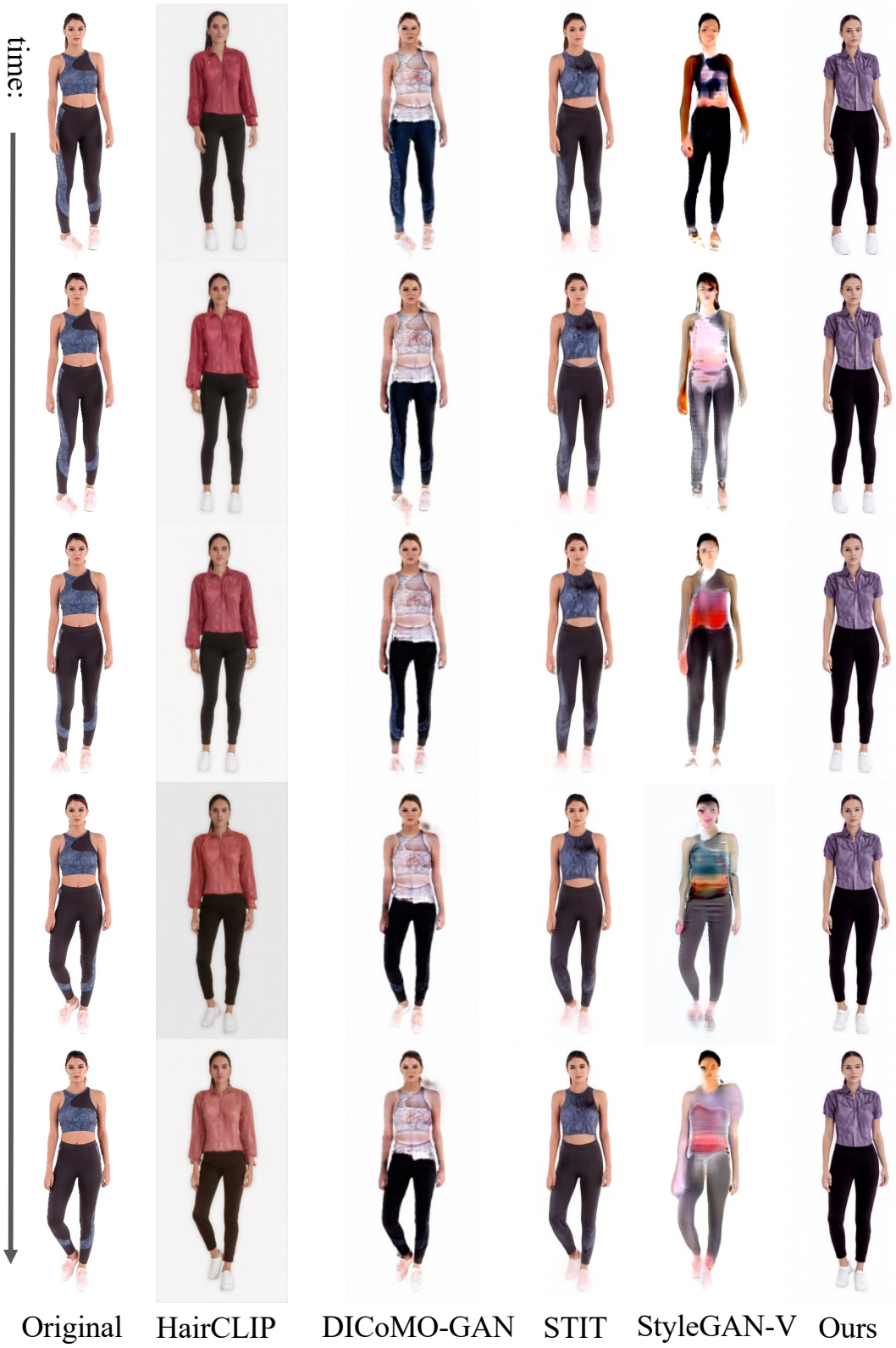

Figure 3. Here, we perform additional manipulations using the baselines. We exclude the latent transformer results since it is unable to perform complex manipulations without multiple steps. The source text is "*a photo of a woman wearing a crop top*", and the target text is "*a photo of a woman wearing a* ***blouse***".

