VIVE3D: Viewpoint-Independent Video Editing using 3D-Aware GANs
SUPPLEMENTARY MATERIALS

Anna Frühstück[2], Nikolaos Sarafianos[1], Yuanlu Xu[1], Peter Wonka[2], Tony Tung[1]

[1] Meta Reality Labs Research, Sausalito [2] KAUST

afruehstueck.github.io/vive3D

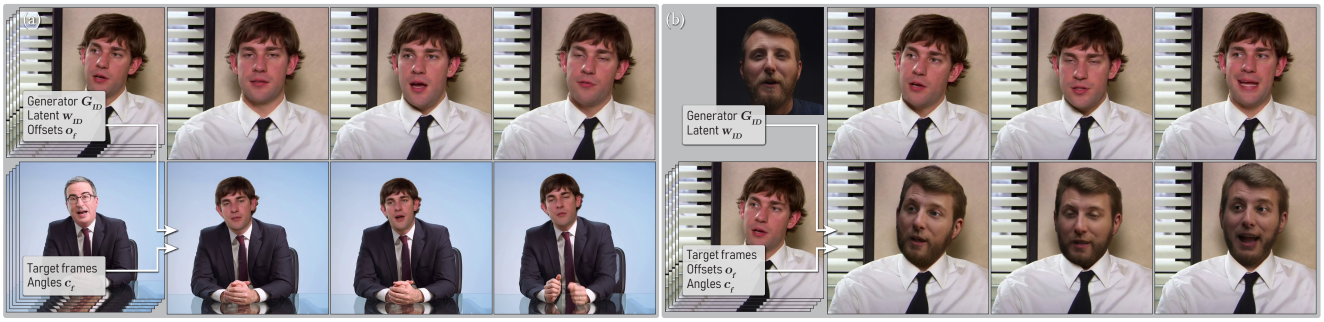


Figure 1. **VIVE3D generalization to new identities**. The benefits from decomposing the inversion of the input into an identity latent and a set of offsets unlock applications of face/motion re-targeting with minimal effort. This is possible due to our novel personalized generator that can be trained on a specific person's identity and then applied to edit an unseen video. We show two examples: In example **(a)**, we use a person *(top row)* to invert and fine-tune the generator, and we determine the video offsets based on this video sequence. The bottom row determines the target frames as well as the face location and angles. For example **(b)**, we use a personalized generator *(top left)*, but the target frames, angles, as well as motion, stem from a distinct video, driving the motion of the target person.

1. Additional Results

1.1. Supplementary Video

Please see our supplementary video (on the project webpage) for video sequences illustrating our proposed method and a set of results demonstrating the unique capabilities of our technique as well as comparisons to related methods.

1.2. Experimental Edits

We showcase some additional experimental edits to illustrate the generalization abilities of our approach for general-purpose video editing. For example, we are able to use two completely disjoint videos of different subjects and achieve reasonable results at compositing them. We show in Figure 1 two instances of such applications: On the left (Figure 1 (a)), we use one personalized Generator with its "default" person latent $\mathbf{w}_{ID}$, and a stack of video offsets encoding a sequence of face motions. We then compose these with a different body by running our inset optimization, using the target video frames and head angles from that particular video. On the right (Figure 1 (b)), we use the encoded face motion from the target video, projecting the motion onto a different person's face, thereby essentially replacing the head in the target video. Note that in order to achieve plausible results for these instances, we need to copy head and neck due to the slight differences in lighting between the source and target faces. Further, the segmentation masks need to be considered carefully and be big enough, *e.g.* in the right result, the hair sticking out from the source person's head needs to be covered by inpainting with background color during the inset optimization in order to achieve a reasonable result. Otherwise, the optimization will add extra hair and change the hairstyle. We observe that VIVE3D generates realistic results of placing the first person's head on the second person's body that are temporally and spatially consistent and follow the target frame motion. This is possible because of our proposed design (personalized generator, separate identity, and offset latents) which makes VIVE3D unique compared to prior work. It is worth noting that when the source and target videos have different light conditions then the results might have different lighting between the body and the face. This is because we do not explicitly tackle this problem in our architecture and hence lighting is baked in the final generated/edited head before it is placed on top of the new body. We identify the problem of better lighting transfer as an avenue for future work.

