

From 2D Portraits to 3D Realities: Advancing GAN Inversion for Enhanced Image Synthesis

Anonymous CVPR submission

Paper ID 17

Abstract

Image synthesis using StyleGAN has shown remarkable results in 2D portrait image generation. The works of the GAN inversion to manipulate the real image using StyleGAN latent space also show remarkable achievements. 2D GAN inversion has successfully manipulated global attributes such as facial expressions and gender. However, preserving the hairstyle and identity was difficult according to the pose change. We introduce the 3D GAN inversion encoder to make a high-resolution 3D image based on the Geometry Aware 3D Generative Adversarial Network, known as EG3D, which allows explicit control over the pose of the real image subject with multi-view consistency. Our network projects the single 2D portrait images to novel latent space for 3D GAN inversion for the tri-plane of EG3D. We also present multi-view cycle loss, which aims to increase multi-view consistency. By leveraging the new latent space and loss for 3D GAN inversion, our network can successfully convert 2D portrait images into 3D fast.

1. Introduction

Generative Adversarial Networks (GANs) have been actively studied thanks to their ability to synthesize images of high visual quality and diversity. In particular, StyleGANs [18–20] have shown that they effectively encode semantic information in latent space beyond phenomenal results and fidelity in numerous areas of generating and manipulating 2D portrait images. An operation using these characteristics showed promising results in the synthetic image generated by StyleGAN. The GAN inversion is required to apply the operation to the real image. The GAN inversion method converts a specified physical image into a latent space of a pre-trained StyleGAN. This GAN Inversion method has suggested \mathcal{W} and its extension, $\mathcal{W}+$ [1] space, depending on the shape of the target latent space for converting.

Previous studies have mapped a target real image to the corresponding latent code through learning-based methods



Figure 1. Given the desired input image, our framework learned how to find appropriate latent code to achieve accurate image reconstruction in the latent space of a 3D GAN. That way, real images can be quickly converted into 3D images.

[3, 28, 34] and optimization-based methods [1, 19, 29]. In terms of reconstruction accuracy, optimization-based methods can efficiently perform much better in inverting images without additional configuration, but they are time-consuming and challenging to utilize for applications. Learning-based methods train encoders to converge targets to suitable latent space using only a single forward pass,

043 so they are faster, but their visual quality is inferior to
044 optimization-based methods. In addition, the quality will
045 vary depending on the shape of the latent space in which
046 the encoder will reverse the image.

047 These GAN inversion methods showed good performance
048 for 2D GAN. However, as GAN expands, 3D image
049 generation also shows phenomenal results. Specifically,
050 EG3D [5] utilizes tri-plane presentation to create high-
051 definition 3D composite images with fewer parameters. It
052 also demonstrates that the optimization-based method [29]
053 works well for the StyleGAN2-based backbone in EG3D,
054 enabling real-world images to be converted to 3D. However,
055 designing an appropriate training scheme and latent space to
056 convert 2D images to 3D GAN remains challenging.

057 In this paper, we introduce a novel framework for
058 controllable Image-to-Image 3D translation which is based on
059 geometry-aware 3D Generative Adversarial Network. Our
060 framework converts the actual 2D image using 3D GAN so
061 that it can create the appropriate tri-plane feature for the
062 render who creates the 3D image. This model allows the pull
063 of style vectors with different scales. Our model utilizes a
064 novel latent space, \mathcal{W}_{tri} so that the learning-based method's
065 encoder can convert real-world images to 3D GAN fea-
066 ture maps, which have different characteristics from con-
067 ventional 2D GANs. The tri-plane feature maps for ren-
068 dering consist of three orthogonal planes, and the encoder cre-
069 ates a latent code that allows the real-world image to focus
070 separately on each plane through the \mathcal{W}_{tri} space. Finally,
071 the encoder for the super-resolution module is isolated to
072 extract latent code for high-resolution results. The results
073 of 3D GAN differ from those of 2D GAN, which expresses
074 only the visible area because rendering synthesizes multi-
075 view images. Accordingly, if the encoder is trained only
076 with a loss for 2D image restoration like conventional meth-
077 ods, information on the invisible image area for 3D image
078 synthesis may be omitted. Therefore, we trained the en-
079 coder by adding a new loss function so that we can extract
080 features for multi-view images. This loss function allows
081 the characteristics of the target image to be well expressed
082 even when the target image is converted to another view-
083 point.

084 Our main contributions are summarized as follows:

- We introduce novel latent space \mathcal{W}_{tri} and a framework that could make 3D portrait image by learning-based GAN inversion.
- We trained the encoder with new loss functions to create multi-view image characteristics.

090 2. Related Work

091 2.1. GAN Inversion.