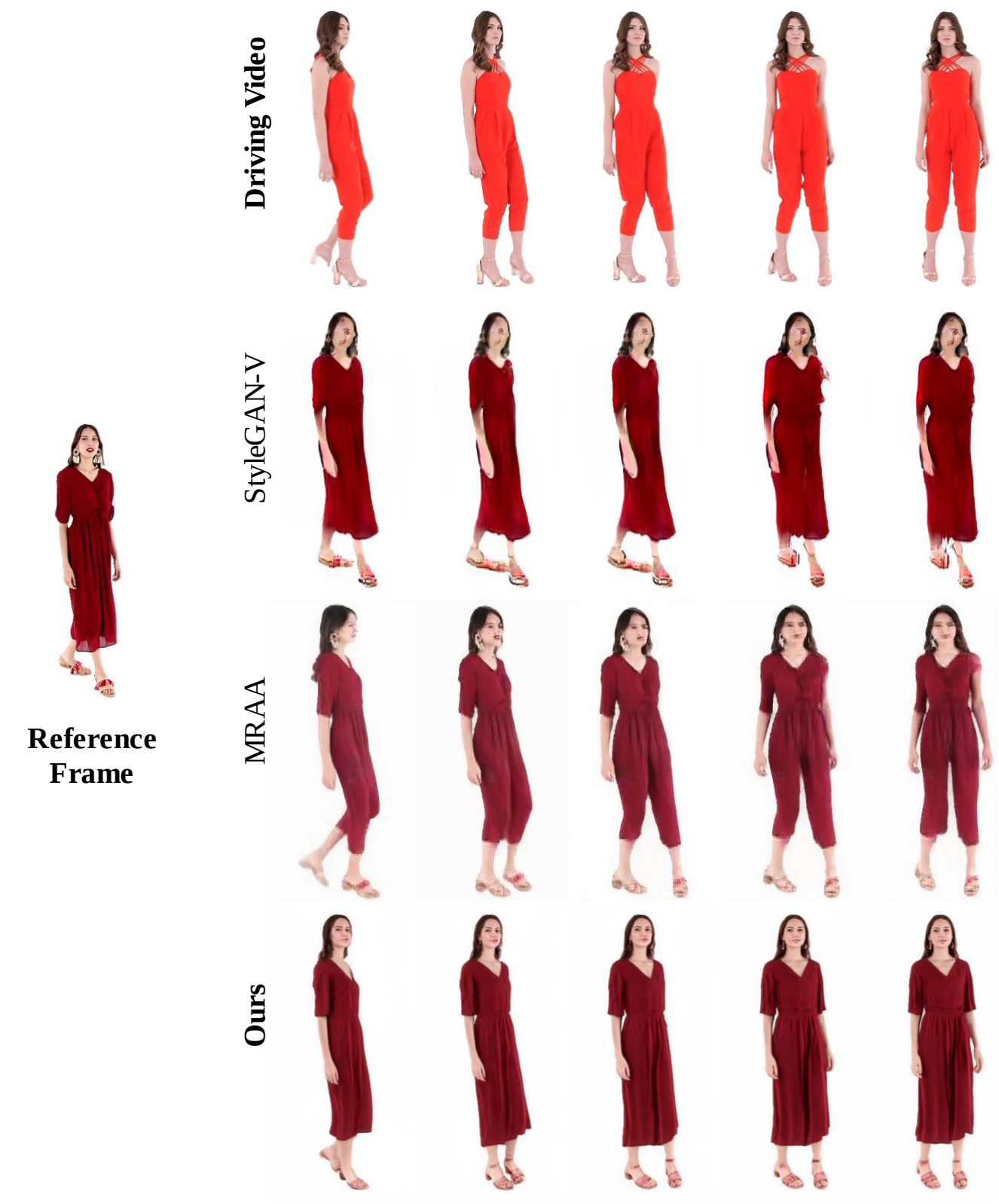

Figure 4. We perform image animation using our model and multiple baselines. We obtain the dynamics from the driving video in the first row and apply it to the reference frame to generate the videos. Our results obtain the highest perceptual quality, while also matching the dynamics of the driving video, 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.