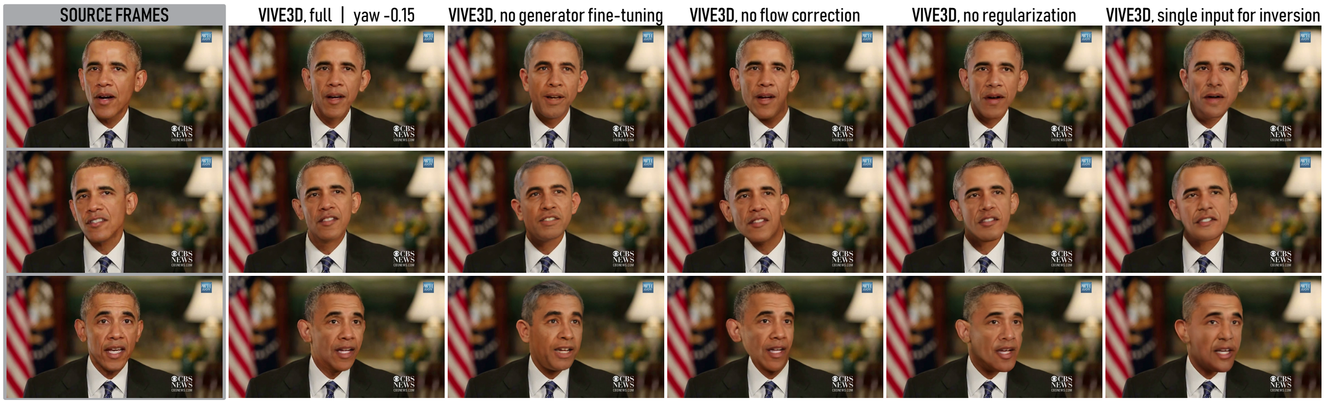

SOURCE FRAMES
VIVE3D, full | yaw -0.15
VIVE3D, no generator fine-tuning
VIVE3D, no flow correction
VIVE3D, no regularization
VIVE3D, single input for inversion

Figure 2. **VIVE3D Ablation Study.** Our proposed approach VIVE3D *(2^{nd} column)* with all the proposed components demonstrates better identity preservation, fixes the spatial misalignment between the face and neck when rendered from a novel view and better captures the fine-level details of the face and results in high-fidelity faithful renders of the person from new views.

1.3. Ablation Study

To evaluate the impact of each module we conduct ablation studies and report our quantitative and qualitative results in Table 4 of the main paper and Figure 2 of the supplementary respectively. Given a video of a person talking *(1^{st} column)* , we demonstrate our complete approach when rendering the output video from a new viewpoint *(1^{st} column)*. In the next 4 columns of results, we strip one component at a time and observe different performance quality drops. For example, if we do not fine-tune the generator it is clear that the identity of the individual is not properly preserved *(3^{rd} column)*. If we remove the flow correction module which is a key contribution of our approach, we observe that the face and the neck are not well aligned which makes the results seem unnatural *(4^{th} column)*. The impact of the flow correction module is demonstrated also in Figure 3 and discussed in detail in the supplementary video. If we strip the regularization *(5^{th} column)*, we remove the joint latent $\mathbf{w}_{ID}$ and treat each target face in the initial inversion separately. This means we don't constrain the individual latents to stay close to the common latent. We can see that this leads to a deterioration in the inversion quality of the video frames and produced artifacts, as the inverted latents no longer share information, and there is no constraint on the projected location in latent space. Finally, if we were to only perform single-frame inversion rather than multi-frame, we also observe a significant drop in fidelity *(6^{th} column)*, which indicates that the proposed approach of performing multi-frame fidelity is beneficial as it better captures the identity and the fine details of the face.

1.4. Qualitative Results of Method Comparisons

We provide a qualitative comparison to related methods of GAN-based video editing, Stitch it in Time (StiiT) [12] and VideoEditGAN (VEG) [13]. First, we discuss some of the differences between our proposed method and the related work to establish the parameters of our comparison.