092 With the development of GAN, many studies have emerged
093 to control latent space. Many recent papers have used Style-

094 GANs [18–20] because of its efficient image performance
095 and semantic abundance of latent space. GAN Inversion is
096 a method of extracting the image by pulling out the latent
097 vector and putting it in the generator. GAN Inversion refers
098 to creating a real-world image through latent manipulation
099 by locating the image as a pre-trained GAN latent space in
100 the latent code. In general, there are two methods of inver-
101 sion. First, the optimization-based method directly tunes
102 the parameter of the generator to express the target image.
103 It exhibits high-quality results without additional learnable
104 parameters, but it takes a long time. I2S [1] embeds the im-
105 age into the extension latent space $\mathcal{W}+$ of StyleGAN. The
106 author of StyleGAN2 [19] proposed a method of optimizing
107 the noise map corresponding to each layer of the synthesis
108 network and latent code. PTI [29] performs pivotal tuning
109 in the latent space for real images. It finds the latent code
110 representing the image, then fixes it and fine-tunes the gen-
111 erator to derive the results. Second, learning-based methods
112 train encoders that create latent codes that can represent real
113 images. Although there is a time advantage because a single
114 inference produces results, it usually results in poor visual
115 quality compared to optimization methods. pSp [28] and
116 e4e [34] directly extract the late vector of $\mathcal{W}+$ space using
117 the encoder network. It extended the GAN inversion prob-
118 lem to the image-to-image translation problem by changing
119 the input image because only the encoder network needs to
120 be learned. Another study, Restyle [3], also improves visual
121 quality by allowing encoders to modify latent codes repeat-
122 edly. Recently, the studies [4, 9] using a hypernetwork that
123 allows the addition of offset to the generator parameter have
124 revealed promising results.

125 2.2. 3D-aware GAN.

126 Extending 2D generative adversarial networks to 3D set-
127 tings has also begun to gain momentum. It has evolved
128 from mesh-based approaches [22, 33] to voxel-based GAN
129 [10, 15, 24, 25, 36], which directly extends CNN genera-
130 tors in 2D settings to 3D. However, it is difficult to adapt
131 to high-resolution 3D GAN training due to the high mem-
132 ory requirements of the voxel grid and the computational
133 burden of 3D convolution. Typically, there is NeRF [23] in
134 which position and direction are added to the 2D image to-
135 gether to make the result. NeRF, which is the content of this
136 neural implicit presentation, performs positional encoding
137 using a fully connected layer and provides a new position
138 view as a result. The 3D-aware GAN using this method in-
139 cludes StyleNeRF [13] and CIPS-3D [38]. However, Neural
140 implicit representations use fully connected layers with po-
141 sitional encoding, which can be slow to query. EG3D [5],
142 which we utilize as a 3D generator, does not use conven-
143 tional inefficient voxel grids. It uses hybrid tri-plane repre-
144 sentations because it scales quickly and efficiently with the
145 resolution. It provides greater details for equal capacity for

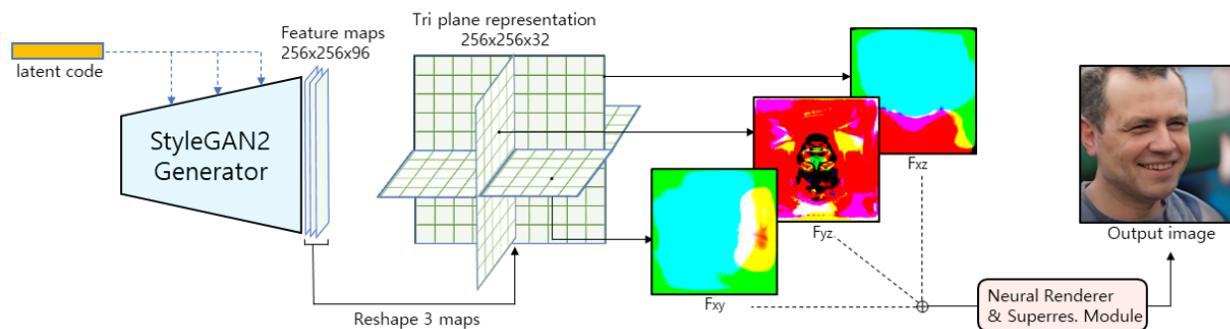
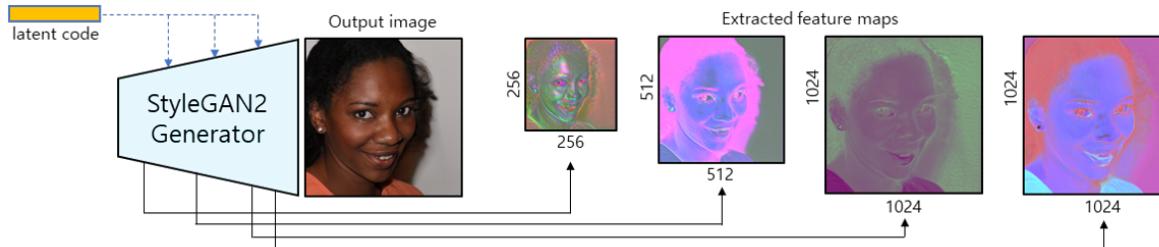


Figure 2. The above is the constructed feature maps with the generator of StyleGAN2 and the extracted feature maps. Below is the image of EG3D with a tri-plane presentation connected to the generator of StyleGAN2. In the case of the StyleGAN2 generator, it can be seen that the latent code gets closer to the output image, the more it becomes a face shape. On the other hand, it cannot be said that the feature map of EG3D's triplane has a proper face shape. This is because these feature maps must go through the neural render and Super-resolution module before the output image comes out.

