International Journal of Computer Vision
https://doi.org/10.1007/s11263-024-02085-9

DisCO: Portrait Distortion Correction with Perspective-Aware 3D GANs

Zhixiang Wang[1,2] · Yu-Lun Liu[3] · Jia-Bin Huang[4] · Shin'ichi Satoh[1,2] · Sizhuo Ma[5] · Gurunandan Krishnan[5] · Jian Wang[5]

Received: 2 August 2023 / Accepted: 2 January 2024

Abstract

Close-up facial images captured at short distances often suffer from perspective distortion, resulting in exaggerated facial features and unnatural/unattractive appearances. We propose a simple yet effective method for correcting perspective distortions in a single close-up face image. We first perform 3D GAN inversion using a perspective-distorted input facial image by jointly optimizing the intrinsic and extrinsic camera parameters and the face latent code. To address the ambiguity inherent in this joint optimization, we develop starting from a short distance, optimization scheduling, reparametrizations, and geometric regularization. Re-rendering the portrait at a proper focal length and camera distance effectively corrects perspective distortions and produces more natural-looking results. We also incorporate a workflow to handle full images rather than limiting our method to cropped faces. Our experiments show that our method compares favorably against previous approaches qualitatively and quantitatively. We showcase numerous examples validating the applicability of our method on *in-the-wild* portrait photos. Our code is available at https://github.com/lightChaserX/DisCO.

Keywords Portrait · Perspective distortion · 3D GANs

1 Introduction

Every day, millions of people enjoy taking selfies with their smartphones. Although these devices have high-quality cameras that can capture high-resolution images with accurate colors, the captured selfies tend to suffer from perspective distortion, which is particularly noticeable when the camera-to-subject distance is extremely short (usually 20–60cm), as shown in the first row of Fig. 1. Such distortion prominently exaggerates frontal facial features, like the nose, making the face appear unnatural and asymmetrical. Additionally, the distortion often obscures the side of the face, including the ears. Consequently, this distortion creates unflattering images and could negatively impact face identification and other related tasks.

Existing efforts that automatically correct portrait perspective distortions often involve reconstruction-based warping (Fried et al., 2016) or learning-based warping (Nagano et al., 2019; Zhao et al., 2019). However, these methods rely on estimating a 2D flow map to warp the image, leading to incorrect face shapes after correction, as shown in Fig. 2a, b. Moreover, they cannot generate disoccluded pixels, such as ears and hair, which may be revealed in the background. Additionally, the warping-based methods cannot correct the non-face regions and cause misalignment between the face and body, as shown in Fig. 2b.

In this paper, we propose to correct portrait perspective distortion through *3D GAN inversion*, as 3D GANs (Chan et al., 2021, 2022; Deng et al., 2022; Or-El et al., 2022; Niemeyer & Geiger, 2021; Sun et al., 2022; Zhou et al., 2021) are effectiveness in generating 3D-consistent and realistic facial features. Our approach inverts a distorted input face image into the corresponding facial latent code, camera pose, and focal length. However, optimizing these parameters from a single distorted face is challenging, and existing

Communicated by Chongyi Li.

Part of this work was done while Z.W was an intern at Snap Research, NYC.

✉ Jian Wang
jwang4@snapchat.com
https://portrait-disco.github.io/

[1] The University of Tokyo, Tokyo, Japan
[2] National Institute of Informatics, Tokyo, Japan
[3] National Yang Ming Chiao Tung University, Hsinchu, Taiwan
[4] University of Maryland, College Park, USA
[5] Snap Inc., Santa Monica, USA

Published online: 17 June 2024

GAN inversion methods like PTI (Roich & Mokady, 2021) fail to provide accurate results when applied to 3D GANs, as shown in Fig. 2c, d. To address this issue, we propose four designs: (1) closeup camera-to-face distance initialization, (2) separate optimization of face and camera parameters, (3) reparameterizations, and (4) landmark constraints. We also incorporate a workflow to handle full images rather than cropped faces. Our method can correct perspective distortion by adjusting the camera-to-face distance (as shown in the second row of Fig. 1) and applying special visual effects such as dolly-zoom by adjusting camera parameters.

We make the following contributions:

- We propose a pipeline for correcting portrait distortion using perspective-aware 3D GAN inversion. Our pipeline integrates GAN inversion for the face region and a workflow to achieve camera-consistent full-image manipulation, avoiding inharmonious composition between the face and body. This enables various visual effects, including dolly-zoom videos.
- We explore several design choices to avoid the optimization falling into sub-optimal solutions, including better initialization, separate optimization of face and camera parameters, reparameterizations, and landmark loss.
- We establish a comprehensive evaluation for portrait perspective distortion correction, including quantitative, qualitative, full-image, and video evaluation, which will benefit future research in this area.

2 Related Work

2.1 Portrait Perspective Undistortion

Selfie photos taken from close distances often suffer from perspective distortions, resulting in unappealing distortions such as an enlarged nose, uneven facial features, asymmetry, and hidden ears and hair. These distortions are commonly referred to as “selfie effects" and are a significant concern for many people, with some even considering plastic surgery as a solution (Ward et al., 2018). Research indicates that the camera distance plays a vital role in portrait perception, and studies have identified an “optimal distance" for capturing undistorted facial images (Bryan et al., 2012; Cooper et al., 2012). Specifically, it has been found that 50mm lenses are ideal for producing natural-looking and flattering images. In response, smartphone manufacturers have attempted to encourage users to take selfies from a greater distance by reducing the field of view (Williams & Motta, 2017).

Current perspective distortion methods either model distortion as a warping function parameter (Valente & Soatto, 2015) or manipulate camera-to-face distance in a reconstructed model (Fried et al., 2016). While deep learning-based methods (Zhao et al., 2019) can correct minor distortions, they struggle with severe distortions due to inaccurate 3D face-fitting steps and the inability to inpaint occluded regions like ears using 2D warping flow maps. 3D radiance field-based methods (Athar et al., 2022; Gafni et al., 2021; Gao et al., 2020) provide full control of camera parameters but require many training images and do not leverage face priors. Our method uses 3D GAN inversion to correct close-range input images, fill in unobserved regions, and

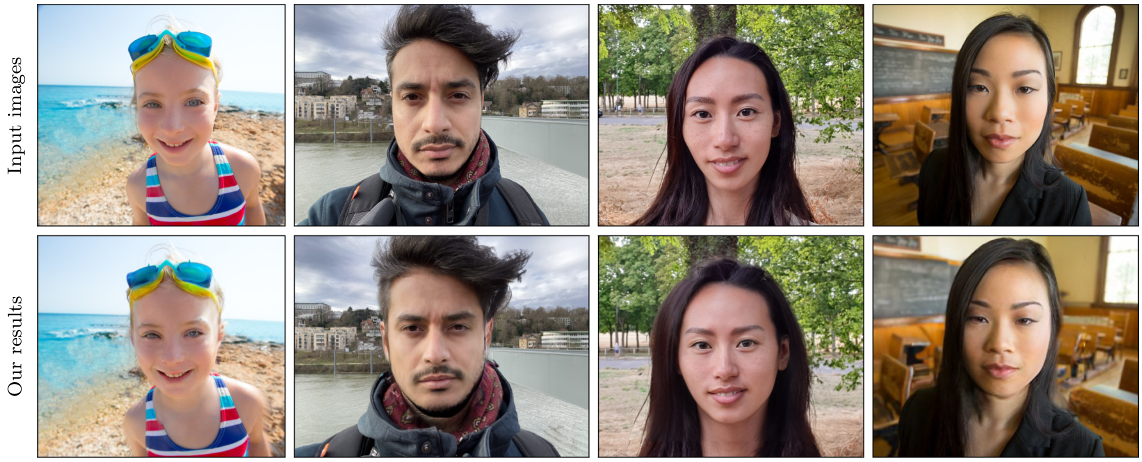


Fig. 1 Portrait distortion correction. Portrait photos captured from a short distance (e.g., selfie) often suffer from undesired perspective distortions (the first row). Our approach corrects these perspective distortions and synthesizes visually pleasant views by *virtually* enlarging the focal length and moving the camera further away from the subject. Please visit our website to view video results

allow flexible camera-to-face distances, effectively correcting severe distortions.

2.2 3D GANs

The neural 3D representation (Park, 2019; Michalkiewicz, 2019; Atzmon & Lipman, 2020; Gropp et al., 2020; Sitzmann et al., 2019; Peng, 2020; Mescheder et al., 2019; Chen & Zhang, 2019; Mildenhall, 2020; Wang et al., 2021; Mildenhall, 2020) has shown impressive photorealism in novel view synthesis and is a foundational representation for 3D-aware generation. Implicit 3D representations have been leveraged by recently proposed 3D GANs (Chan et al., 2021, 2022; Deng et al., 2022; Niemeyer & Geiger, 2021; Or-El et al., 2022; Zhou et al., 2021) to generate high-resolution outputs with remarkable details and 3D consistency. Our work uses a pre-trained model of EG3D (Chan et al., 2022) due to its computational efficiency and its ability to produce photorealistic 3D-consistent images, similar to those generated by StyleGANs (Karras et al., 2019, 2020). However, our method is agnostic to the choice of 3D GANs.

2.3 GAN Inversion

GAN inversion is a technique that maps a real image back into the latent space of a pre-trained GAN, which can expand the model's editing capability to real photos. There are two main categories of GAN inversion methods according to the type of GANs: 2D and 3D. 2D GAN inversion methods optimize the latent code for a single image (Abdal et al., 2019; Creswell & Bharath, 2018) or use a learned encoder to project images to the latent space (Alaluf & Patashnik, 2021; Richardson et al., 2021; Tov et al., 2021). Some hybrid strategies combine both methods to refine the encoder's output latent code by optimization (Guan et al., 2020; Zhu et al., 2016). Recent 2D GAN inversion methods achieve high editing capabilities and have been extended for video editing (Abdal et al., 2022; Xu et al., 2022; Tzaban et al., 2022). However, editing 3D-related attributes such as camera parameters and head pose remains inconsistent and prone to severe flickering, as the pre-trained generator is unaware of the 3D structure.

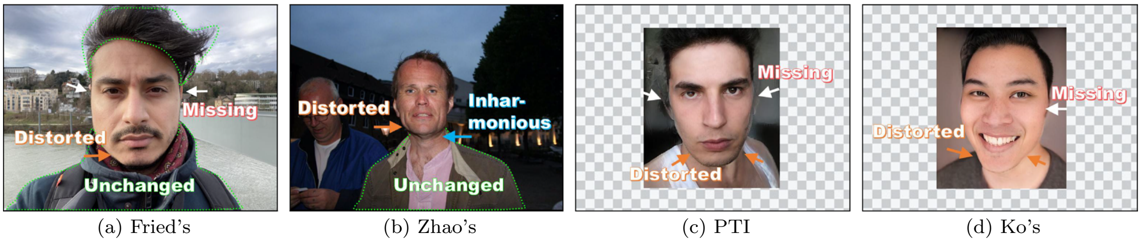
On the contrary, 3D GAN inversion methods (Ko et al., 2023; Lin et al., 2022; Sun et al., 2022; Xie et al., 2023; Xu et al., 2023; Wang et al., 2022) achieve 3D-consistent reconstruction and manipulation by incorporating 2D GAN inversion methods, such as PTI (Roich & Mokady, 2021), with estimated camera parameters obtained from 3DMM or other algorithms. While some recent methods like Lin et al. (2022) and Wang et al. (2022) estimate all camera parameters from 3DMM and keep them fixed, Ko et al. (2023) assume known camera intrinsics and camera-to-face distances to jointly optimize the face latent code and the rest of camera parameters. However, correcting perspective distortion requires estimating the face latent code, camera-to-face distance, and focal length, posing a challenge due to ambiguity among these parameters. To address this, we propose a perspective-aware 3D GAN inversion method to estimate the face latent code and camera parameters accurately.