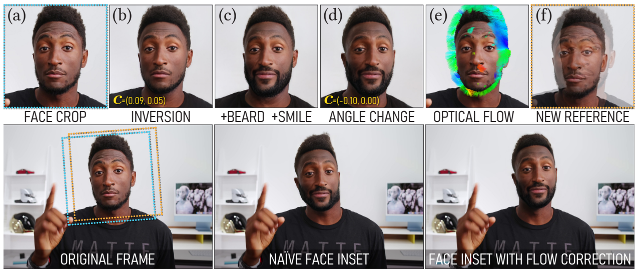


Figure 3. **View Adjustment Additional Results.** We illustrate the problem arising when attempting to composite a changed view of a person's head on top of the original body. After cropping **(a)** and inversion **(b)**, we perform face editing **(c)** and change the camera viewpoint to an unseen angle **(d)**. Replacing the face in the original frame with this edit yields poor quality *(bottom center)* even for small angular changes because the rotated face is in the wrong location with respect to the body. We address this by estimating the optical flow **(e)** between the face crop and the edit and use the flow direction to correct the location of the reference face based on the prospective inset **(f)**. This allows us to composite the edited face into the frame in a natural-looking fashion *(bottom right)*.

- StiiT and VEG, which is closely related to StiiT, both use a StyleGAN2 backbone which outputs high quality images at 1024×1024px resolution, whereas our backbone's (EG3D) output resolution is 512×512px (obtained by super-resolution given 128×128px inputs), providing 3D-awareness at the expense of slightly inferior image quality to classic StyleGAN2. In order to compare quantitatively, we downsample all results generated by StiiT and VEG to 512×512px, unless we compare at the full video resolution, in which case the output of the respective generator is already resampled to fit the resolution of the original face crop in the video frame.

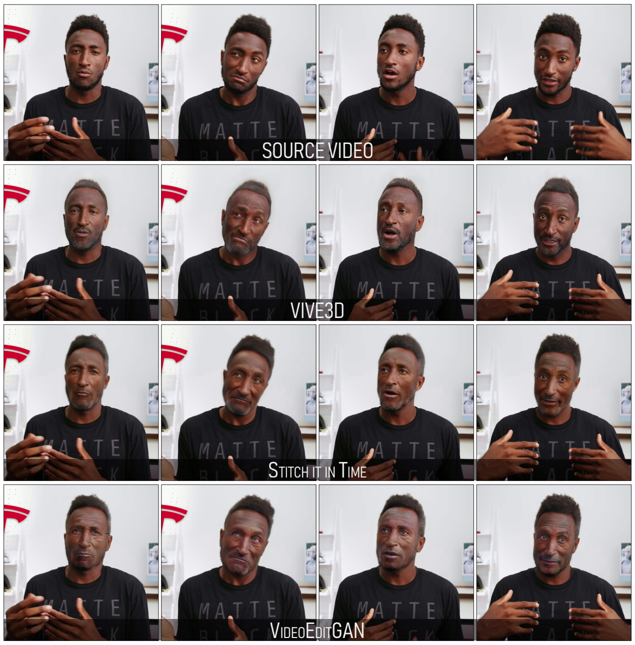


Figure 4. **Comparison to related work on age editing.** Due to the distinctly different InterfaceGAN editing directions, we hand-picked the edit strength for the respective latent space edits to showcase a similar effect in aging the target person. Our technique yields results that are at least on par with the previous StyleGAN2-based editing techniques *(bottom two rows)*.