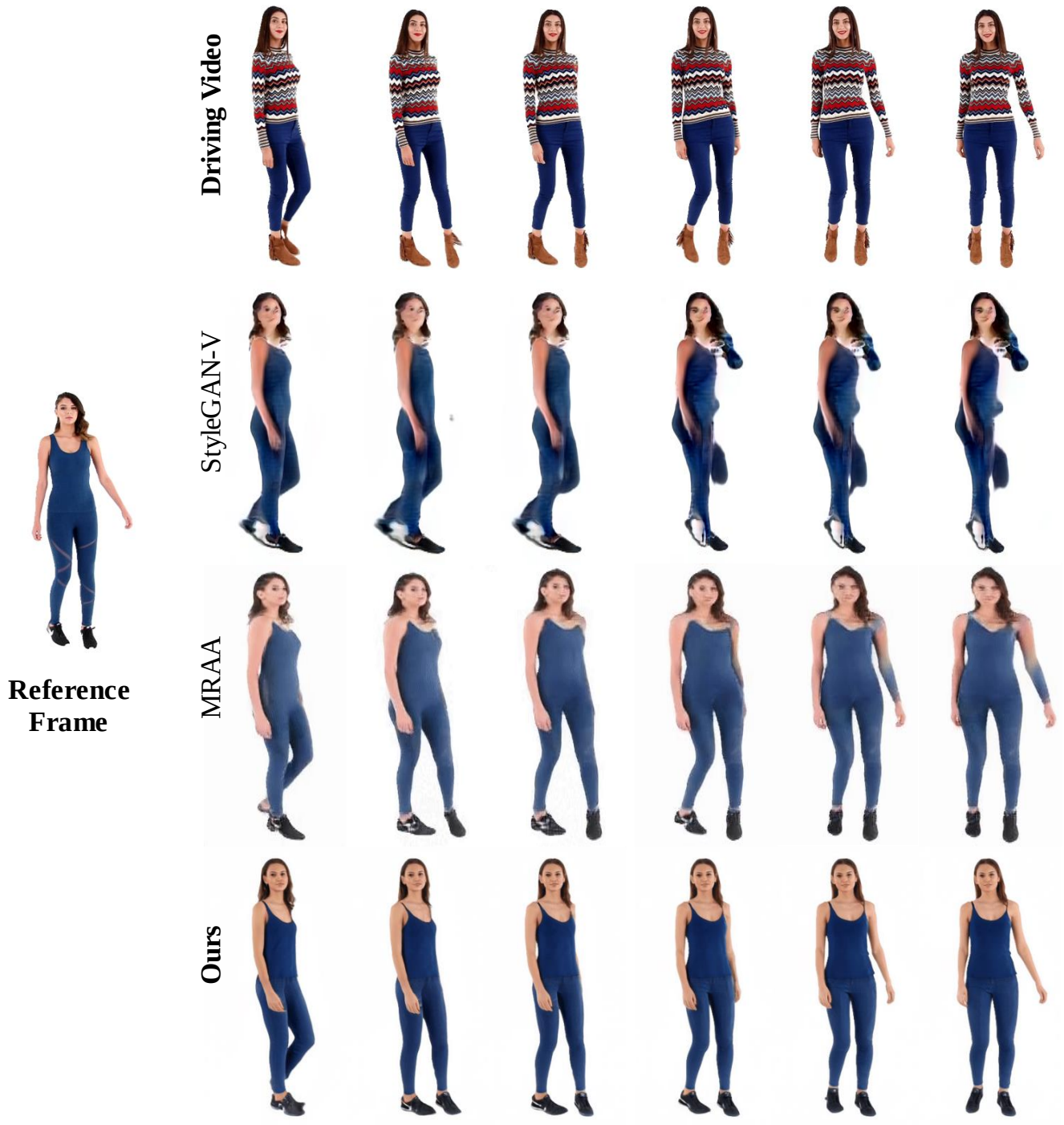

Figure 5. Another example on image animation. Our model produces the result with the highest perceptual quality while being consistent with the driving video and reference frame. MRAA has some inconsistencies in the arm, and StyleGAN-V is unable to capture the motion in any meaningful way.

Figure 6. 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.