3 Preliminaries

We will briefly introduce the basics of StyleGAN and StyleGAN inversion, followed by those of 3D GANs.

StyleGAN Given a random sample $\mathbf{z} \in \mathbb{R}^{512}$ drawn from a latent normal distribution, StyleGAN (Karras et al., 2019) creates a new sample from the data distribution. It first maps $\mathbf{z}$ to an intermediate latent vector $\mathbf{w} \in \mathbb{R}^{512}$ using a learned mapping $\mathbf{w} = H_{\boldsymbol{\theta}}(\mathbf{z})$. The space of the latent vector $\mathbf{w}$ (style code) is commonly referred to as $\mathcal{W}$ and that of the random sample $\mathbf{z}$ is termed as $\mathcal{Z}$. Taking the vector $\mathbf{w}$ as input, the generator $G_{\boldsymbol{\theta}}$ renders

$$I = G_{\boldsymbol{\theta}}(\mathbf{w}) = G_{\boldsymbol{\theta}}(H_{\boldsymbol{\theta}}(\mathbf{z}))\,. \qquad (1)$$



(a) Fried's (b) Zhao's (c) PTI (d) Ko's

Fig. 2 Limitations of state-of-the-art techniques. **a**, **b** Fried's (Fried et al., 2016) and Zhao's (Zhao et al., 2019) are 2D warping-based methods that cannot fully recover the correct face geometry or generate missing content, such as ears. Moreover, **b** shows that the corrected image using Zhao et al. (2019) exhibits an inharmonious composition of the face and neck, in contrast to our result in Fig. 12. **c**, **d** are GAN inversion methods that can manipulate camera parameters. **c** PTI (Roich & Mokady, 2021) is a 2D GAN inversion method that may produce sub-optimal solutions and incorrect facial geometry when applied to 3D GANs. **d** is a 3D GAN inversion method that jointly optimizes face and partial camera parameters but cannot generate correct geometry. Both **c** and **d** can only correct facial regions instead of the full body

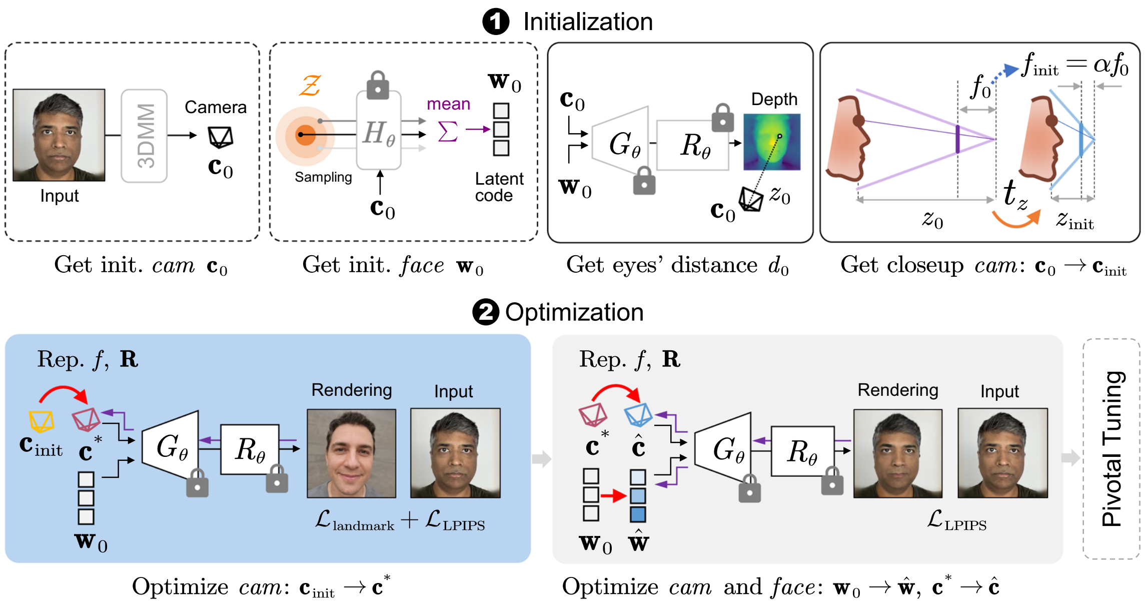


Fig. 3 Perspective-aware 3D GAN inversion. **Step 1**: *Initialization*. We first fit a 3DMM model to the image to get an initial camera pose and average randomly sampled latent codes to initialize the face latent code. The initialized camera pose can roughly match the face direction and size, but the estimated focal length and camera-to-subject distance are inaccurate. Then, we get a closeup camera by pushing the camera-to-face distance z_0 to a small value z_{int} and changing the focal length according to the reparameterization method. **Step 2**: *Optimization*. We fix the face latent code, generator, and neural renderer to optimize the camera parameters. Here, we reparameterize the focal length and rotation to further ease optimization. After optimizing the camera poses, we simultaneously optimize the face latent code and camera parameters. Finally, we perform pivotal tuning to fine-tune the generator to achieve high-fidelity results on real images

StyleGAN inversion enables the back-projection of an input real image, denoted as $\mathbf{x}$, to the latent space of the pre-trained generator. This projection allows us to perform various editing operations on the input image. Images are usually inverted to the $\mathcal{W}$ space instead of the $\mathcal{Z}$ space due to its exceptional fine-grained editing ability. To obtain the optimal latent vector $\hat{\mathbf{w}} \in \mathcal{W}$, we minimize the perceptual loss function (Zhang & Isola, 2018) to find

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \mathcal{L}_{\text{LPIPS}}(G_{\boldsymbol{\theta}}(\mathbf{w}), \mathbf{x})\,. \tag{2}$$

Due to potential disparities between the input real image and the data distribution of the pre-trained generator, the reconstructed image using the inverted latent code $\hat{\mathbf{w}}$ might suffer from change of appearance. To address this, Roich *et al.* Roich and Mokady (2021) propose *pivotal tuning* that unfreezes and fine-tunes the generator using fixed $\hat{\mathbf{w}}$. The objective is to optimize the generator's parameters

$$\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{LPIPS}}(G_{\boldsymbol{\theta}}(\hat{\mathbf{w}}), \mathbf{x}) + \lambda_{L2}\mathcal{L}_{L2}(G_{\boldsymbol{\theta}}(\hat{\mathbf{w}}), \mathbf{x})\,. \tag{3}$$

3D GAN combines StyleGAN and implicit 3D representations for 3D controllable image generation. In our used EG3D (Chan et al., 2022), the StyleGAN, including $H_{\boldsymbol{\theta}}$ and $G_{\boldsymbol{\theta}}$, uses the latent code $\mathbf{z}$ and camera parameters $\mathbf{c}$ as input to generate an implicit 3D representation. Then, the neural renderer $R_{\boldsymbol{\theta}}$ takes the implicit representation and camera parameters to produce the final image. The formulation of the whole generation process is given by:

$$I = R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}(\underbrace{H_{\boldsymbol{\theta}}(\mathbf{z}, \mathbf{c})}_{\mathbf{w}}, \mathbf{c})\,, \quad \mathbf{w} \in \mathcal{W}\,, \tag{4}$$

where $\mathbf{c}$ includes the intrinsic and extrinsic parameters. The training images are sourced from the FFHQ dataset (Karras et al., 2019), with an underlying assumption that the cameras used for capturing these images were placed on a spherical surface, each at a *large* radius of 2.7. All cameras maintained fixed intrinsic parameters.

4 Perspective-Aware 3D GAN Inversion

4.1 Overview

Perspective distortion is caused by the short camera-to-subject distance. We propose the perspective-aware 3D GAN inversion that utilizes pre-trained 3D GANs to invert the perspective-distorted portrait into its corresponding face latent code and camera parameters (see Fig. 4). Then, we adjust the camera parameters—the z translation and focal

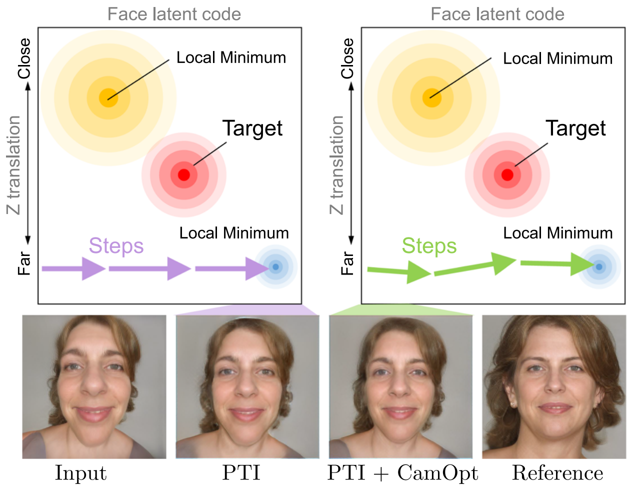


Fig. 4 Difficulty in inverting distorted portraits. Applying the 2D GAN inversion method PTI (Roich & Mokady, 2021) with an incorrect yet fixed camera seems to fit the distorted input image well. However, projecting the face to a distant view reveals the reconstructed geometric shape is wrong. This is because the optimization falls into a local minimum for the incorrect camera. Naïvely adding camera optimization to PTI does not provide significant improvement, and its performance is close to PTI with a fixed camera as the optimization of z translation is subtle

length—to re-render a novel portrait with alleviated distortion.

4.2 Ambiguity in Joint Optimization

Existing 3D GAN inversion methods (Sun et al., 2022; Lin et al., 2022; Ko et al., 2023; Xie et al., 2023) target undistorted face images captured from distant viewpoints. They usually adapt 2D GAN inversion techniques (Roich & Mokady, 2021) to 3D GANs with camera parameters estimated from a 3D morphable model (Deng et al., 2019). While these parameters, particularly focal length and camera-to-subject distance (Fried et al., 2016; Burgos-Artizzu et al., 2014), may be significantly inaccurate, these methods can still yield reasonable results. The effectiveness is attributed to the use of *undistorted* input images, allowing an approximation with a weak perspective model. Consequently, inaccuracies in focal length and camera-to-subject distance typically result in just minor scale discrepancies in the reconstructed 3D face geometries.