146

creating high-quality 3D images.

147

2.3. Latent Space Manipulation

148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169

The field has recently seen a proliferation of studies that capitalize on StyleGAN's potential for semantic latent code editing. The intrinsic structure of StyleGAN's latent space is characterized by a high degree of disentanglement, which has inspired a multitude of methods to investigate semantic latent directions under different levels of supervision. Owing to its finely segregated latent space, StyleGAN has become the framework for these explorations. To uncover these semantic latent directions, the literature offers a panoply of methods. There exist fully-supervised approaches such as [2, 8, 12, 31] which make use of semantic labels, as well as unsupervised methodologies [14, 30, 35] that operate without such labels. Further, innovative techniques [11, 26, 35] have been introduced that utilize the Contrastive Language Image Pre-training (CLIP) model [27] to discover potential directions that facilitate new editing functionalities. However, these techniques have primarily been demonstrated on synthetic images generated within the confines of a pre-trained StyleGAN's latent space. For generative models that aspire to modify actual real-world imagery, it is imperative to have an effective inversion method that can accurately map these images into the gen-

170

erative latent space.

171

3. Method

172

3.1. Base 3D GAN

173

The existing 2D CNN-based generator model has limitations in modeling 3D images. This is because there is no information on position when a 2D single image is received as an input. When we visualize the feature map in the process of generating the 2D image as an example of StyleGAN2 [19], which we include in the generator, we can see that a low-resolution coarse-style image comes out at the beginning, and a fine image comes out toward the end. This means that as the input vector flows toward the end, the generator makes the feature map that is closer to the final image. Therefore, if the image is projected with \mathcal{W}^+ space using an encoder, the pose becomes fixed gradually during the generation process. To solve this problem, we mapped latent code into the EG3D [5] network, which enables 3D grounded neural rendering by transforming the feature map of StyleGAN2. However, when a 2D image is directly inverted through the encoder network to EG3D with this advantage, the desired form of the 3D image does not come out immediately. The reason is that EG3D reshaped the output feature map of the StyleGAN2 generator into a tri-plane

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

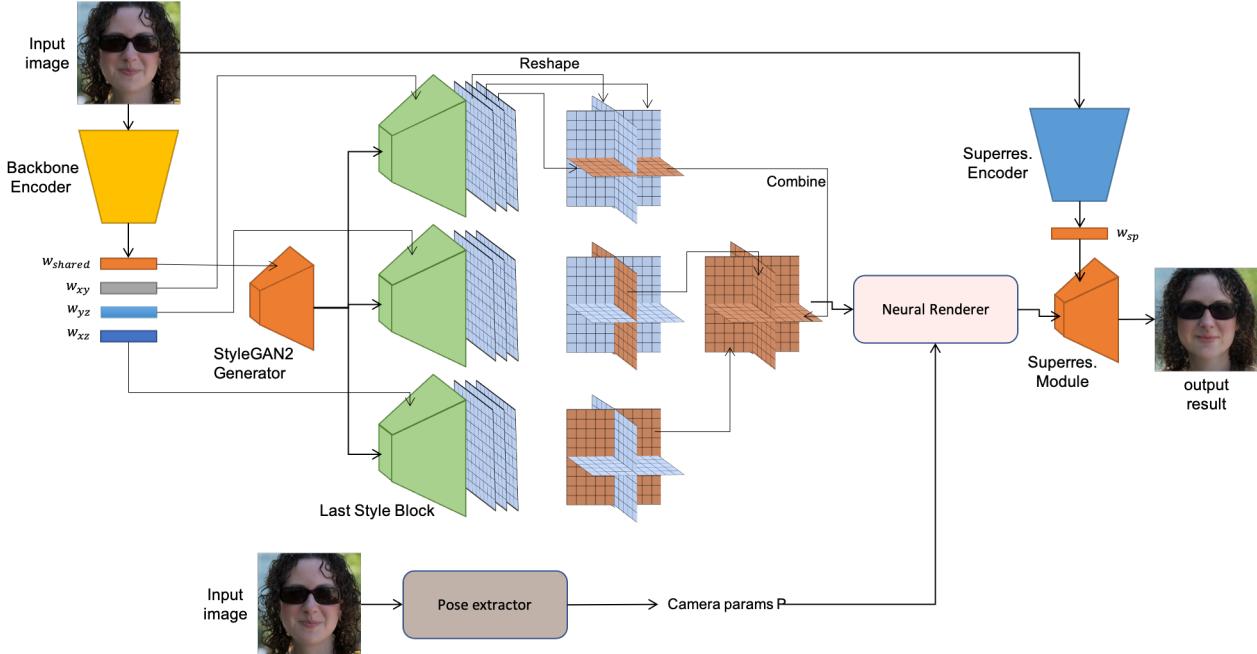


Figure 3. Our framework for converting single portrait photographs into 3D consists of a backbone encoder for neural renders and a super-resolution encoder for super-resolution modules. This architecture establishes a new feature map for rendering at each coordinate. By limiting the area the encoder needs to focus on, our framework can project real images into the latent space of the 3D GAN effectively and quickly.

representation by dividing it into three different output feature maps. We visualize the feature maps of 2D and 3D GAN in Figure 2. Visualization of the tri-plane’s three feature maps F_{xy} , F_{yz} , and F_{xz} with respect to the input image confirms that they are entirely different from the results of the 2D GAN. In 2D GAN, feature maps appear to represent the image’s shape itself, but 3D GAN does not simply represent it. This is because the process of neural rendering for 3D implementation is subsequently continued. Therefore, unlike projecting to 2D generator, we solve this problem by creating a new latent space called \mathcal{W}_{tri} .