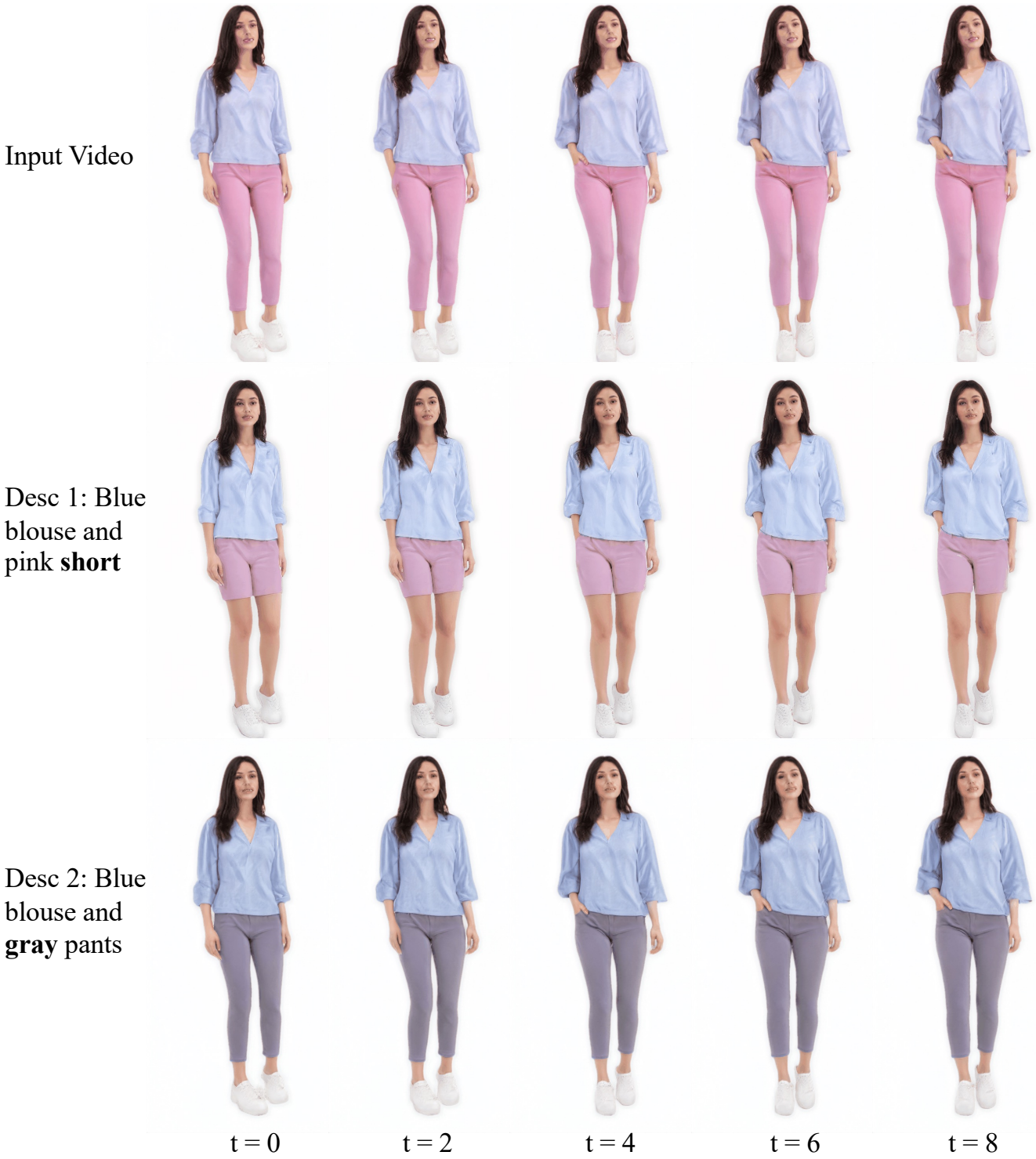

Figure 7. We perform two different manipulations to a sample video (the source video) from the Fashion Videos dataset and display the corresponding results here. Target 1 uses the target text "*A photo of a woman wearing blue blouse and pink **short***". Target 2 uses the target text "*A photo of a woman wearing blue blouse and **gray** pants*".