- StiiT and VEG rely on prior work for a reliable encoding framework, e4e [10], to yield good and coherent inversions, whereas we implemented an optimization strategy to obtain per-frame inversions. Implementing an encoding strategy for EG3D was outside the scope of our project, but would be an interesting topic for future endeavors. We expect that using an encoder would lower the embedding quality, but improve computation speeds.
- StiiT and VEG fine-tune their generator on all video frames simultaneously, thus achieving very good coherence to the input. In contrast, we fine-tune on a select few target faces, yielding a more generalizable generator, which, in consequence, is not optimized to replicate the video frame-by-frame.
- Like our approach, StiiT relies on InterfaceGAN [11] for discovering and applying latent space editing directions for many of their results. VEG shows their results using edits based on StyleClip [9]. To compare with their method, we adapted their code to also allow edits with InterfaceGAN – analogous to StiiT– before applying their temporal consistency strategy. Note that the discovered directions in StyleGAN2 and EG3D latent space do not yield identical results for the same attribute type and the strength needed to apply the direction vectors is different. We empirically chose weights to approximate the same edit strength when comparing results.

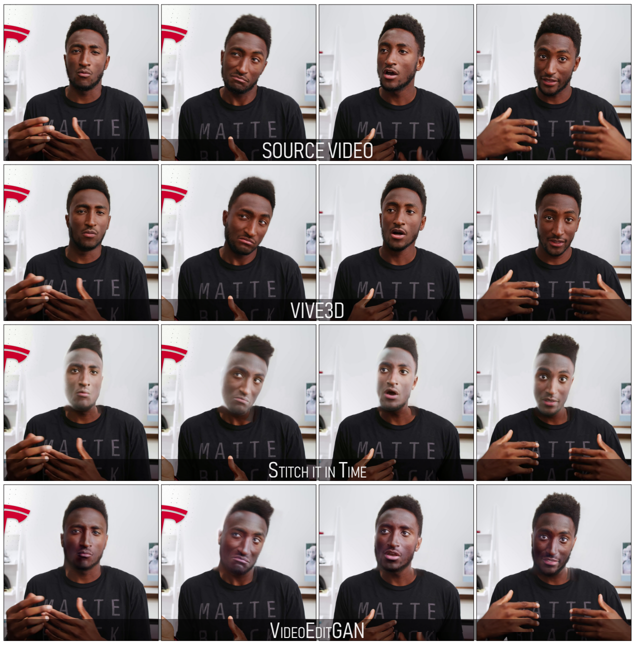


Figure 5. **Comparison to related work on angle editing.** A slight change in angle is achieved for the related methods by applying a yaw-changing latent space direction. Both methods fail at producing a reasonable composition given these edits *(bottom two rows)*. We implement a similar angle change and demonstrate that our method creates a natural-looking composition.

We demonstrate in Figure 4 that all methods provide plausible results for classic semantic editing problems such as aging the target person. However, both related methods fail to yield plausible results for angle editing. In order to compare this task, we utilize a latent space direction discovered in StyleGAN2 that allows for slight angle changes. The comparison for these strategies is illustrated in Figure 5. The artifacts present in these qualitative results mirror the deterioration in quantitative scores indicated in Table 2 in the main paper.

In Figure 6, we show a result of our multi-target inversion strategy (row **((a))**) compared to another single-image

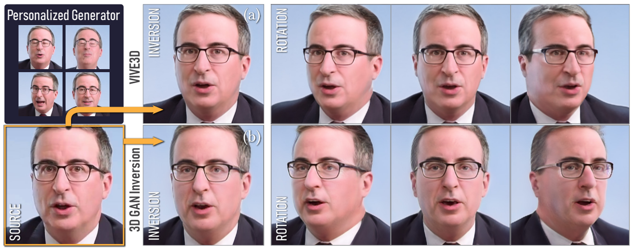


Figure 6. **Comparison to another 3D inversion technique.** To demonstrate the effectiveness of our Personalized Generator inversion, we compare the quality of our inversion with a recent technique of 3D GAN inversion [7]. Note how the head shape and identity correspondence deteriorate when rotating the head away from the original pose.

Table 1. **Reconstruction Metrics.** We compare the quality of our inversion with StiiT using reconstruction metrics on a subset of videos. We also evaluate the Fréchet Inception Distance (FID) of inversion and edits with respect to the source video.