3.2. \mathcal{W}_{tri} latent space & Encoder

As described above, the feature maps created by the backbone network of EG3D are distinctive. The EG3D divides the final output of the backbone network into three feature maps F_{xy} , F_{yz} , and F_{xz} on the channel axis, and the rendering module uses them to synthesize the image. The feature maps for rendering have different characteristics from feature maps in 2D GAN. To embed real images using the encoder, we propose \mathcal{W}_{tri} latent space that can echo the characteristics of the orthogonal plans of the tri-plane.

In order to embed the real image to latent space for fitting these feature maps, we use the latent code $w_{tri} = (w_{shared}, w_{xy}, w_{yz}, w_{xz}, w_{sp})$. The w_{shared} code is used as an input to the style blocks of styleGAN2, the backbone

network of EG3D. w_{xy} , w_{yz} , and w_{xz} codes are used to create F_{xy} , F_{yz} , and F_{xz} planes of tri-plane, respectively. We use w_{xy} , w_{yz} , and w_{xz} as inputs to the last style block in the backbone network of EG3D. From these three feature maps, we divide them into three by each channel axis for tri-planes, respectively. Extract F_{xy} , F_{yz} , and F_{xz} planes from each tri-plane and synthesize them to generate the final tri-plane feature maps for rendering. These feature maps will obtain refined orthogonal feature maps optimized for rendering.

EG3D uses the super-resolution module to obtain the conclusive high-resolution image from the raw rendering image. The w_{sp} code is used as an input to this super-resolution module. Unlike the latent codes of the backbone network, the w_{sp} code improves the image’s detail for the high-resolution image. This proposed latent space for 3D GAN embedding, which we call \mathcal{W}_{tri} space, provides the effect of converting a 2D image into 3D with free pose change.

We used Restyle [3] as the basis. Our encoder consists of a super-resolution encoder that makes w_{sp} code and a backbone encoder that makes the rest. The super-resolution encoder has the same FPN-based structure as the pSp [28], and the backbone encoder is also composed of an off-the-shelf structure from Restyle (see Figure 3).

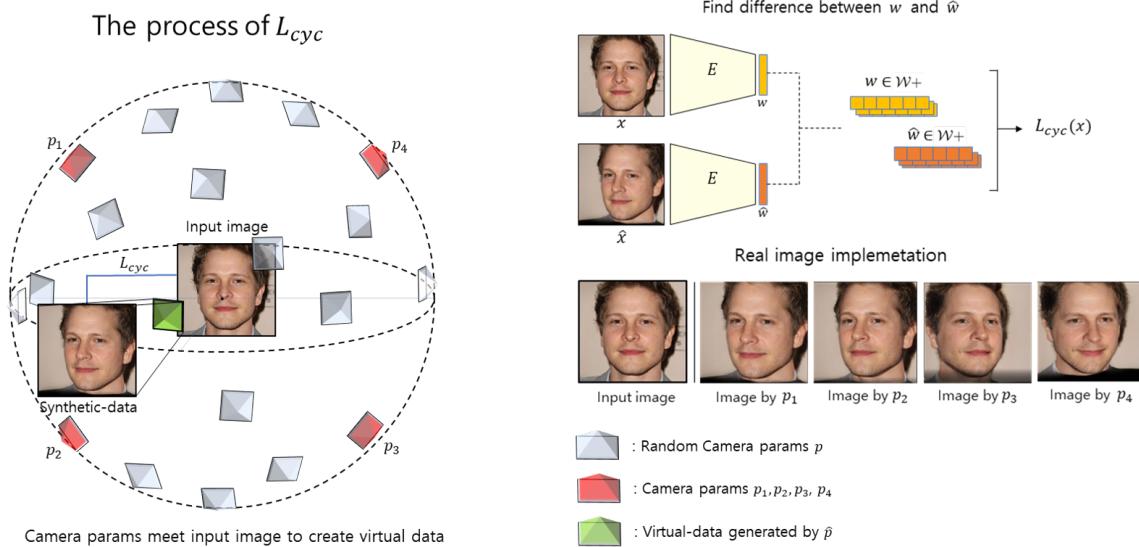


Figure 4. The role of \mathcal{L}_{cyc} included in a total loss is to create a virtual data creation effect by meeting the camera parameters with a new input image. This results are similar to learning data with pose information in various directions for learning data that lacks pose information. By passing the generated virtual data and input image through the encoder E , w and \hat{w} are obtained, as shown in the upper right figure. We defined the difference in the generated latent vector present in $\mathcal{W}+$ latent space as \mathcal{L}_{cyc} and trained in the direction of reducing it. The result of the implementation shows that camera parameters at different positions produce output images of various poses for the same input image.