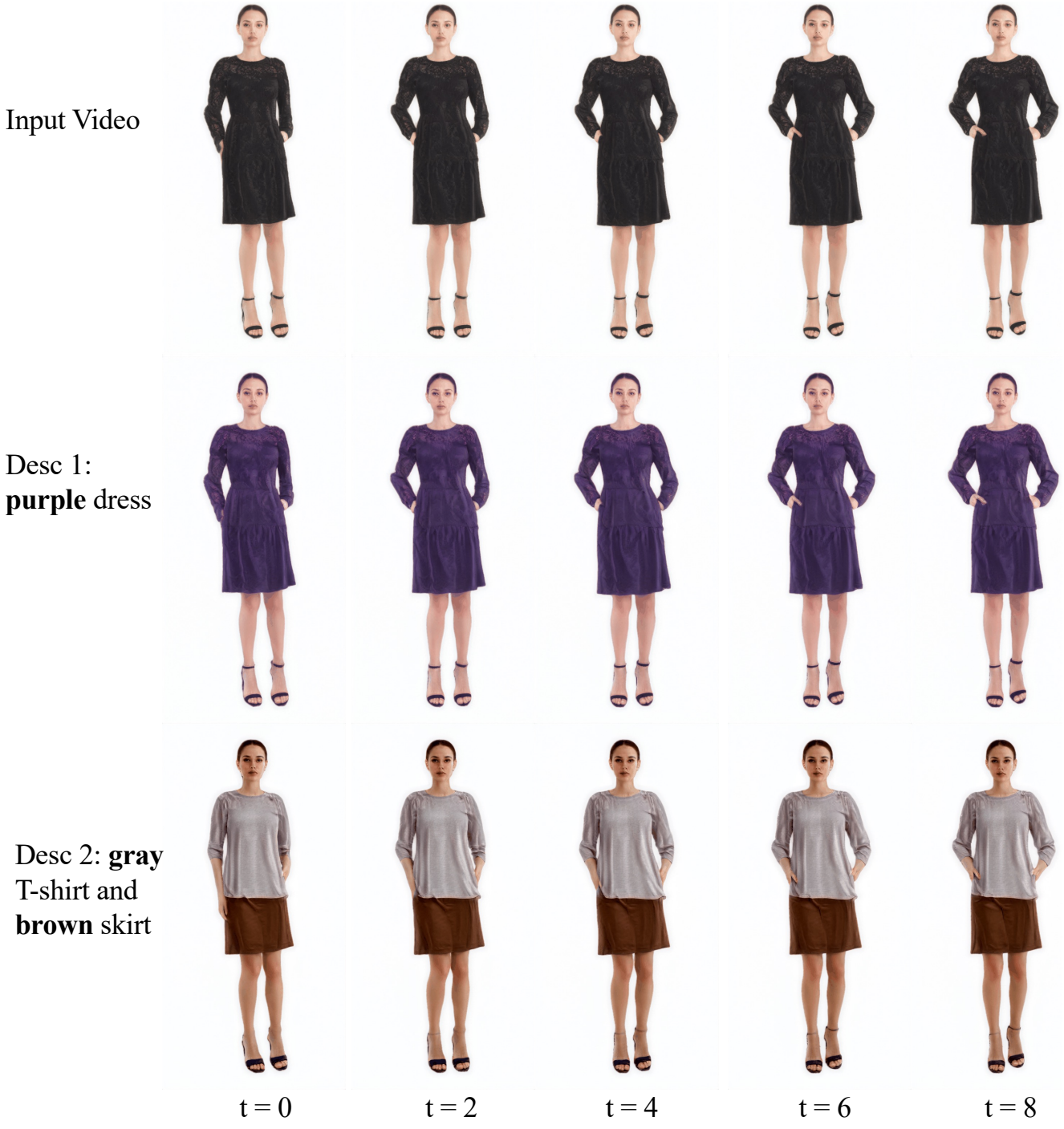

Figure 8. We perform two different manipulations to a sample video (the source video) from the Fashion Videos dataset and display the corresponding results here. Target 1 uses the target text "*A photo of a woman wearing a **purple** dress*". Target 2 uses the target text "*A photo of a woman wearing **gray** T-shirt and **brown** skirt.*".

Figure 9. 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).