However, close-up photography is an **entirely different** story due to the perspective projection, and the distortion that makes the face appearance differ from the real face. Inverting using these inaccurate parameters directly could lead to faces with *distorted* geometry (see Fig. 3). For recovering the correct 3D face geometry of a distorted image, accurate estimation of *both* camera-to-subject distance and focal length is essential. Therefore, we propose to jointly optimize[1] the camera parameters and the face latent code[2]:

$$\hat{\mathbf{w}}, \hat{\mathbf{c}} = \underset{\mathbf{w},\, \mathbf{c}}{\arg\min}\, \mathcal{L}(R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}(\mathbf{w}), \mathbf{c}), \mathbf{x})\,. \tag{5}$$

Inferring unknown face and camera parameters from a single input image is an *ill-posed* problem, as there exist infinite combinations of focal length, camera-to-subject distance, and face shape producing images similar to the input image. Due to this ambiguity, combining naïve camera optimization with the 2D GAN inversion method PTI (Roich & Mokady, 2021) encounters significant challenges, as shown in Fig. 3.

4.3 Designs for Reducing Ambiguity

To alleviate the ambiguity in joint optimization, we propose four designs: starting from a short distance, optimization scheduling, reparameterizations, and landmark regularization.

4.3.1 Reparameterizations

An intuitive solution to relieve ambiguity is to reduce the number of parameters.

Focal length reparameterization The focal length is coupled with camera extrinsic parameters and the face geometry, inhibiting the inversion (Ponimatkin et al., 2022). Even having the ground-truth face latent code $\mathbf{w}$, and other camera parameters, it is still ambiguous to invert the focal length and the z translation, as shown in Fig. 5. We observe the optimization of focal length negatively affecting that of the z translation. During joint optimization, their updates are small, but in different directions.

However, focal length only controls the scale of the face, not the distortion level. Therefore, we propose to reparameterize the focal length based on an underlying relationship—no matter the manipulation, the scale of the input face should be constant. We derive a solution when adjusting the camera, relating the focal length f to the z translation t_z according to

$$f = \alpha f_0\,, \quad \text{where} \tag{6}$$

$$\alpha = 1 - (t_{z0} - t_z)/z_0\,. \tag{7}$$

[1] Though Ko's method (Ko et al., 2023) also jointly optimize the camera parameters and the face latent code, their camera parameters exclude focal length and camera-to-subject distance and therefore they cannot invert a distorted portrait.

[2] The 3D GAN we use is EG3D (Chan et al., 2022) trained on FFHQ (Karras et al., 2019) and our optimization is based on the $\mathcal{W}+$ space of EG3D.

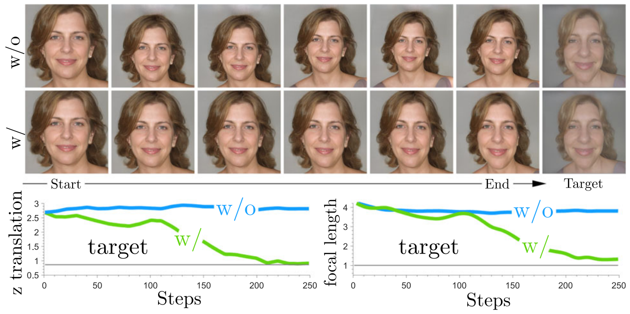
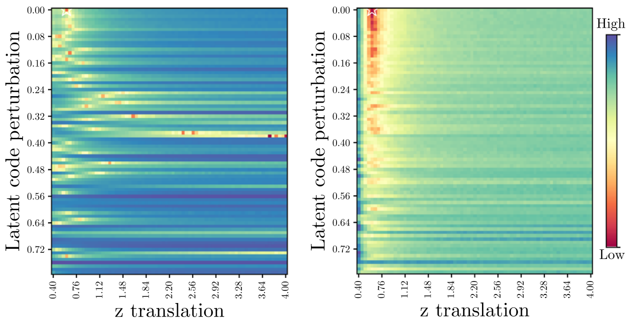


Fig. 5 Motivation of the focal length reparameterization. We input an image rendered by our 3D GAN to find the rendering camera parameters (focal length and z translation) with a face latent code near the ground truth. Without focal length reparameterization (w/o), optimization is difficult. However, with our focal length reparameterization (w/), optimized parameters approach the target



(a) $L2$ loss (b) Landmark loss

Fig. 6 Partial landscape of loss functions. For visualization, we randomly sample a latent code **w** and a close-up camera **c** to render an image, marked with a white star at the top left of each box. After enlarging the z translation and the focal length, and perturbing the latent code by $\mathbf{w} = \mathbf{w} + e\mathbf{n}$, where **n** is a Gaussian noise, the 3D GAN renders new images. We calculate the losses between them and the original image. The $L2$ loss is more complex than the landmark loss as it has many local minima distributed over the landscape. In the visualized region, the landmark loss approximately exhibits an unimodal and convex loss surface.

We set z_0 as the coordinate of the left or right eye in the camera coordinate system. During the optimization, we update the focal length by $f = \gamma \alpha f_0$, where γ is a learnable parameter with a small learning rate to accommodate approximation error. We will elaborate on the derivation below.

Suppose the world-to-camera transformation is:

$$\begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix} = [x, y, z, 1]^{\top} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{q} \\ 1 \end{bmatrix}, \tag{8}$$

where $\mathbf{p}, \mathbf{q} \in \mathbb{R}^3$, $\mathbf{R} = [\mathbf{r}_x, \mathbf{r}_y, \mathbf{r}_z]^{\top} \in \mathbb{R}^{3\times3}$ is the rotation matrix and $\mathbf{t} = [t_x, t_y, t_z]^{\top} \in \mathbb{R}^{3\times1}$ is the translation vector. The intrinsic matrix **K** transforms a point from camera space to the image plane as:

$$z\mathbf{u} = z\,[u, v, 1]^{\top} = \mathbf{K}\,\mathbf{p} = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} [x, y, z]^{\top}. \tag{9}$$

Let $\mathbf{p}_0 = [x_0, y_0, z_0]^{\top} \in \mathbb{R}^3$ denotes the initial coordinate of a 3D point in the camera system. Changing the camera from pose $\mathbf{R}_0, \mathbf{t}_0$ to $\mathbf{R}_1, \mathbf{t}_1$ and the camera intrinsic matrix from $\mathbf{K}_0$ to $\mathbf{K}_1$ yield a corresponding new coordinate of the point

$$\mathbf{p}_1 = \mathbf{R}_1\mathbf{R}_0^{-1}(\mathbf{p}_0 - \mathbf{t}_0) + \mathbf{t}_1\,, \text{ and} \tag{10}$$

$$\mathbf{u}_1 = \mathbf{K}_1\mathbf{R}_1\mathbf{R}_0^{-1}\left(z_0\mathbf{K}_0^{-1}\mathbf{u}_0 - \mathbf{t}_0\right)/z_1 + \mathbf{K}_1\mathbf{t}_1/z_1\,. \tag{11}$$

To guarantee the project of the point in the image plane maintains fixed, we have $\mathbf{u}_0 = \mathbf{u}_1$. It is possible to derive a relationship between the focal length and the camera extrinsic parameters.

We approximate the relationship to ease computation. Since the estimation of camera rotation is not problematic, the initialization is close to the true value. Therefore, we let the camera rotation change slightly, that is $\mathbf{R}_1 = \mathbf{R}_0 + \boldsymbol{\delta}$ with $\boldsymbol{\delta}$ is a matrix close to $\mathbf{0}$. Hence, we get the approximation:

$$\mathbf{p}_1 \approx \mathbf{p}_0 - \mathbf{t}_0 + \mathbf{t}_1\,, \text{ and} \tag{12}$$

$$\mathbf{u}_1 \approx \mathbf{K}_1\left(z_0\mathbf{K}_0^{-1}\mathbf{u}_0 - \mathbf{t}_0\right)/z_1 + \mathbf{K}_1\mathbf{t}_1/z_1\,. \tag{13}$$

We further assume t_x, t_y, c_x and c_y do not change. With these assumptions, it is easy to derive the relationship from Eq. (13): $f_1/f_0 = z_1/z_0 := \alpha$. Substituting Eq. (12) into the relationship, we obtain Eq. (7). We keep the position of the eyes constant to guarantee the scale of the face does not change as the eyes usually are not occluded and hold stable landmark detection.

Camera rotation reparameterization Besides the focal length, camera rotation contributes to a certain level of degree of freedom. We formulate the rotation matrix **R** with a Gram-Schmidt based 6D representation, as done in SC-NeRF (Jeong & Ahn, 2021), to ensure orthogonality and reduce the number of parameters:

$$\mathbf{R} = \begin{bmatrix} | & | & | \\ \mathbf{r}_x & \mathbf{r}_y & \mathbf{r}_z \\ | & | & | \end{bmatrix} = F\left(\mathbf{A}\right) = F\left(\begin{bmatrix} | & | \\ \mathbf{a}_1 & \mathbf{a}_2 \\ | & | \end{bmatrix}\right), \tag{14}$$

where $\mathbf{r}_x, \mathbf{r}_y, \mathbf{r}_z \in \mathbb{R}^3$ are $\mathbf{r}_x = N(\mathbf{a}_1)$, $\mathbf{r}_y = N(\mathbf{a}_2 - (\mathbf{r}_x \cdot \mathbf{a}_2)\mathbf{r}_x)$, and $\mathbf{r}_z = \mathbf{r}_x \times \mathbf{r}_y$, × denotes cross product, and $N(\cdot)$ denotes $L2$ normalization.

4.3.2 Landmark Regularization

Except for the parameters, the loss function also results in ambiguity. The LPIPS and $L2$ loss functions used in GAN inversion are based on image similarities and care more about

Algorithm 1: Optimization of Camera

```
Fix face latent code $\mathbf{w}$ and weights of $G_{\boldsymbol{\theta}}$.
while iterations $k < T_{\text{cam}}$ do
    Get the gradients $\nabla_{\mathbf{t}}\mathcal{L}$, $\nabla_{\mathbf{R}}\mathcal{L}$, $\nabla_{\gamma}\mathcal{L}$.
    Update $\delta t_z \leftarrow \delta t_z + \lambda_{\text{cam}}\nabla_{t_z}\mathcal{L}$.
    Update $t_z \leftarrow \delta t_z t_{z0}$.
    Get $\alpha$ according to Eq. (7).
    Update $f \leftarrow \gamma\alpha f_{\text{init}}$.
    Update $\mathbf{v} \leftarrow \mathbf{v} + \lambda_{\text{tiny}}\lambda_{\text{cam}}\nabla_{\mathbf{v}}\mathcal{L}$, for all $\mathbf{v} \in \{\mathbf{R}, t_x, t_y, \gamma\}$.
end
```

appearances than geometric shapes. It probably views two images with different geometric shapes similar. Thus, their landscape comprises homogeneous and isotropic local minima restricting the optimization, as Fig. 6a shows. On the contrary, we employ a landmark loss to increase the sensibility of optimization to geometric shapes. We use the dense landmarks estimated from MediaPipe (Lugaresi et al., 2019) to calculate their $L2$ distances. As illustrated in Fig. 6b, the landmark loss approximately has a unimodal and convex loss surface that is easier to optimize. Since there exist many *unreliable* landmarks in the distorted input image, such as the occluded regions, we define the landmark loss in an uncertainty-based format given by:

$$\mathcal{L}_{\text{LMK}}(m, m') = \sum_{i=1}^{N} \left(\log\left(\sigma_i^2\right) + \frac{\|m_i - m_i'\|_2^2}{2\sigma_i^2} \right), \tag{15}$$

where m_i denotes the coordinate of a landmark in the input face and m_i' is that in the rendered face, N that equals 468 is the number of landmarks, and σ_i is a learnable parameter to control the uncertainty of each landmark so that the loss can ignore unreliable landmarks and focus on reliable ones.