243 3.3. Losses

244 We trained the encoder networks with popular loss in
 245 encoder-based methods. It employs a weighted combina-
 246 tion of a pixel-wise L2 loss, perceptual loss, identity loss
 247 [28] and similarity loss [34] with regularization loss. This
 248 loss objective is given by:

$$\begin{aligned} \mathcal{L}_{rec} = & \lambda_2 \mathcal{L}_2 + \lambda_{LPIPS} \mathcal{L}_{LPIPS} \\ & + \lambda_{id} \mathcal{L}_{id} + \lambda_{sim} \mathcal{L}_{sim} + \lambda_{reg} \mathcal{L}_{reg} \end{aligned} \quad (1)$$

$$\mathcal{L}_{reg}(x) = \|E(x) - \bar{w}\|_2 \quad (2)$$

254 where x means input image, p means camera pose parame-
 255 ter of x , G means a EG3D generator, E means out encoders,
 256 and θ implies weight.

257 In order to improve the quality of 3D images, we addi-
 258 tionally use pose cycle loss (Equation 3). The dataset of
 259 the real-world face has various poses for a person. How-
 260 ever, traditional losses in 2D GAN inversion tasks focus on
 261 2D image reconstruction. So, the training performance is
 262 insufficient. Since we use 3D GAN, we can change poses
 263 through neural rendering. This means that our encoder has
 264 to create the same codes from pose-changed images from
 265 the fixed latent code. Figure 4 shows our idea that is to

randomly pick one from multiple random camera parameters and calculate the cycle code loss. As a result, FFHQ or CelebA-HQ, a given 2D image dataset, lacks the poses of the image (person A does not exist in various poses). Therefore, it is possible to produce the same effect learned from data with various poses per each identity image.

$$\mathcal{L}_{cyc}(x) = \|E(x), E(G(E(x), \hat{p}))\|_1 \quad (3)$$

Finally, the total loss function that combines the above losses is as follows. Each of the lambda values is a hyper parameter that determines the loss weight. Learning can be regulated to the situation while easily changing values.

$$\mathcal{L}_{total} = \mathcal{L}_{rec} + \lambda_{cyc} \mathcal{L}_{cyc} \quad (4)$$

4. Experiments

4.1. Settings

Datasets. We perform an evaluation on a set of face image domains to demonstrate the approach. We use FFHQ datasets for training, and CelebA-HQ test sets for evalua-
 280 tion.

Baseline. For our approach, we employ the Restyle training process. Therefore, the w_{tri} latent code is updated repeatedly 5 times through the encoder. To generate the 3D

266
267
268
269
270
271

273
274
275
276

278

279

280
281
282
283

284
285
286

287 image, we use the pre-trained EG3D framework and fix
 288 all parameters of it. We use off-the-shelf face detection
 289 Deep3DFaceRecon [7] to extract a 16-size camera extrinsic
 290 parameter from test images. We use the same fixed 9-size
 291 camera intrinsic parameters from EG3D. For the random
 292 identity loss \mathcal{L}_{cyc} , we randomly extract a camera parame-
 293 ter from 120 aligned camera parameters and use it for each
 294 iteration.

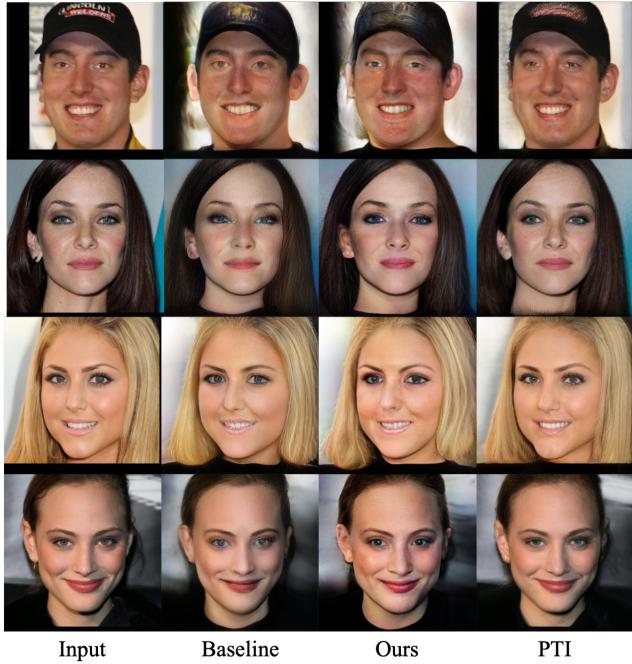


Figure 5. Inversion comparison results. Our approach makes the baseline Restyle [3] even better. However, the optimization-based PTI [29] shows better visual results. Zoom-in is recommended.