		Reconstruction Quality		**Editing Quality**		
		PSNR ↑	**SSIM ↑**	**Fréchet Inception Distance (FID) ↓**		
	Method	INVERSION		INVERSION	AGE EDIT	ANGLE EDIT
Marques	StiiT	**36.477**	0.965	10.11	18.44	21.58
	VIVE3D	33.791	**0.987**	**7.25**	**12.73**	**11.63**
Obama	StiiT	34.969	**0.976**	**3.80**	16.49	17.12
	VIVE3D	**36.282**	0.969	3.88	**8.67**	**7.22**
Dennis	StiiT	40.708	**0.993**	6.32	12.88	16.96
	VIVE3D	**40.804**	0.990	**4.07**	**8.36**	**8.07**

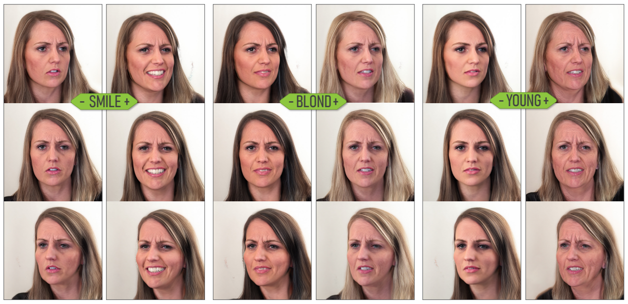


Figure 7. **InterfaceGAN Edits Additional Results.** We show InterfaceGAN editing directions discovered in the latent space of the pretrained 3D GAN by applying them to our personalized generator. The attribute edits are plausible and consistent in 3D.

3D GAN inversion [7] (row (**(b)**)). Please zoom in to observe the degradation of head shape and loss of identity when the head rotation diverges from the source image.

1.5. Additional Quantitative and Qualitative Results

We provide some additional quantitative metrics on individual videos shown throughout this supplementary material. In Table 1, we analyze the reconstruction quality and editing quality for three individual videos, showing that our reconstruction and editing capabilities are on par with our main competitor technique StiiT for video inversion and editing tasks. We also analyze the inversion and editing performance of VIVE3D and related methods for two distinct videos in more detail in Table 2. We use the ArcFace [4] metric to calculate the minimum, maximum, and average similarity to the source video as well as the temporal difference by evaluating the metric on adjacent video frames. The quantitative scores show that our method is superior for both attribute and angular edits, the latter being a task at which the previous 2D-GAN-based methods fail.

We showcase some supplemental qualitative results that demonstrate in further examples that VIVE3D is able to (1) apply existing latent space editing techniques such as InterfaceGAN [11] to generate natural-looking results (Figure 7) with performance comparable to previous 2D techniques, (2) create plausible image compositions for diverging from the original head angle, generalizing to various camera viewpoints given a source frame (Figure 8), (3) generate high-quality results that are temporally consistent for combined angle and attribute editing (Figure 9), and (4) synthesize spatially consistent results (Figure 10) even for challenging boundary cases.

Table 2. **Face Similarity Metrics.** We evaluate the identity preservation of inversion and edits based on the cosine similarity of ArcFace features extracted from generated face crops with respect to the face crops of the source video. To evaluate coherence over time, we measure the dissimilarity between consecutive frames.

			Similarity to Source ↑			**Temporal Diff ↓**		
		Method	MIN	MAX	MEAN	MIN	MAX	MEAN
Marques	Inversion	e4e	0.487	0.816	0.663	0.1	36.3	5.6
		StiiT	0.720	0.877	0.820	0.1	8.6	**1.3**
		VIVE3D	0.820	0.932	**0.894**	0.1	7.5	1.4
	Age Edit	StiiT	0.720	0.877	0.820	0.1	8.8	1.3
		VEG	0.430	0.668	0.551	0.1	11.5	1.9
		VIVE3D	0.730	0.923	**0.857**	0.1	5.2	**1.1**
	Angle Edit	StiiT	0.654	0.801	0.740	0.1	8.9	1.4
		VEG	0.424	0.685	0.568	0.1	12.1	1.7
		VIVE3D	0.762	0.899	**0.849**	0.1	8.1	**1.1**
Obama	Inversion	e4e	0.469	0.801	0.665	0.1	44.8	5.9
		StiiT	0.935	0.982	**0.968**	0.0	7.6	1.0
		VIVE3D	0.882	0.961	0.930	0.0	2.1	**0.5**
	Age Edit	StiiT	0.717	0.862	0.781	0.1	6.1	1.0
		VEG	0.522	0.763	0.671	0.2	7.1	1.5
		VIVE3D	0.758	0.903	**0.850**	0.1	4.0	**0.8**
	Angle Edit	StiiT	0.668	0.827	0.753	0.0	7.8	1.3
		VEG	0.771	0.874	0.840	1.0	5.9	1.2
		VIVE3D	0.782	0.916	**0.868**	0.1	4.2	**0.9**