4.3.3 Optimization Scheduling

Though landmark loss facilitates optimization, it cannot supervise the reconstruction of the appearance. It is important to use the image similarities to optimize the face latent code to reproduce the input image. However, joint optimization with a combined loss of the landmark loss and image similarities, $\mathcal{L} = \mathcal{L}_{\text{LPIPS}} + \mathcal{L}_{\text{LMK}}$, still frequently produces sub-optimal results. Considering we have reduced the number of camera parameters, the problem is mainly caused by the high-dimensional nature of the face latent code. The optimization of face latent code is more sensitive compared to camera parameters. Scaling learning rates (Meuleman et al., 2023) for different parameters could be a solution but probably needs to find a good trade-off otherwise leads to oscillations or subtle updates of some parameters. Therefore, we propose a *coarse-to-fine* optimization strategy.

Algorithm 2: Joint Optimization

```
Fix weights of $G_{\boldsymbol{\theta}}$.
while iterations $k < T_{\text{face}}$ do
    Get the gradients $\nabla_{\mathbf{t}}\mathcal{L}$, $\nabla_{\mathbf{R}}\mathcal{L}$, $\nabla_{\mathbf{w}}\mathcal{L}$, $\nabla_{\gamma}\mathcal{L}$.
    Update $\delta t_z \leftarrow \delta t_z + \lambda_{\text{cam}}\nabla_{t_z}\mathcal{L}$.
    Update $t_z \leftarrow \delta t_z t_{z0}$.
    Update $\mathbf{w} \leftarrow \mathbf{w} + \lambda_{\text{face}}\nabla_{\mathbf{w}}\mathcal{L}$.
    Get $\alpha$ according to Eq. (7).
    Update $f \leftarrow \gamma\alpha f_{\text{init}}$.
    Update $\mathbf{v} \leftarrow \mathbf{v} + \lambda_{\text{tiny}}\lambda_{\text{cam}}\nabla_{\mathbf{v}}\mathcal{L}$, for all $\mathbf{v} \in \{\mathbf{R}, t_x, t_y, \gamma\}$.
end
```

Specifically, we optimize the camera parameters first with a coarse face shape produced by the initialized face latent code $\mathbf{w}_0$ by:

$$\mathbf{c}^* = \arg\min_{\mathbf{c}} \mathcal{L}(R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}(\mathbf{w}_0), \mathbf{c}), \mathbf{x}) \,. \tag{16}$$

The optimization process is depicted in Algorithm 1. This phase yields a camera $\mathbf{c}^*$ roughly close to the target. Then, in the next phase, by setting $\mathbf{c}$ as $\mathbf{c}^*$, and $\mathbf{w}$ as $\mathbf{w}_0$, we find the face latent code and refine the coarse camera parameters $\mathbf{c}^*$:

$$\hat{\mathbf{w}}, \hat{\mathbf{c}} = \arg\min_{\mathbf{w}, \mathbf{c}} \mathcal{L}(R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}}(\mathbf{w}), \mathbf{c}), \mathbf{x}) \,. \tag{17}$$

The optimization process is given in Algorithm 2.

Given that our input images are in the wild, we incorporate pivotal tuning (Roich & Mokady, 2021) into our pipeline. Specifically, we fine-tune the generator while freezing the inverted parameters after the optimization of camera parameters and face latent code.

4.3.4 Starting from a Short Distance

The optimization usually relies heavily on the initial starting point. However, the task of obtaining an exact initial value for the camera-to-subject distance is challenging (Fried et al., 2016). Poor initialization makes the process easily trapped into a local minimum, producing a weird face shape. Considering distorted input images are captured at short camera-to-subject distances, we propose initiating the optimization process from a significantly reduced camera-to-subject distance. Specifically, we obtain a camera $\mathbf{c}_0$ by fitting a 3D morphable model (Deng et al., 2019). This camera generates faces matching the direction and scale of the input face. We push it to a close-up viewpoint by changing the z translation. It is critical to subsequently update the focal length following Eq. (7).

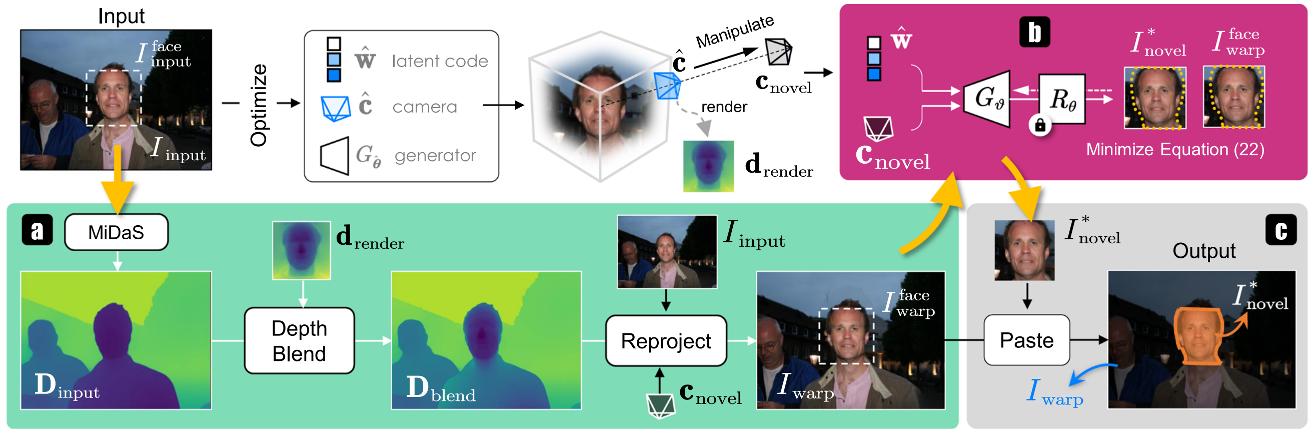


Fig. 7 Pipeline of processing full image. Taking a full close-up face image, we crop the closest face from the input image and perform 3D GAN inversion to infer its face latent code and camera parameters. After inversion, we manipulate the camera distance and focal length to render virtual images. **a**–**c** Geometry-aware stitching tuning. **a** We align and blend the rendered face depth map with the depth estimated from the entire image using a monocular depth estimation algorithm (MiDaS (Ranftl et al., 2020)). We project the entire input image to the same virtual camera positions of the result face image. **b** We fine-tune the generator by minimizing border loss and content loss to refine the border of the generated long-distance image. **c** Finally, we blend the warped full image with the generated face image.

5 Perspective-Aware Manipulation

After 3D GAN inversion, we acquire parameters to reconstruct the input face. Manipulating the camera parameter to $\mathbf{c}_{\text{novel}}$ yields a virtual image

$$I_{\text{novel}} = R_{\theta}(G_{\hat{\theta}}(\hat{\mathbf{w}}), \mathbf{c}_{\text{novel}}) \,. \tag{18}$$

To reduce the face perspective distortion, we increase the z translation. When the camera-to-subject distance is sufficiently large, the distortion will vanish. Note that the focal length is adjusted following Eq. (7) to maintain a similar eyes' position as the input image.

6 Workflow for Full Image

Since 3D GANs focus on cropped face regions to facilitate the training, we develop a geometry-aware stitching method (see Fig. 7) to extend the capability of our distortion correction technique to full images. Specifically, we reproject the full input image to the same viewpoint as the manipulated face patch. Then, the stitching tuning step fine-tunes the generator while freezing the inverted parameters to mitigate the discrepancy between the boundary regions of the face in the rendered patch and that of the reprojected input image. After that, we seamlessly blend the optimized face patch with the reprojected image.

The basic idea of stitching tuning comes from STIT (Tzaban et al., 2022) for 2D GAN.

However, applying STIT (Tzaban et al., 2022) directly is infeasible. Because the perspective-aware manipulation yields a face image I_{novel} with different camera parameters from the input full image I_{input}, leading to geometric *inconsistencies* between them. Merely fine-tuning the generator and then blending the generated face image and the input full image can reduce seams but introduce suspicious distortion, such as a disproportionately large face[3] and a slim neck. On the contrary, our method maps the manipulated face back into a reprojection of the input image.

6.1 Input Image Reprojection

Initially, we acquire the depth map $\mathbf{D}_{\text{input}}$ for the input image through a monocular depth estimator (Ranftl et al., 2020). However, this depth map has a different range with the 3D GAN rendered face depth map $\mathbf{d}_{\text{render}}$. Moreover, the accuracy of the monocular depth map in face regions $\mathbf{d}_{\text{input}} = \texttt{Crop}(\mathbf{D}_{\text{input}})$ is low. Therefore, we align the monocular depth to the GAN rendered depth map by minimizing a least square error:

$$\hat{s}, \hat{b} = \underset{s,b}{\arg\min} \|\big((s \times \mathbf{d}_{\text{input}} + b) - \mathbf{d}_{\text{render}}\big) \odot \Omega\|_2^2 \,, \tag{19}$$

where s and b represent scale and shift, $\odot$ is the element-wise multiplication, and Ω is the mask for the face region. The aligned depth is given by

$$\mathbf{D}_{\text{align}} = \hat{s} \times \mathbf{D}_{\text{input}} + \hat{b} \,. \tag{20}$$

[3] Here, we ensure the eyes' position constant. After correction, the disoccluded face regions will be recovered and make the face look larger than the distorted face.

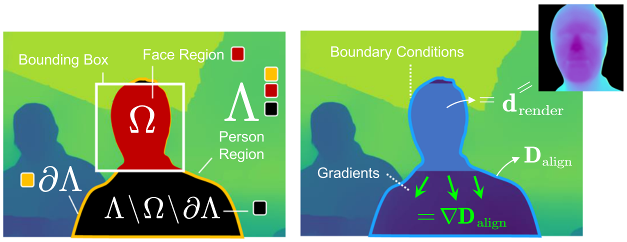


Fig. 8 Depth propagation. (*Left*) the illustration of notations. (*Right*) the illustration of the definition

Notice that the aligned depth is still worse than the rendered face depth due to the limitation of the monocular depth estimator. For example, the aligned depth in the face region is almost constant (see Fig. 7).