295 4.2. Comparison

296 We first compare our learning-based inversion work with
 297 the current state-of-the-art optimization-based technique
 298 PTI on EG3D. Although optimization methods for each
 299 input image exhibit excellent visual restoration quality
 300 results, there is a long computational time cost. In addition,
 301 we compare quality and identity preservation according
 302 to viewpoint changes with additional first-order-model
 303 [32] that are not GAN inversion methods but can create
 304 animations with single images. These results show that
 305 our framework effectively uses latent spaces suitable for
 306 multi-viewpoint 3D-aware synthetic image generation.
 307

308 **Qualitative Evaluation.** We start with ReStyle, the
 309 base model, and begin by showing a qualitative comparison
 310 of alternative inversion approaches. Figure 5 shows that
 311 our approach to new latent space and loss improves the
 312 quality of inversion. However, quality is inferior

313 to optimization-based methods. However, our model can
 314 achieve the 3D reconstruction quality with considerably
 315 low inference time. We also compare the 3D GAN model
 316 with how to warp the image to fit the given pose by utilizing
 317 the existing flow map. To generate the image of target
 318 poses, we use the average face of the pre-trained EG3D
 319 generator and add the camera poses to make different pose
 320 images for the same specimen. This method produces
 321 good results when there is a lot of video frame information
 322 enough to compute the flow map, as shown in Figure 6. It
 323 produces results that seem to have only failed to reflect the
 324 three-dimensional viewpoints where there will be too many
 325 pose changes. In addition, our model can produce results
 326 that fit the target pose much more accurately.

	Restyle	Ours	PTI
LPIPS ↓	0.25	0.22	0.10
ID ↑	0.35	0.36	0.57
sec/img	0.54	0.54	457.0

Table 1. Quantitative reconstruction results over the CelebA-HQ [17] test dataset.

327 **Quantitative Evaluation.** We compare Restyle baseline
 328 and PTI, the latest inversion techniques mentioned above,
 329 with our method for quantitative comparison. We apply
 330 the commonly-used LPIPS [37] metrics to measure
 331 reconstruction and the face recognition method [16] for
 332 identity similarity. For the training process of Restyle,
 333 we iteratively update \mathcal{W}_{tri} latent vectors five times. For
 334 optimization, we projected 500 steps for a single image to
 335 make latent vectors in \mathcal{W}^+ space. Meanwhile, for PTI, 350
 336 pivotal tuning steps are processed. In table 1, we show that
 337 our new latent space and loss contribute to the quantitative
 338 performance as well as the quality of the baseline. It also
 339 offers advantages in the application field because it can
 340 create 3D-aware synthetic images much faster, although it
 341 is less reproducible than optimization methods. We also
 342 showed performance improvement when our proposed
 343 method was applied sequentially. This result contains
 344 important ablations and gives us various quantitative results
 345 related to 3D inversion. As shown in the Table 2, \mathcal{W}_{tri} has
 346 strengths in LPIPS and \mathcal{L}_{cyc} strengthens ID consistency.
 347 To sum up, the result shows that the main contribution of
 348 our model, which is expressed by \mathcal{W}_{tri} and \mathcal{L}_{cyc} gives a
 349 meaningful difference between baseline and would have
 350 strengthened the result.
 351

352 5. Conclusions

353 In this paper, we introduced the 3D GAN inversion encoder
 354 to make a high-quality 3D-aware synthetic image using a
 355 single 2D image. To improve the quality of inversion, we



Figure 6. Comparison with pose changes. We compared our method with the warp-based model and the first-order model. The top right images are the target poses, and the top left images are the input images.

	LPIPS↓	ID↑
baseline	0.254	0.353
+ \mathcal{W}_{tri}	0.227	0.352
+ \mathcal{L}_{cyc}	0.223	0.362

Table 2. Quantitative comparison results between baseline, baseline+ \mathcal{W}_{tri} , and baseline + $\mathcal{W}_{tri} + \mathcal{L}_{cyc}$

356 created a new feature map corresponding to each coordinate
 357 and newly designed \mathcal{L}_{cyc} , which is added to the total loss.
 358 By creating a new feature map, the performance of the encoder network was highly improved.
 359 The new customized loss, \mathcal{L}_{cyc} , corrects the error when the 2D image
 360 is projected on the new feature map. This \mathcal{L}_{cyc} was
 361 slightly different from the existing cycle consistency loss,
 362 which was determined by making latent codes from the im-
 363 age and setting the difference as the loss value. Likewise,
 364 when looking at the learning results, it was possible to con-
 365 firm the quality improvement of the image by correcting
 366 the loss. Our method performs a way of inverting real im-
 367 ages to 3D GANs, but apart from providing excellent visual
 368 quality, there are limitations in expressing the 3D volume of
 369 the dataset. In other words, the tendency to create a back-
 370

bone feature map that depends on the viewpoint the input image views is vital, so reproducibility is sufficient. Still, the rendering volume expression tends to be insufficient. Therefore, the challenge for volume expression is a future task. Since our model is generated from the StyleGAN2 backbone, we can proceed with various application experiments using the StyleGAN2 latent space for future work. This contains the reconstruction of a 2D sketch, 2D semantic mask map using CelebAMask-HQ [21], and various image control methods [11, 26, 35] such as hairstyle and facial expression into 3D. Also, we can use our network to make 3D-aware synthetic images using non-human datasets such as AFHQ Dataset [6]. In summary, to create a 3D portrait image from the 2D image, new feature maps and loss to correct the feature map were newly presented. Compared with the existing methods, a greater effect was obtained.