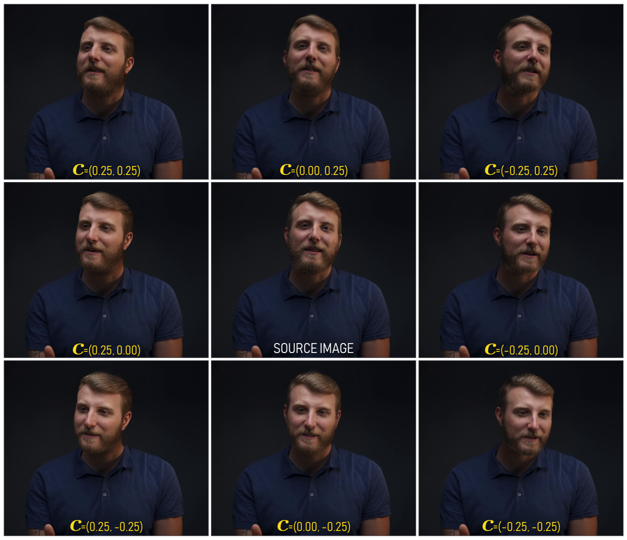
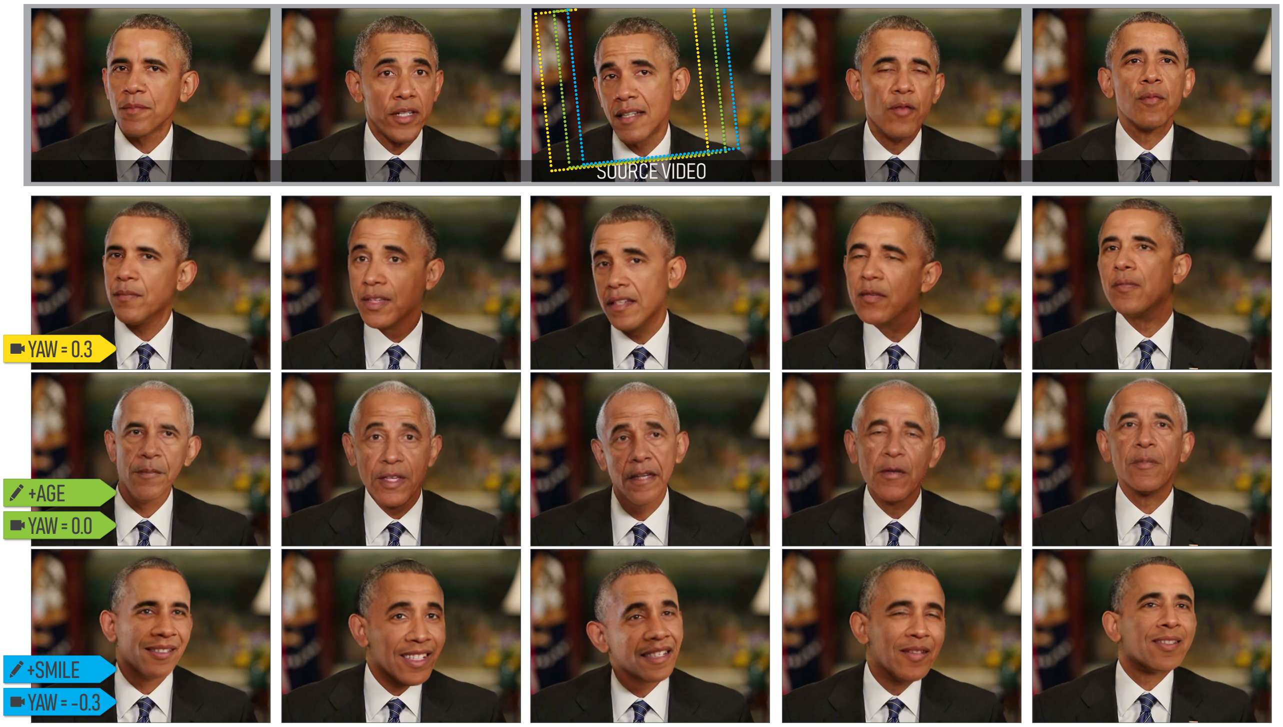