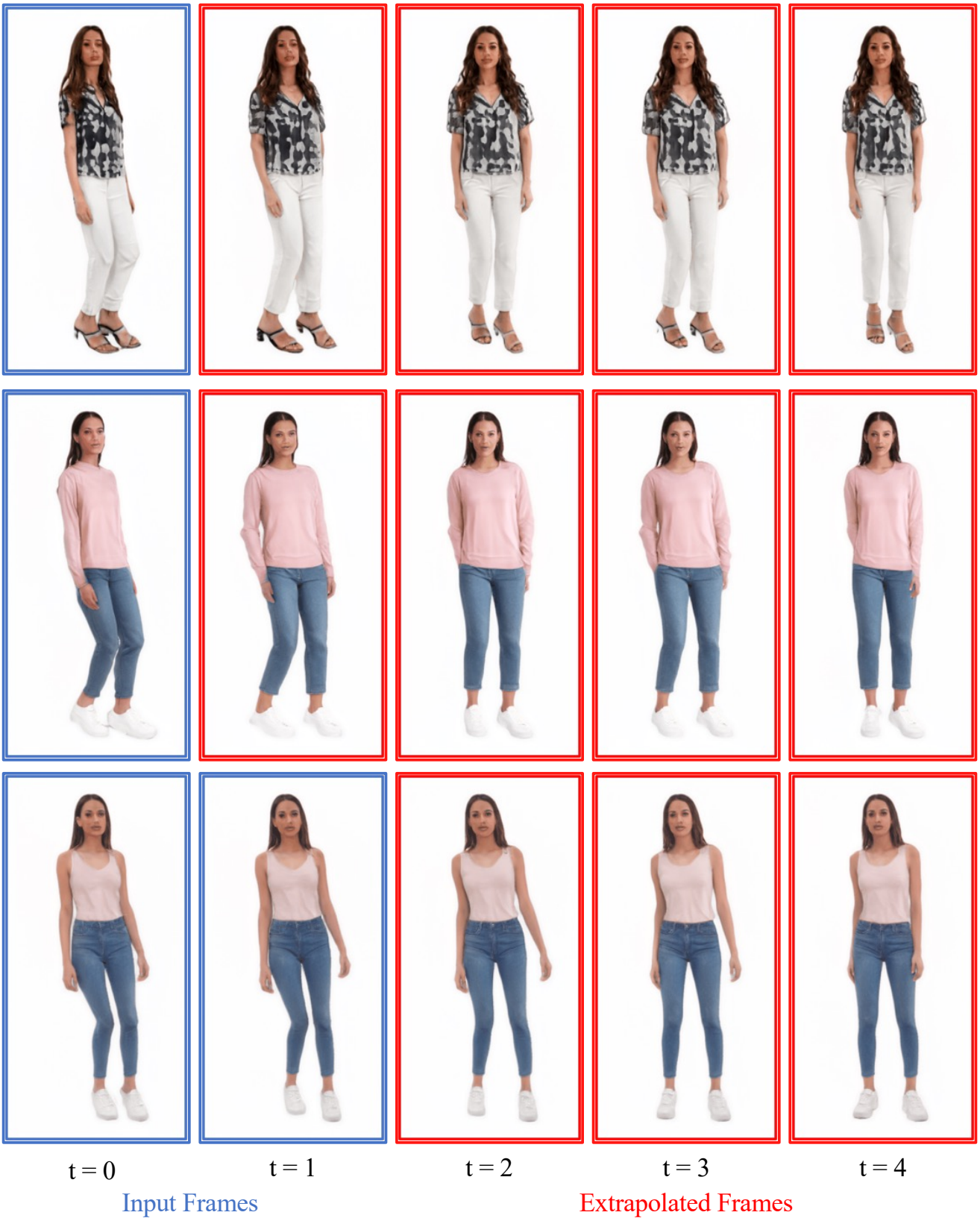

Figure 10. To perform extrapolation from a single frame, we provide just the initial frame (shown in blue), and then generate the next 4 frames (shown in red).

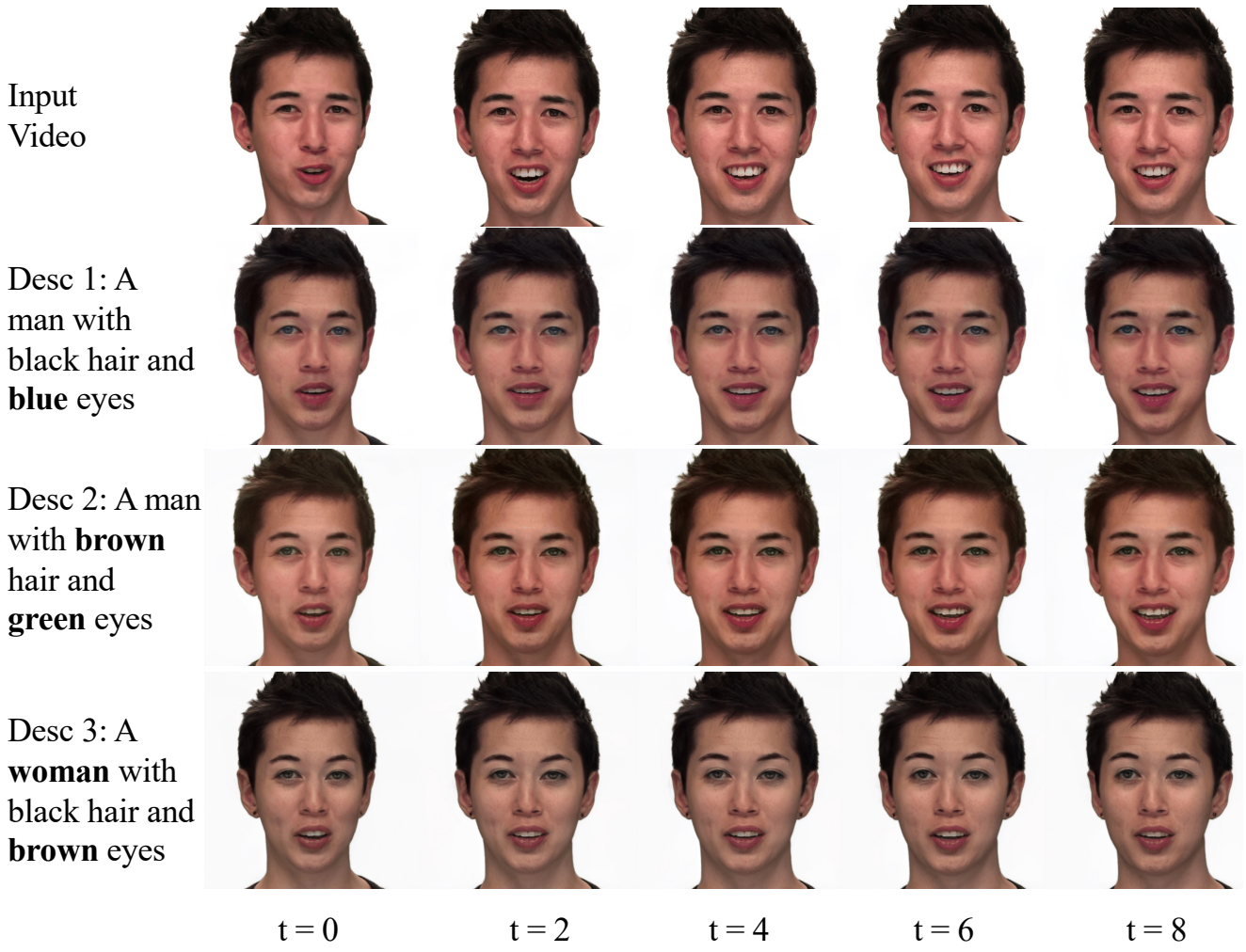

Figure 11. We perform three different manipulations to a sample video (the source video) from the RAVDESS dataset, and display the corresponding results here. Target 1 uses the target text "*A man with black hair and **blue** eyes*". Target 2 employs the target text "*A man with **brown** hair and **green** eyes.*". Target 3 uses the target text "*A **woman** with black hair and brown eyes.*".