References

- [1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 1, 2
- [2] Rameen Abdal, Peihao Zhu, Niloy J Mitra, and Peter Wonka. Styleflow: Attribute-conditioned exploration of stylegan- 388
389
390
391

- 394 generated images using conditional continuous normalizing
395 flows. *ACM Transactions on Graphics (ToG)*, 40(3):1–21,
396 2021. 3
- [3] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle:
397 A residual-based stylegan encoder via iterative refinement.
398 In *Proceedings of the IEEE/CVF International Conference
399 on Computer Vision*, pages 6711–6720, 2021. 1, 2, 4, 6
- [4] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and
402 Amit Bermano. Hyperstyle: Stylegan inversion with hy-
403 pernetworks for real image editing. In *Proceedings of
404 the IEEE/CVF Conference on Computer Vision and Pattern
405 Recognition*, pages 18511–18521, 2022. 2
- [5] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki
407 Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo,
408 Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero
409 Karras, and Gordon Wetzstein. Efficient geometry-aware 3D
410 generative adversarial networks. In *CVPR*, 2022. 2, 3
- [6] Yunjey Choi, Youngjung Uh, Jaejun Yoo, and Jung-Woo Ha.
411 Stargan v2: Diverse image synthesis for multiple domains.
412 In *Proceedings of the IEEE/CVF conference on computer vi-
413 sion and pattern recognition*, pages 8188–8197, 2020. 7
- [7] Yu Deng, Jiaolong Yang, Sicheng Xu, Dong Chen, Yunde
415 Jia, and Xin Tong. Accurate 3d face reconstruction with
416 weakly-supervised learning: From single image to image set.
417 In *Proceedings of the IEEE/CVF Conference on Computer
418 Vision and Pattern Recognition Workshops*, pages 0–0, 2019.
419 6
- [8] Emily Denton, Ben Hutchinson, Margaret Mitchell, and
422 Timnit Gebru. Detecting bias with generative counterfactual
423 face attribute augmentation. 2019. 3
- [9] Tan M. Dinh, Anh Tuan Tran, Rang Nguyen, and Binh-Son
425 Hua. Hyperinverter: Improving stylegan inversion via hy-
426 pernetwork. In *Proceedings of the IEEE/CVF Conference on
427 Computer Vision and Pattern Recognition (CVPR)*, 2022. 2
- [10] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape
429 induction from 2d views of multiple objects. In *2017 In-
430 ternational Conference on 3D Vision (3DV)*, pages 402–411.
431 IEEE, 2017. 2
- [11] Rinon Gal, Or Patashnik, Haggai Maron, Gal Chechik,
433 and Daniel Cohen-Or. Stylegan-nada: Clip-guided do-
434 main adaptation of image generators. *arXiv preprint
435 arXiv:2108.00946*, 2021. 3, 7
- [12] Lore Goetschalckx, Alex Andonian, Aude Oliva, and Phillip
437 Isola. Ganalyze: Toward visual definitions of cognitive im-
438 age properties. In *Proceedings of the IEEE/CVF Interna-
439 tional Conference on Computer Vision*, pages 5744–5753,
440 2019. 3
- [13] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian
442 Theobalt. Stylenet: A style-based 3d-aware genera-
443 tor for high-resolution image synthesis. *arXiv preprint
444 arXiv:2110.08985*, 2021. 2
- [14] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and
446 Sylvain Paris. Ganspace: Discovering interpretable gan con-
447 trols. *Advances in Neural Information Processing Systems*,
448 33:9841–9850, 2020. 3
- [15] Peter Hedman, Pratul P Srinivasan, Ben Mildenhall,
449 Jonathan T Barron, and Paul Debevec. Baking neural
450 radiance fields for real-time view synthesis. In *Proceedings
451 of the IEEE/CVF International Conference on Computer Vi-
452 sion*, pages 5875–5884, 2021. 2
- [16] Yuge Huang, Yuhang Wang, Ying Tai, Xiaoming Liu,
453 Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang.
Curricularface: adaptive curriculum learning loss for deep
454 face recognition. In *proceedings of the IEEE/CVF con-
455 ference on computer vision and pattern recognition*, pages
456 5901–5910, 2020. 6
- [17] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen.
457 Progressive growing of gans for improved quality, stability,
458 and variation. *arXiv preprint arXiv:1710.10196*, 2017. 6
- [18] Tero Karras, Samuli Laine, and Timo Aila. A style-based
459 generator architecture for generative adversarial networks.
In *Proceedings of the IEEE/CVF conference on computer vi-
460 sion and pattern recognition*, pages 4401–4410, 2019. 1, 2
- [19] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten,
461 Jaakko Lehtinen, and Timo Aila. Analyzing and impro-
462 ving the image quality of stylegan. In *Proceedings of
463 the IEEE/CVF conference on computer vision and pattern
464 recognition*, pages 8110–8119, 2020. 1, 2, 3
- [20] Tero Karras, Miika Aittala, Samuli Laine, Erik Häkkinen,
465 Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free
466 generative adversarial networks. In *Proc. NeurIPS*, 2021. 1,
467 2
- [21] Cheng-Han Lee, Ziwei Liu, Lingyun Wu, and Ping Luo.
468 Maskgan: Towards diverse and interactive facial image ma-
469 nipulation. In *Proceedings of the IEEE/CVF Conference
470 on Computer Vision and Pattern Recognition*, pages 5549–
471 5558, 2020. 7
- [22] Yiyi Liao, Katja Schwarz, Lars Mescheder, and Andreas
472 Geiger. Towards unsupervised learning of generative mod-
473 els for 3d controllable image synthesis. In *Proceedings of
474 the IEEE/CVF conference on computer vision and pattern
475 recognition*, pages 5871–5880, 2020. 2
- [23] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik,
476 Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf:
477 Representing scenes as neural radiance fields for view syn-
478 thesis. In *European conference on computer vision*, pages
479 405–421. Springer, 2020. 2
- [24] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian
481 Richardt, and Yong-Liang Yang. Hologan: Unsupervised
482 learning of 3d representations from natural images. In
483 *Proceedings of the IEEE/CVF International Conference on
484 Computer Vision*, pages 7588–7597, 2019. 2
- [25] Thu H Nguyen-Phuoc, Christian Richardt, Long Mai,
486 Yongliang Yang, and Niloy Mitra. Blockgan: Learning
487 3d object-aware scene representations from unlabelled im-
488 ages. *Advances in Neural Information Processing Systems*,
489 33:6767–6778, 2020. 2
- [26] Or Patashnik, Zongze Wu, Eli Shechtman, Daniel Cohen-Or,
492 and Dani Lischinski. Styleclip: Text-driven manipulation of
493 stylegan imagery. In *Proceedings of the IEEE/CVF Interna-
494 tional Conference on Computer Vision*, pages 2085–2094,
495 2021. 3, 7
- [27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya
496 Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry,
497 Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learn-
498