Figure 8. **Changing Camera Poses Additional Results.** Our method can freely alter the camera pose and composite the result back with the source frame by fixing the divergence between the source and target pose using our optical flow correction. The generated results look natural despite the static body pose.

Figure 9. **Additional Qualitative Results.** Given a video sequence, we process individual frames, cropping the face region to correspond with our generator's field of view. VIVE3D faithfully modifies several facial attributes as well as the camera viewpoint of the head crop. Finally, we seamlessly composite the edited face with the source frame in a temporally and spatially consistent manner, while retaining a plausible composition with the static components of the frame outside of the generator's region. The dotted squares in the center frame denote the reference regions for the three different camera poses in the column below.



2. Optimization Details and Parameter Settings

Our approach is implemented in Python 3.8 and uses PyTorch. We build our approach on the pretrained models and the publicly available codebase of EG3D [2, 3]. During our pipeline, we propose several optimization steps. Each of them is relying on ADAM as an optimizer. We run all experiments on a single NVIDIA A100 GPU and provide timings and hyperparameters for our various pipeline steps.

1. **Generator Inversion**: For the initial inversion of the generator, we use a standard learning rate scheduler. We also ramp down the regularization weight of $\mathbf{o}_f$ from λ_{wdist} to λ_{wdist_target}.
   Optimization hyperparameters
   λ_{L1} = 0.05, λ_{face} = 1.0, λ_{LPIPS} = 0.75, λ_{wdist} = 0.05, λ_{wdist_target} = 0.005, `initial_learning_rate = 1e-2`, `num_steps = 600`
   Duration 3 min 33 sec (5 target images)
2. **Generator Fine-Tuning**: We fine-tune the weights of the StyleGAN2 backbone of EG3D as well as the neural renderer, leaving learned weights of the Upsampling module untouched.
   Optimization hyperparameters
   λ_{L1} = 1.0, λ_{LPIPS} = 0.3, `learning_rate = 1e-3`, `num_steps = 300`
   Duration 3 min 27 sec (5 target images)
3. **Video Inversion**: During this optimization, we run a frame-per-frame inversion, starting from the average offset of all offsets $\mathbf{o}_f$ discovered in step 1. Using this strategy, we invert the first frame for `init_num_steps`. Each consecutive frame is started from the previous offset and optimized for `num_steps`. We provide an early stopping criterion `loss_threshold` and finish the current frame optimization in case the total loss falls below this threshold.
   Optimization hyperparameters
   λ_{L1} = 0.25, λ_{face} = 1.2, λ_{LPIPS} = 1.0, λ_{wdist} = 0.01, `learning_rate = 1e-2`, `loss_threshold = 0.25`, `init_num_steps = 200`, `num_steps = 50`
   Duration 19 sec (first frame), ~4 sec/frame (consecutive frames)
4. **Optical Flow Evaluation**: During this step, we evaluate the flow between the source face crop and the (angle-edited) target face to estimate the correction of the source

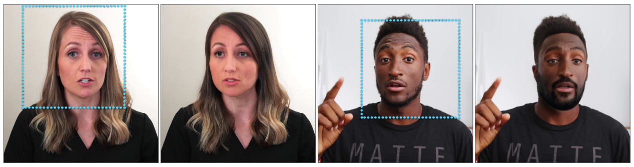


Figure 10. **Spatial Consistency Additional Results.** VIVE3D composites images with challenging boundaries such as long hair *(left)*, yielding faithful hair color change results. For hard boundary cases, such as matching with a static piece of hair outside the boundary crop *(right)*, it plausibly connects the contents of the two images.