To refine $\mathbf{D}_{\text{align}}$, we replace its face region with the rendered depth and then propagate the depth from the face to other regions of this person, e.g., body, and hair, by solving a Poisson equation with Dirichlet boundary conditions (Pérez et al., 2003):

$$\begin{aligned} &\Delta \mathbf{D}_{\text{blend}} = \Delta \mathbf{D}_{\text{align}} \text{ over } \Lambda \backslash \Omega \text{ with} \\ &\mathbf{D}_{\text{blend}}|_{\Omega} = \mathbf{d}_{\text{render}}|_{\Omega}\,,\ \mathbf{D}_{\text{blend}}|_{\partial\Lambda} = \mathbf{D}_{\text{algin}}|_{\partial\Lambda}\,, \end{aligned} \tag{21}$$

where Δ is the Laplacian operator. The above equation seeks to minimize the gap between the gradient of the new depth $\mathbf{D}_{\text{blend}}$ and that of the aligned monocular depth $\mathbf{D}_{\text{align}}$ in the non-face regions $\Lambda \backslash \Omega \backslash \partial\Lambda$. At the same time, two conditions should hold: (i) $\mathbf{D}_{\text{blend}}$ in the face region equals rendered depth $\mathbf{d}_{\text{render}}$ and (ii) in the outer boundary region $\partial\Lambda$ of the person equal the aligned depth map $\mathbf{D}_{\text{align}}$. Figure 8 explains the notations and definitions.

With the refined depth map $\mathbf{D}_{\text{blend}}$, we then project the input image $I_{\text{warp}} = \mathcal{P}(I_{\text{input}}, \mathbf{D}_{\text{blend}}, \mathbf{c}_{\text{novel}})$ to a distant viewpoint $\mathbf{c}_{\text{novel}}$ the same as I_{novel}. In practice, we employ 3DP (Shih et al., 2020) to reproject the input image so that there will be no hole.

6.2 Stitch Tuning

Given the reprojected full image I_{warp}, we fine-tune the generator's weights, as depicted in Fig. 7b. We minimize the difference of border pixels between our refined face image and the warped full image while maintaining the integrity of our synthesis:

$$\begin{aligned} \arg\min_{\vartheta} &\left\| \left(R_{\boldsymbol{\theta}}(G_{\vartheta}(\hat{\mathbf{w}}), \mathbf{c}_{\text{novel}}) - I_{\text{warp}}^{\text{face}} \right) \odot \partial\Omega \right\|_2^2 + \\ &\left\| \left(R_{\boldsymbol{\theta}}(G_{\vartheta}(\hat{\mathbf{w}}), \mathbf{c}_{\text{novel}}) - I_{\text{novel}} \right) \odot \Psi \right\|_2^2\,, \end{aligned} \tag{22}$$

where $\partial\Omega$ masks out the boundary region of the face, Ψ denotes a mask for the interior region of the face, and $I_{\text{warp}}^{\text{face}} = \texttt{Crop}(I_{\text{warp}})$. The weight $\boldsymbol{\theta}$ is initialized to $\hat{\boldsymbol{\theta}}$. With the optimized parameter $\boldsymbol{\theta}^*$, we render the final image $I_{\text{novel}}^* = R_{\boldsymbol{\theta}}(G_{\boldsymbol{\theta}^*}(\hat{\mathbf{w}}), \mathbf{c}_{\text{novel}})$.

Finally, we blend the refined synthetic face image I_{novel}^* and the warped full image I_{warp} to produce an entire image I_{output} virtually captured at a long distance, as shown in Fig. 7c.

7 Experiments

7.1 Experimental Setup

Dataset We use three different datasets for evaluation:

- Caltech Multi-Distance Portraits (CMDP) (Burgos-Artizzu et al., 2014): This dataset contains portrait images of different people taken from various distances. It provides the same identities taken from different distances. We use the CMDP dataset for quantitative evaluations.
- USC perspective portrait database (Zhao et al., 2019): This database contains images with single faces with different levels of perspective distortions. There are no references or ground truth images, so we only use these images for visual comparisons.
- In-the-wild images: We also collect many in-the-wild photos online with severe perspective distortions on faces. We use these images for visual comparisons.

Compared methods We compare our method with:

- **Perspective undistortion methods:** Fried's (Fried et al., 2016) and Zhao's (Zhao et al., 2019) work targets the same task as addressed in this study, yet they employ 2D warping-based approaches. Since neither releases official implementations, we reproduce Fried's method (Fried et al., 2016). In addition to comparing with our implementation,
we also compare several results sourced from the website of Fried et al. (2016) and provided by the authors of Zhao et al. (2019).
- **Wide-angle undistortion methods:** Shih's Shih et al. (2019) work tackles a different undistortion problem: distortion caused by a wide-angle lens. Their basic idea is to apply the stereographic projection to the distorted image.
- **GAN inversion methods:** PTI (Roich & Mokady, 2021), Ko's Ko et al. (2023), HFGI3D Xie et al. (2023), and Triplanenet Bhattarai et al. (2024). Although not dealing with portrait perspective distortion correction, these

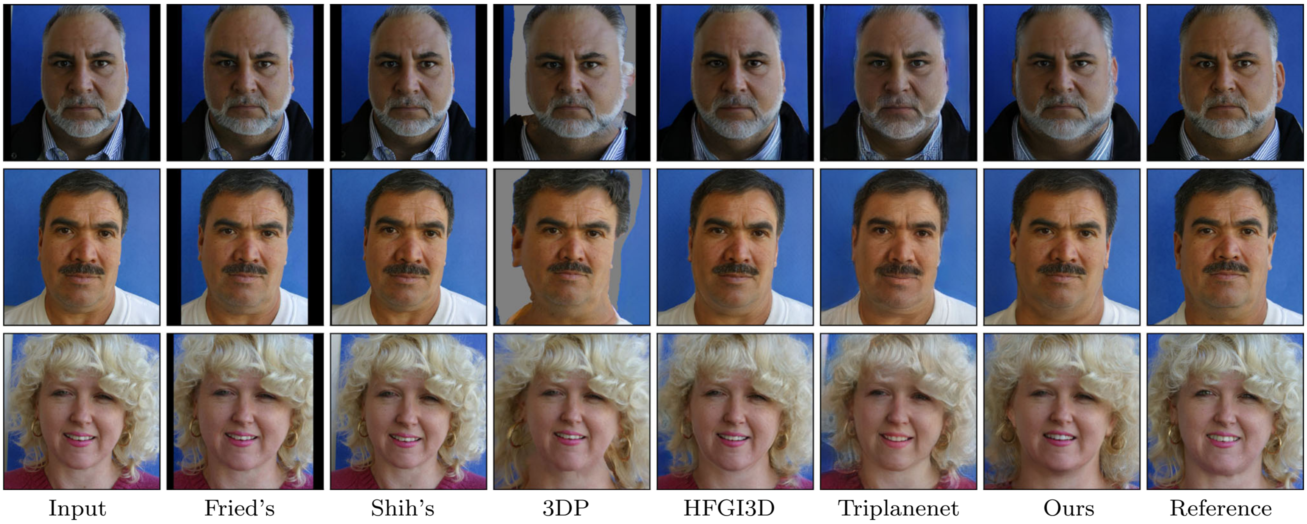


Fig. 9 Qualitative comparisons on the CMDP dataset (Burgos-Artizzu et al., 2014). Results of (Fried et al., 2016) are from their website. Our method renders faces closer to their references while preserving the identity

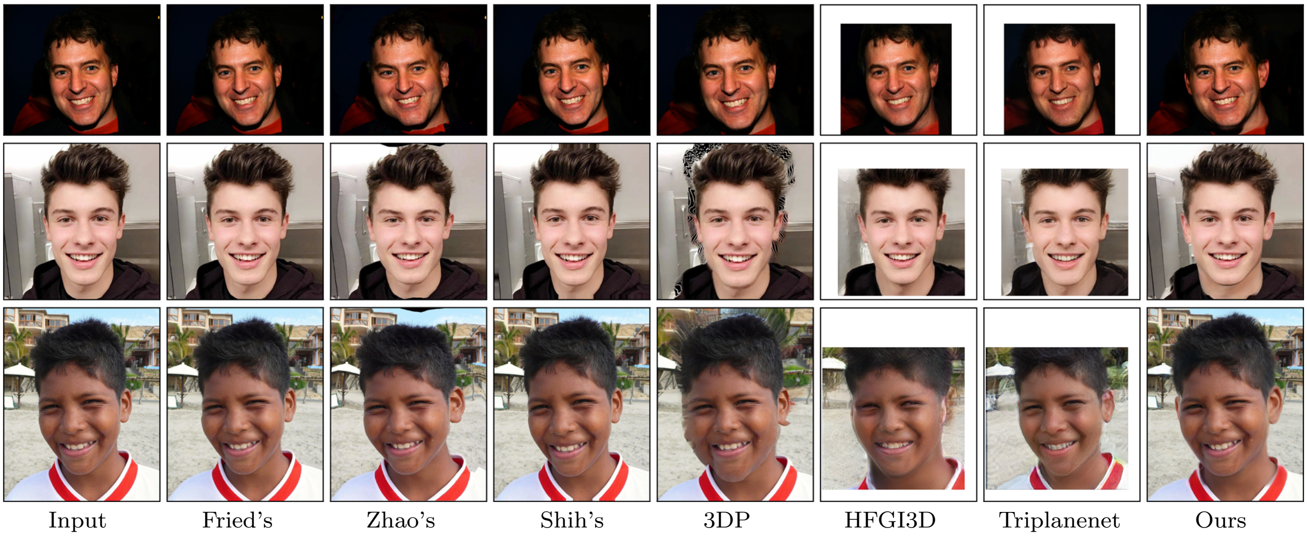


Fig. 10 Qualitative comparisons on images collected by (Zhao et al., 2019). Results of compared methods (Fried et al., 2016; Zhao et al., 2019) are from Zhao et al. (2019). Our method produces the least distorted and the most natural results. Note that with the help of 3D GAN, our method can generate the ear that originally occluded in the input images

GAN inversion methods enable 3D GANs to generate novel views from a single input image.

- **3D photography:** 3DP (Shih et al., 2020) is a method that can render novel views from a single RGB-D image.

Evaluation metrics We use five evaluation metrics to evaluate the performance of distortion correction:

- **Euclidean distance landmark error (LMK-E)**: We first align all output faces, and their corresponding reference face[4] according to the dense facial landmarks detected via Mediapipe (Lugaresi et al., 2019). We follow a similar alignment method by StyleGAN Karras et al. (2019) to align both images and their corresponding landmarks to a canonical pose. We then calculate the normalized landmark distance error in the 2D Euclidean space.
- **Image similarities PSNR, SSIM, and LPIPS**: We calculate image similarities between the aligned output images and corresponding references, including PSNR, SSIM

[4] Reference images are captured asynchronously with the input image, which may contain variations in expressions, lighting, poses, etc.

Table 1 Quantitative comparison on the CMDP dataset (Burgos-Artizzu et al., 2014).