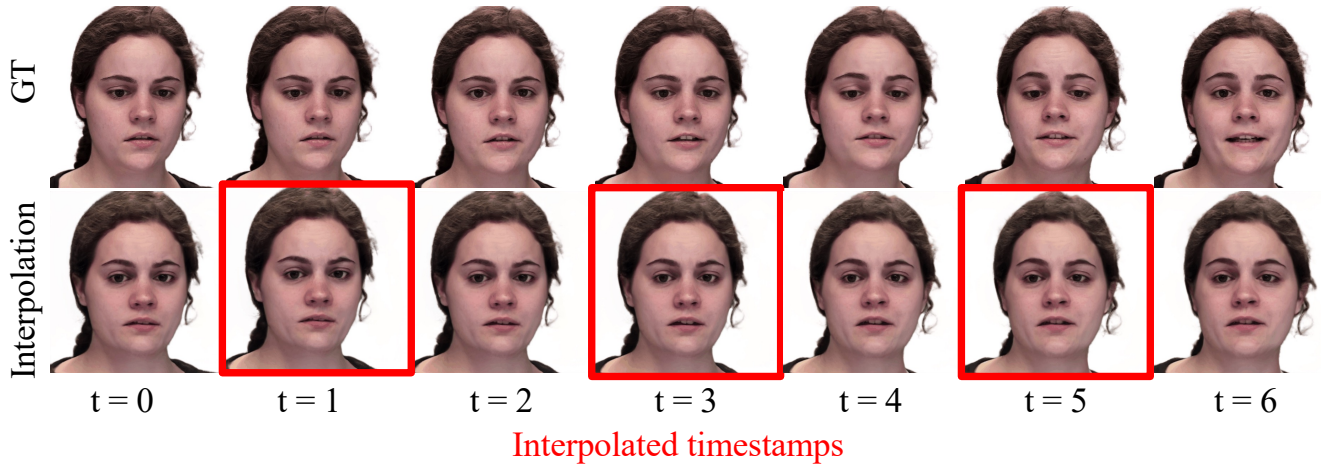

Figure 12. To perform interpolation, we provide four distinct frames with different timestamps ($t = 0$, $t = 2$, $t = 4$, and $t = 6$) (shown in blue) to the model, and then generate the unobserved frames for timestamps $t = 1$, $t = 3$, and $t = 5$ (shown in red).

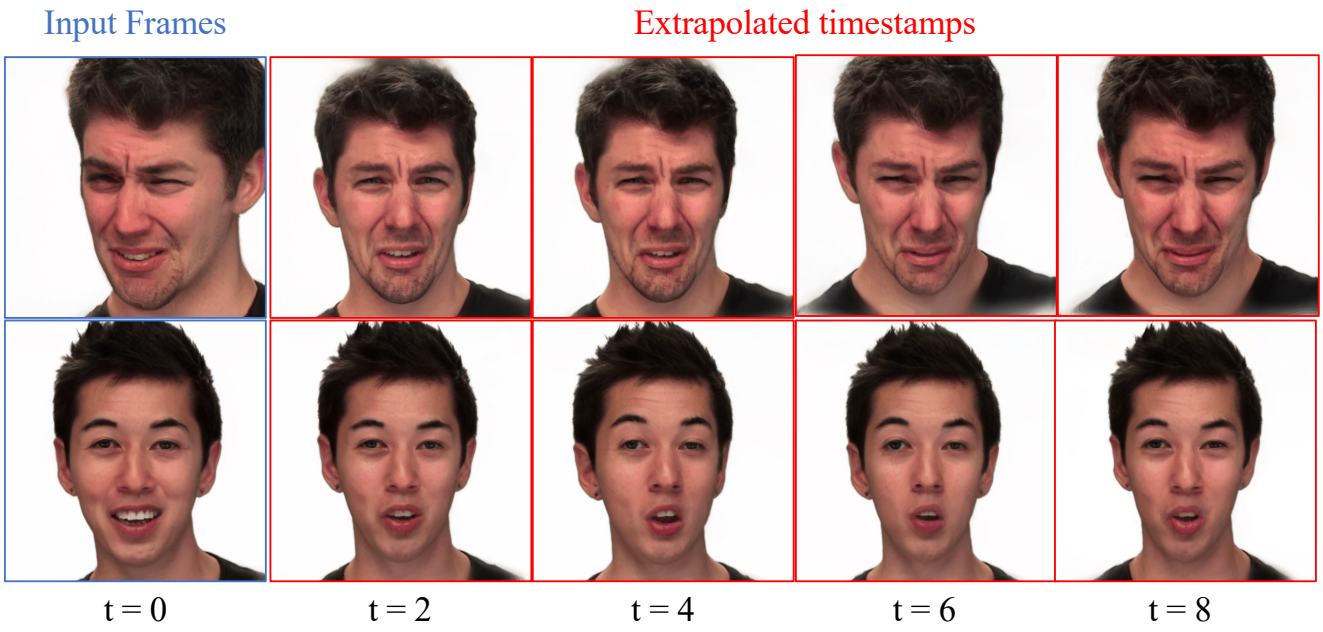

Figure 13. Extrapolation from a single frame: we provide just the initial frame (shown in blue), and then generate the next 4 frames (shown in red).