crop needed to achieve a plausible inset composition.
To estimate the flow, we use Farnebäck optical flow with the following parameters: `pyr_scale = 0.5`, `levels = 8`, `winsize = 25`, `iterations = 7`, `poly_n = 5`, `poly_sigma = 1.2`
Duration ~0.4 sec/frame

5. **Inset Optimization**: In this optimization step, we can specify sizes `border_size` for the width of the segmentation boundary region that is optimized and `edge_size` to provide an offset distance for the border from the image boundary. We can again specify an early stopping criterion `border_loss_threshold` to stop when the border loss falls under this threshold, which in practice provides significant speedup.
Optimization hyperparameters
`weight_foreground = 1.0`, `weight_border = 2.0`, `edge_size = 50`, `border_size = 50`, `num_steps = 150`, `learning_rate = 1e-2`, `border_loss_threshold = 0.05`
Duration ~4 sec/frame (~16 sec/frame w/o early stopping)

3. Social Impact

The ability to provide editability/customization in videos of humans has been an active area of research over the past few years. On one hand, it can have key applications in providing people the ability to express themselves in different ways (*e.g.*, the ability to change hair color, add glasses, etc) during video calls, or more broadly in how they interact in the digital world. At the same time, such techniques introduce use cases for potentially malicious use that are worth discussing. For example, the ability to replace someone's face in a video from another person resembles deepfakes and could be used by bad actors. While the results such as what is shown in Figure 1 are still not at the level that would be perceived as indistinguishable from an original video this is an important conversation to be had regardless. We encourage the interested reader to refer to concurrent work [1, 5, 6, 8] for deep fake detection to discover edited videos.

References

[1] Shruti Agarwal, Hany Farid, Yuming Gu, Mingming He, Koki Nagano, and Hao Li. Protecting world leaders against deep fakes. In *CVPR workshops*, volume 1, page 38, 2019. 6

[2] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *CVPR*, 2022. 5

[3] EG3D Codebase. *https://github.com/NVlabs/eg3d*. 5

[4] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face recognition. In *CVPR*, 2019. 4

[5] Brian Dolhansky, Joanna Bitton, Ben Pflaum, Jikuo Lu, Russ Howes, Menglin Wang, and Cristian Canton Ferrer. The deepfake detection challenge (dfdc) dataset. *arXiv preprint arXiv:2006.07397*, 2020. 6

[6] Luca Guarnera, Oliver Giudice, and Sebastiano Battiato. Deepfake detection by analyzing convolutional traces. In *CVPR Workshops*, 2020. 6

[7] Jaehoon Ko, Kyusun Cho, Daewon Choi, Kwangrok Ryoo, and Seungryong Kim. 3D GAN inversion with pose optimization. In *WACV*, 2023. 3, 4

[8] Thanh Thi Nguyen, Quoc Viet Hung Nguyen, Dung Tien Nguyen, Duc Thanh Nguyen, Thien Huynh-The, Saeid Nahavandi, Thanh Tam Nguyen, Quoc-Viet Pham, and Cuong M Nguyen. Deep learning for deepfakes creation and detection: A survey. *Computer Vision and Image Understanding*, 223:103525, 2022. 6

[9] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or, and Dani Lischinski. StyleClip: Text-driven manipulation of StyleGAN imagery. In *ICCV*, 2021. 3

[10] 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. In *CVPR*, 2021. 3

[11] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. InterFaceGAN: Interpreting the disentangled face representation learned by GANs. *TPAMI*, 2020. 3, 4

[12] Rotem Tzaban, Ron Mokady, Rinon Gal, Amit H. Bermano, and Daniel Cohen-Or. Stitch it in time: GAN-Based facial editing of real videos. In *SIGGRAPH Asia*, 2022. 2

[13] Yiran Xu, Badour AlBahar, and Jia-Bin Huang. Temporally consistent semantic video editing. In *ECCV*, 2022. 2