Method	Type	LMK-E↓	PSNR↑	SSIM↑	LPIPS↓	ID↑
⋆Fried's (Fried et al., 2016)	W	0.175	15.41	*0.724*	*0.188*	**0.893**
†Fried's (Fried et al., 2016)	W	*0.165*	14.41	0.716	0.208	*0.860*
Shih's (Shih et al., 2019)	W	0.236	12.95	0.696	0.258	0.855
3DP (Shih et al., 2020)	W	0.195	13.08	0.696	0.268	0.847
PTI (Roich & Mokady, 2021)	G	0.191	*15.92*	0.717	0.197	0.758
Ko's (Ko et al., 2023)	G	0.180	15.41	0.710	0.206	0.689
HFGI3D (Xie et al., 2023)	G	0.177	15.75	*0.724*	0.198	0.829
Triplanenet (Bhattarai et al., 2024)	G	0.188	14.80	0.705	0.243	0.812
Ours	G	**0.138**	**17.52**	**0.747**	**0.167**	0.859

The best scores are highlighted in bold, while the second-best scores are in italics. We evaluate 43 faces projected from 60 to 480 cm. The PSNR and SSIM scores are low because reference images are captured asynchronously with different camera parameters from the inputs, resulting in different appearances and poses. 'W' represents warping-based and 'G' denotes GAN inversion-based. ⋆Results from the official website. †Our re-implementation. Although the results differ from the original ones, the metric scores are comparable

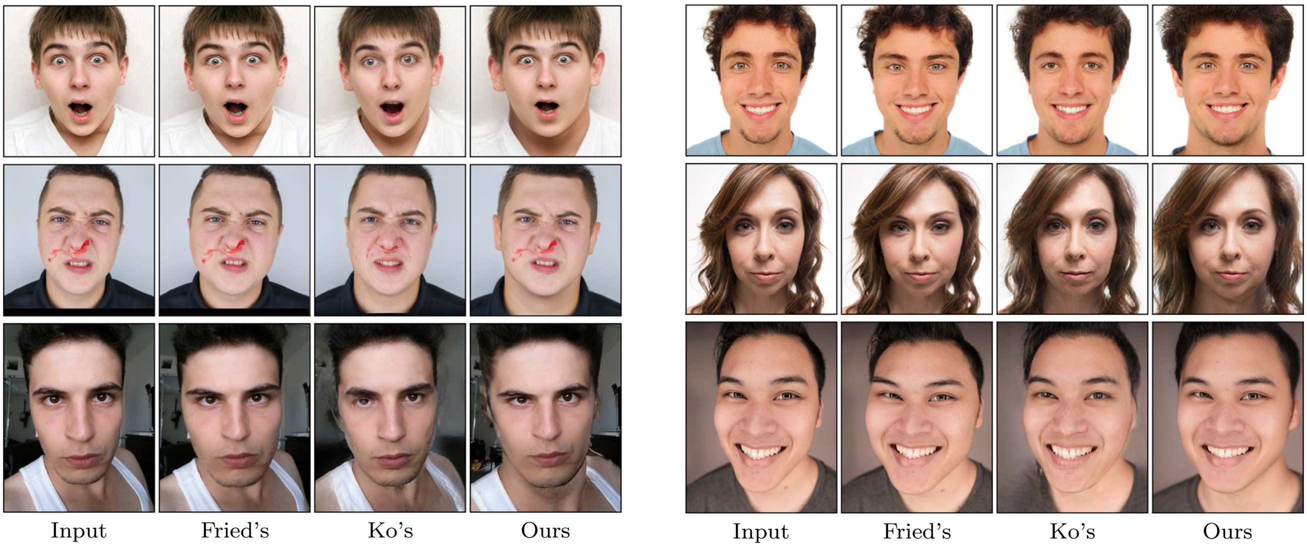


Fig. 11 Qualitative comparisons on our collected severely distorted in-the-wild images. Our method performs well in dealing with these seriously distorted faces and recovering occluded regions, such as ears

(Wang et al., 2004), and LPIPS (Zhang & Isola, 2018). We use a tri-map free matting algorithm (Ke et al., 2022) to remove the background and calculate the photometric distances on the masked foreground.

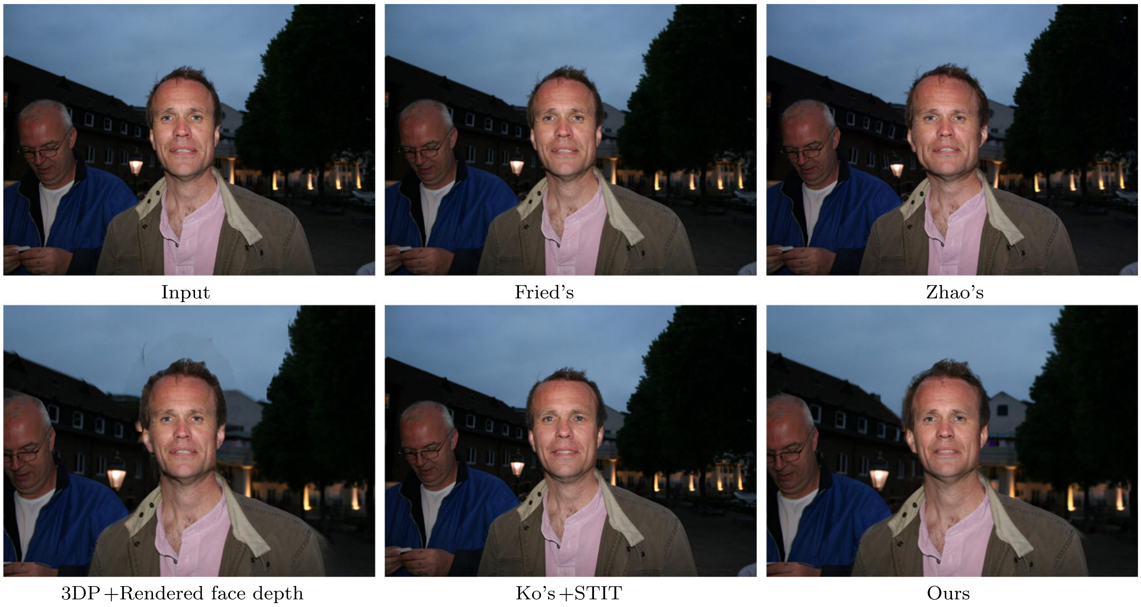
- **Identity similarity**: We use ArcFace (Deng et al., 2019) to extract features for the masked face foregrounds and compute the cosine distance between facial features of output images and reference images.

7.2 Quantitative Evaluation

We evaluate our method on the CMDP dataset (Burgos-Artizzu et al., 2014), and the results in Table 1 indicate: (1) Our method outperforms others in most metrics with a large margin; (2) All methods, including ours, exhibit inferior performance in identity preservation compared to the original version of Fried et al. (2016). This is primarily due to the significance of face details in calculating identity metrics. The original version of Fried et al. (2016) has subtle manipulations and retains many details. GAN inversion-based methods have the lowest identity score among all methods because they may lose some crucial details. (3) Despite the limitations of GAN inversion, our method achieves comparable results to our reimplementation of the warping-based method (Fried et al., 2016) in the identity metric.

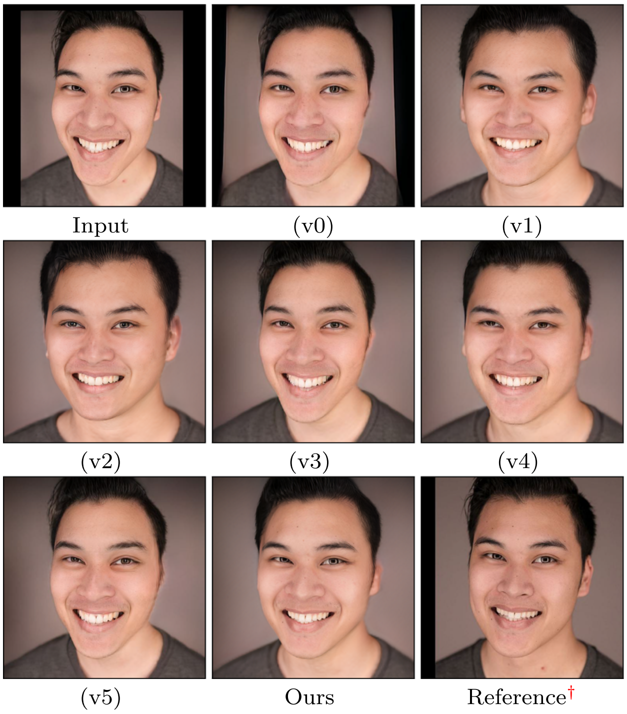
7.3 Qualitative Evaluation

We evaluate our proposed method on cropped face images used by previous methods, and the comparisons are presented


Input Fried's Zhao's

3DP +Rendered face depth Ko's+STIT Ours

Fig. 12 Comparison on in-the-wild full images. Results of compared methods (Fried et al., 2016; Zhao et al., 2019) are from Zhao et al. (2019). Our system produces a visually pleasing result with the least distortions. Note that our rendered face is harmonious with the body, but STIT (Tzaban et al., 2022) and Zhao's (Zhao et al., 2019) don't


Input (v0) (v1)

(v2) (v3) (v4)

(v5) Ours Reference†

Fig. 13 Qualitative results of ablation study. Our full model produces a visually pleasing result closest to the reference. It cannot perform well if any of these designs are removed. Although quantitative results in Table 2 seems to suggest that optimization scheduling is not dominant in our method, it is necessary to avoid sub-optimal results. †Note that the reference is not the ground truth

in Figs. 9 and 10. The changes to distorted faces introduced by Fried et al. (2016) and Shih et al. (2019) are infinitesimal. In contrast, evident changes can be observed when distorted faces are corrected by Zhao et al. (2019) and 3DP (Shih et al., 2020). However, their corrections lead to amplified distortions, where the middle part of faces is less distorted, but the head and chin shapes still appear peculiar (Fig. 10). Our method generates faces with fewer perspective distortions while preserving identity. Moreover, with the aid of 3D GAN, our approach can generate occluded parts present in the original input images, such as ears. It is worth noting that other GAN inversion-based solutions (Xie et al., 2023; Bhattarai et al., 2024) struggle to recover the correct face shape.

We further demonstrate this advantage on the collected **in-the-wild** images with severe distortions and showcase the distortion correction results in Fig. 11. We notice that the re-implemented method (Fried et al., 2016) performs similarly to Zhao et al. (2019). Additionally, we observe that the GAN inversion-based method (Ko et al., 2023) encounters local minima and generates faces with incorrect shapes. The visual results clearly demonstrate that our perspective-aware 3D GAN inversion proves to be an effective approach for correcting portrait perspective distortion, outperforming the warping-based method (Fried et al., 2016) and the existing 3D GAN inversion-based method (Ko et al., 2023).