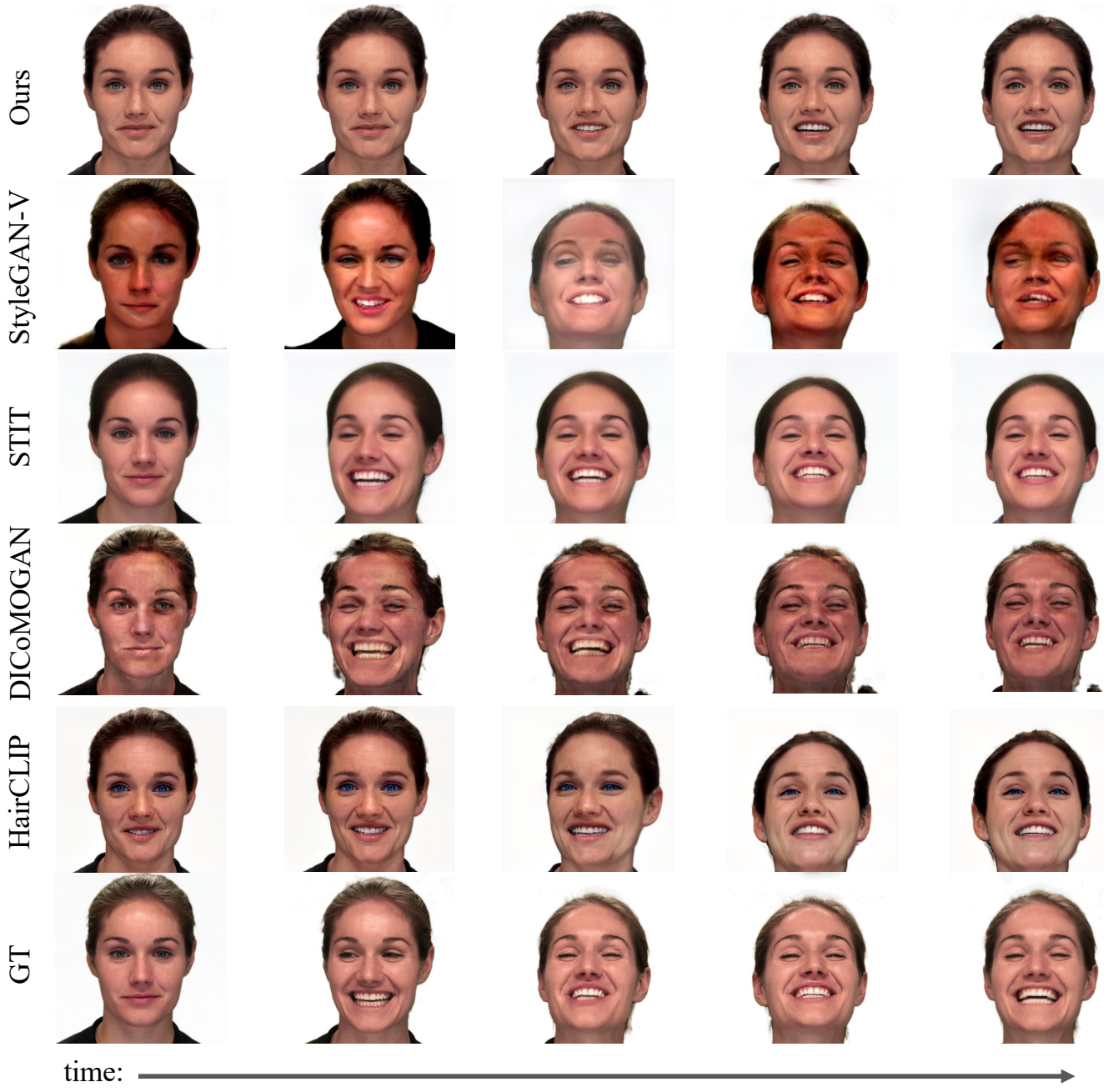

Figure 14. Qualitative results of our approach and the competing editing methods. The description of the source image is "*A woman with blond hair, and green eyes*", while the target description is specified as "*A woman with **brown** hair and **blue** eyes*".