- 509 ing transferable visual models from natural language super-
510 vision. In *International Conference on Machine Learning*,
511 pages 8748–8763. PMLR, 2021. 3
- 512 [28] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan,
513 Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding
514 in style: a stylegan encoder for image-to-image translation.
515 In *IEEE/CVF Conference on Computer Vision and Pattern*
516 *Recognition (CVPR)*, 2021. 1, 2, 4, 5
- 517 [29] Daniel Roich, Ron Mokady, Amit H Bermano, and Daniel
518 Cohen-Or. Pivotal tuning for latent-based editing of real im-
519 ages. *ACM Trans. Graph.*, 2021. 1, 2, 6
- 520 [30] Yujun Shen and Bolei Zhou. Closed-form factorization of
521 latent semantics in gans. In *Proceedings of the IEEE/CVF*
522 *Conference on Computer Vision and Pattern Recognition*,
523 pages 1532–1540, 2021. 3
- 524 [31] Yujun Shen, Jinjin Gu, Xiaoou Tang, and Bolei Zhou. In-
525 terpreting the latent space of gans for semantic face editing.
526 In *Proceedings of the IEEE/CVF conference on computer vi-*
527 *sion and pattern recognition*, pages 9243–9252, 2020. 3
- 528 [32] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov,
529 Elisa Ricci, and Nicu Sebe. First order motion model for
530 image animation. In *Conference on Neural Information Pro-*
531 *cessing Systems (NeurIPS)*, 2019. 6
- 532 [33] Attila Szabó, Givi Meishvili, and Paolo Favaro. Unsu-
533 pervised generative 3d shape learning from natural images.
534 *arXiv preprint arXiv:1910.00287*, 2019. 2
- 535 [34] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and
536 Daniel Cohen-Or. Designing an encoder for stylegan image
537 manipulation. *arXiv preprint arXiv:2102.02766*, 2021. 1, 2,
538 5
- 539 [35] Andrey Voynov and Artem Babenko. Unsupervised discov-
540 ery of interpretable directions in the gan latent space. In
541 *International conference on machine learning*, pages 9786–
542 9796. PMLR, 2020. 3, 7
- 543 [36] Jiajun Wu, Chengkai Zhang, Tianfan Xue, Bill Freeman, and
544 Josh Tenenbaum. Learning a probabilistic latent space of
545 object shapes via 3d generative-adversarial modeling. *Ad-*
546 *vances in neural information processing systems*, 29, 2016.
547 2
- 548 [37] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shecht-
549 man, and Oliver Wang. The unreasonable effectiveness of
550 deep features as a perceptual metric. In *Proceedings of the*
551 *IEEE conference on computer vision and pattern recogni-*
552 *tion*, pages 586–595, 2018. 6
- 553 [38] Peng Zhou, Lingxi Xie, Bingbing Ni, and Qi Tian. Cips-3d:
554 A 3d-aware generator of gans based on
555 conditionally-independent pixel synthesis. *arXiv preprint*
556 *arXiv:2110.09788*, 2021. 2