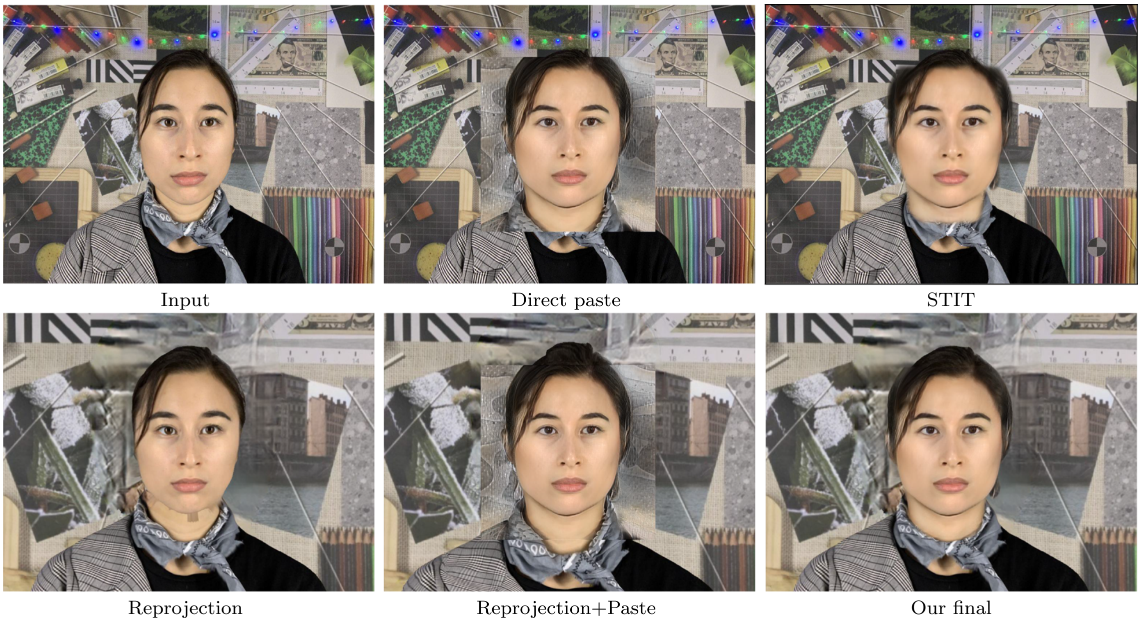


Fig. 14 Qualitative results for ablation study of geometric-aware stitching. 3D GANs can only reproject a cropped face image to a virtual far distance while leaving the rest of the image distorted. Pasting the modified face back into the original image can lead to inconsistencies between the cropped face and the untouched regions. This geometry inconsistency cannot be reduced by STIT (Tzaban et al., 2022) used by 2D GAN inversion/manipulation. To address this issue, we reproject the background and fine-tune the generator to achieve seamless blending

7.4 Full-Image Qualitative Evaluation

We validate our system's ability to process **in-the-wild full** images, as demonstrated by the visually pleasing results in Figs. 1 and 13. In comparison, other methods fail to reduce perspective distortion or generate harmonious results effectively. Specifically, (1) the changes caused by Fried's Fried et al. (2016) are subtle, and the manipulated face remains distorted. (2) Zhao's Zhao et al. (2019) significantly alters the face, but the result still exhibits an asymmetric face shape, weird head and chin shapes, and inconsistency between the body and face. (3) Although 3DP (Shih et al., 2020) can manipulate the body and somewhat mitigate face distortion by using the depth from 3D GAN, the face is still distorted. (4) Combining Ko's (Ko et al., 2023) and STIT (Tzaban et al., 2022) results in a seamless image but lacks harmony. On the other hand, our manipulated faces exhibit harmonious integration with corresponding bodies, with fewer distortions.

7.5 Video Evaluation

In comparing our method with others in rendering dolly-zoom videos from distorted input, the results on our website demonstrate that only our approach can consistently generate continuous dolly-zoom videos. In contrast, other methods show the following limitations: (1) Fried's (Fried et al., 2016) corrects distortion but performs worse than ours, with minimal manipulation in non-face regions. (2) 3DP (Shih et al., 2020) is unable to manipulate the face. (3) Combining Ko's (Ko et al., 2023) with STIT (Tzaban et al., 2022) leads to serious distortion.

7.6 Ablation Study

We conduct ablation studies on both the CMDP dataset and our collected seriously distorted face images. The results are presented in Table 2 and Fig. 12. Without camera optimization or any of our proposed designs for easing optimization, the face parameter gets stuck in a sub-optimal solution, leading to poor performance. The proposed focal length reparameterization and distance initialization are crucial for achieving good results, and removing any of them results in a significant degradation in performance, with the reconstructed face geometry being wrong and the corrected image remaining distorted as the input. While removing optimization scheduling, rotation reparameterization and camera optimization can still correct the distortion to some extent, it is more prone to fall into a local minimum, generating a face far away from the reference. The rotation reparameterization reduces the degree of freedom and regularizes the orthogonality of the camera rotation matrix.

We also conduct ablation studies on the full-image pipeline to investigate the stitching post-processing, as shown in Fig. 14. When we directly paste the manipulated face into the input image, it results in an inconsistency between the face and body parts. However, we can achieve seamless blend-

Table 2 Quantitative results of ablation study

	cam opt	rot. repa.	focal repa.	schedule	closeup	LMK-E↓	LPIPS↓
low bound (input)	–	–	–	–	–	0.227	0.249
(v0): w/o all	×	×	×	×	×	0.190	0.198
(v1): w/o cam. opt.	×	–	✓	–	✓	0.159	0.204
(v2): w/o rot. repa.	✓	×	✓	✓	✓	0.167	0.203
(v3): w/o focal repa.	✓	✓	×	✓	✓	0.183	0.200
(v4): w/o opt. sche.	✓	✓	✓	×	✓	0.151	0.182
(v5): w/o closeup cam	✓	✓	✓	✓	×	0.185	0.198
Ours	✓	✓	✓	✓	✓	**0.138**	**0.167**

Focal length reparameterization and distance initialization are crucial. Removing any of them (v3 and v5) significantly degrades performance. Optimization scheduling is important to avoid sub-optimal results. Discarding camera optimization yields the worst LPIPS. Our method achieves the best performance

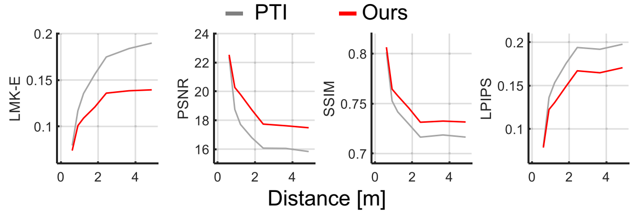


Fig. 15 Evaluation of rendering at different camera-to-subject distances. We project the input distorted images to various distances, with the result at each distance being an average of 43 faces. Notably, our method outperforms PTI (Roich & Mokady, 2021) by a significant margin when the camera-to-subject distance is large

ing with the geometric-aware stitching method, producing a more harmonious and natural result.

7.7 Manipulation to Different Distances

We evaluate our method's ability to render images across various camera-to-subject distances using the CMDP (Burgos-Artizzu et al., 2014) dataset. This dataset comprises images of subjects captured from seven distinct distances. We select the closest image for each subject as our input and then project it into the remaining six distances. As shown in Fig. 15, our method consistently outperforms the baseline PTI (Roich & Mokady, 2021) across all distances, with its superiority increasing as the distance grows.

7.8 User Study

We conduct two user studies to compare our perspective 3D GAN inversion method with conventional GAN inversion method PTI (Roich & Mokady, 2021) with estimated cameras. In the first study, we presented results on 15 CMDP images alongside reference images to 56 participants and asked them to identify which method yields an image that closely resembles the reference. In the second study, we showed results on 10 in-the-wild images to 25 users and asked which method produces a less distorted image. Results in Fig. 18 demonstrate that our method consistently outperforms PTI (Roich & Mokady, 2021) in correcting distortion. However, we also find that in some instances, PTI (Roich & Mokady, 2021) performs better because the input faces in these cases have lower distortion levels.

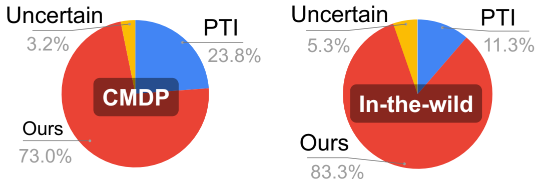


Fig. 16 User study. We conducted two user studies, one on the CMDP dataset (Burgos-Artizzu et al., 2014) and another on our collected in-the-wild dataset. User prefer our results than PTI (Roich & Mokady, 2021)

7.9 Visualization of Inversion Process

In Fig. 16, we visualize the optimization process. We verify that without our perspective-aware designs, 3D GAN inversions often get trapped in local minima and fail to reconstruct the correct face geometry or correct the perspective distortion. Our proposed method overcomes these limitations and produces more accurate geometries (Fig. 17) and visually pleasing results.

7.10 Bonus Features

Thanks to the generative ability of 3D GANs, our method enjoys additional advantages over warping-based methods in face completion and semantic editing.

Face completion Figure 19 demonstrates that our method can effectively correct the distortion in partially occluded faces. This capability is beneficial for seriously distorted

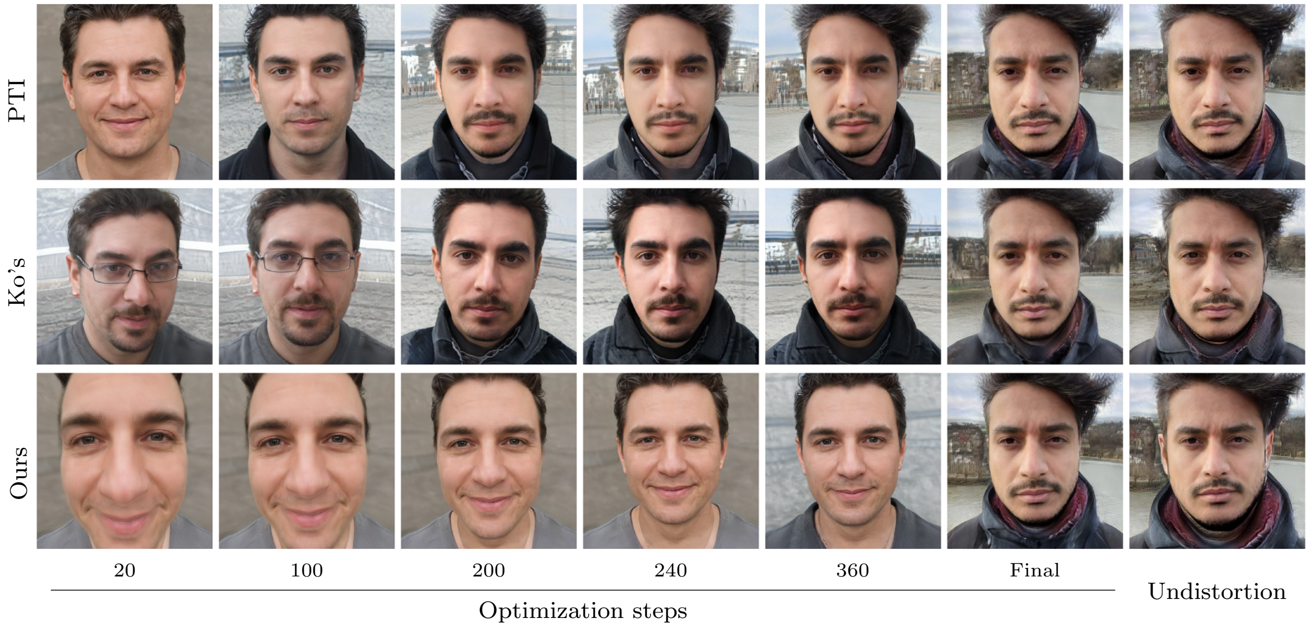


Fig. 17 Visualization of optimization. Our method first optimizes the camera parameters and then jointly optimizes the camera parameters and the face latent code. In contrast, PTI (Roich & Mokady, 2021) and Ko's (Ko et al., 2023) optimize the face latent code while maintaining a fixed, incorrect camera-to-subject distance, making them susceptible to local minima, resulting in inaccurate shapes, such as those lacking ears

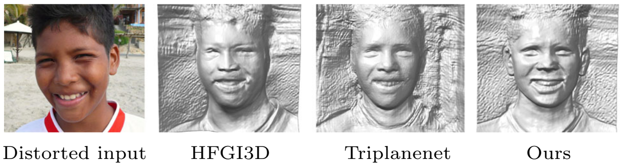


Fig. 18 Face shape recovered by 3D GANs. The face geometric shape recovered by our method has less distortion than other approaches

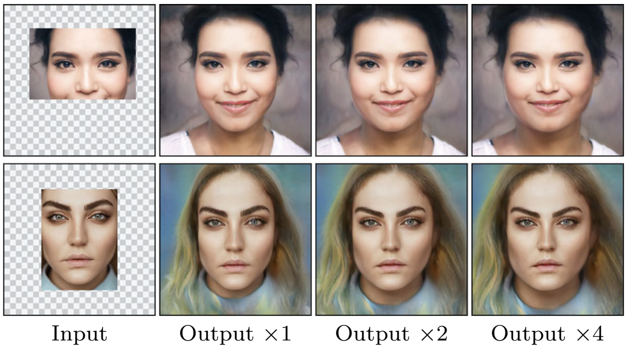


Fig. 19 Face completion. Our method can apply directly to partially-completion faces and does not expect a well-processed face

faces near image boundaries, which cannot be handled by warping-based methods like Fried's (Fried et al., 2016) due to the absence of face landmarks, or Zhao's (Zhao et al., 2019), which cannot generate occluded regions.

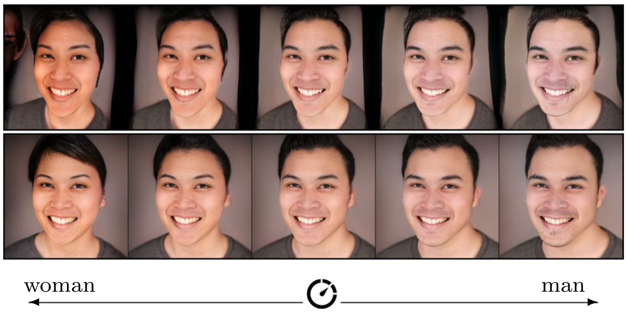


Fig. 20 Editing ability. Our method (*bottom*) improves the editing ability of 3D GAN on perspective-distorted faces. Without our method (*top*), inverting the input distorted face leads to an *out-of-distribution* face latent code. Consequently, it leads to poor editing quality. On the other hand, our method inverts an *in-distribution* face latent code that enables us to edit. It facilitates downstream applications

GAN editing Figure 20 shows that our method improves the editing ability of 3D GAN on perspective-distorted input face images. Inverting the input distorted face with PTI (Roich & Mokady, 2021) can lead to an out-of-distribution facial latent code. Editing these latent codes could generate unwanted artifacts. Instead, our method inverts the image to an in-distribution face latent code that can be edited more accurately.

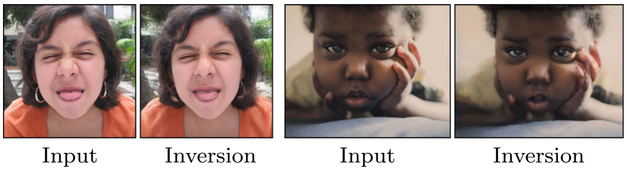


Fig. 21 Failure cases. Limited by 3D GANs, our method cannot handle out-of-distribution faces, e.g., tongue outside the mouth (*left*), hand touching face (*right*). A potential solution is first to mask these regions for GAN inversion. Then, transfer the textures to the manipulated face

7.11 Limitations

While the proposed method has shown promising performance, we acknowledge its limitations in this section.

Out-of-distribution faces As shown in Fig. 21, our method fails for out-of-distribution faces, including extreme expressions and occluded faces (by hand or other objects). In these cases, GAN inversion struggles to comprehend the face and may generate the face based on its own interpretation (e.g., the left example in Fig. 21 where the tongue is mistaken as part of the lip in the output). This can result in dreadful artifacts, as seen in the right example of Fig. 21, where the hand looks distorted in the output. A potential solution is first to mask these regions for GAN inversion, then transfer the textures to the manipulated face.

Inference speed We recognize that the current system does not operate in real time. Specifically, the GAN inversion process takes approximately 130 seconds to process a cropped face. This is because we implement our method based on the optimization-based inversion. The time required for optimization is in line with PTI (Roich & Mokady, 2021). However, recent advancements (Trevithick et al., 2023; Yuan et al., 2023; Bhattarai et al., 2024) explored the *encoder-based* inversions for 3D GANs have successfully reduced inference times to less than 1 second. These methods hold the potential to be seamlessly integrated into our perspective-aware 3D GAN inversion, significantly enhancing inference speed. Additionally, the encoder-based approach can overcome our current limitation of optimizing each individual photo. Applying these encoder-based methods to our task would require training the encoder with paired perspective-distorted and ground-truth undistorted images. We leave the extension of speed improvement to future work.

8 Conclusions

We present a method for portrait perspective distortion correction. Our core idea is to leverage a 3D GAN inversion method to recover plausible facial geometry and reveal hidden facial parts such as ears. We explore several design choices such as closeup camera-to-face distance initialization, optimization scheduling, reparameterizations, and landmark constraints. We propose a geometric-aware stitching method to extend our model to full images. Furthermore, we establish a protocol of quantitative evaluation for the portrait perspective distortion correction. Quantitative and visual comparisons demonstrate the improved performance of our pipeline over existing methods.

Appendix: Implementation Details

3D GAN: Our experiments employ the EG3D model (Chan et al., 2022) pre-trained on the FFHQ dataset (Karras et al., 2019). Our method, however, is agnostic to the underlining 3D GAN models. For example, other 3D GANs such as IDE-3D (Sun et al., 2022) could also be used.

Optimization: We use the Adam optimizer. We set learning rates: $\lambda_{\text{cam}} = 1 \times 10^{-2}$, $\lambda_{\text{face}} = 5 \times 10^{-3}$, $\lambda_{\text{gan}} = 3 \times 10^{-4}$, and $\lambda_{\text{tiny}} = 0.1$. We let the parameter ϵ equal 0.5. We set the rendering parameters `ray_start` and `ray_end` to `auto` for close-up faces

Masked loss: Close-up portraits often have faces that extend close to the image boundary, creating issues with the crop operation and potentially causing the cropped images to have incomplete faces and black boundaries. As a result, directly fitting such images may yield unusual facial features. To address this concern, we implement a **masked loss** to ignore the black boundaries.

Background inpainting: As 3DP (Shih et al., 2020) may not sufficiently reveal the hidden background and could result in undesirable gaps, we first use Stable Diffusion (Rombach et al., 2022) or DALL·E2 to inpaint the background when processing full-frame input images. We then reproject the inpainted background and utilize it to replace the background in our rendered full-frame image. For this task, we leverage MODNet (Ke et al., 2022) to separate the person from the background.

Texture transfer: Note that if the inverted face loses details, we can alleviate such artifacts by warping the residual between input and inversion using the face geometry, and then adding it to the final images.

Acknowledgements We thank Yajie Zhao and Ohad Fried for providing their results and/or dataset, Fangzhou Mu, Dejia Xu, Lixiong Chen, and Yinqiang Zheng for valuable discussions and suggestions. We also thank anonymous people who attended our user study. This work is supported by JST AIP Acceleration Research JPMJCR22U4, by JSPS KAKENHI JP22H03620 and JP22H05015, and by Value Exchange Engineering, a joint research project between Mercari, Inc. and RIISE.

References

Abdal, R., Qin, Y., Wonka, P. (2019). Image2StyleGAN: How to embed images into the styleGAN latent space?. In *ICCV*.

Abdal, R., Zhu, P., Mitra, N J., & Wonka, P. (2022). Video2StyleGAN: Disentangling local and global variations in a video. arXiv preprint arXiv:2205.13996.

Alaluf, Y., & Patashnik, O., and Daniel C.-O. (2021). Restyle: A residual-based styleGAN encoder via iterative refinement. In *ICCV*.

Athar, S., Xu, Z., Sunkavalli, K., Shechtman, E., Shu, Z. (2022). Rignerf: Fully controllable neural 3D portraits. In *CVPR*.

Atzmon, M., Lipman, Y. (2020). SAL: Sign agnostic learning of shapes from raw data. In *CVPR*.

Bhattarai, A. R., Nießner, M., & Sevastopolsky, A. (2024). Triplanenet: An encoder for EG3D inversion. In *WACV*.

Bryan, R., Perona, P., & Adolphs, R. (2012). Perspective distortion from interpersonal distance is an implicit visual cue for social judgments of faces. *PLoS One, 7*(9), e45301.

Burgos-Artizzu, Xavier P., Ronchi, Matteo Ruggero, Perona, Pietro. (2014). Distance estimation of an unknown person from a portrait. In *ECCV*.

Chan, E. R., Lin, C. Z., Chan, M. A., Nagano, K., Pan, B., De Mello, S., Gallo, O., Guibas, L. J., Tremblay, J., Khamis, S., et al. (2022). Efficient geometry-aware 3D generative adversarial networks. In *CVPR*.

Chan, E. R., Lin, C. Z., Chan, M. A., Nagano, K., Pan, B., De Mello, S., Gallo, O., Guibas, L. J., Tremblay, J., Khamis, S. et al. (2022). Efficient geometry-aware 3D generative adversarial networks. In *CVPR*.

Chan, E. R., Monteiro, M., Kellnhofer, P., Wu, J., & Wetzstein, G. (2021). pi-GAN: Periodic implicit generative adversarial networks for 3D-aware image synthesis. In *CVPR*.

Chen, Z., & Zhang, H. (2019). Learning implicit fields for generative shape modeling. In *CVPR*

Cooper, E. A., Piazza, E. A., & Banks, M. S. (2012). The perceptual basis of common photographic practice. *Journal of Vision, 12*, 8.

Creswell, A., & Bharath, A. A. (2018). Inverting the generator of a generative adversarial network. *IEEE Transactions on Neural Networks and Learning Systems, 30*, 1967.

Deng, J., Guo, J., & Xue, N. and Stefanos Z. (2019). Arcface: Additive angular margin loss for deep face recognition. In *CVPR*.

Deng, Y., Yang, J., Xiang, J., Tong, X. (2022). Gram: Generative radiance manifolds for 3D-aware image generation. In *CVPR*.

Deng, Y., Yang, J., Xu, S., Chen, D., Jia, Y., & Xin T. (2019). Accurate 3D face reconstruction with weakly-supervised learning: From single image to image set. In *CVPRW*.

Fried, O., Shechtman, E., Goldman, D. B., Finkelstein, A. (2016). Perspective-aware manipulation of portrait photos. *ACM TOG (Proc. SIGGRAPH)*.

Gafni, G., Thies, J., Zollhofer, M., Nießner, M. (2021). Dynamic neural radiance fields for monocular 4d facial avatar reconstruction. In *CVPR*.

Gao, C., Shih, Y., Lai, W.-S., Liang, C.-K., Huang, J.-B. (2020). Portrait neural radiance fields from a single image. arXiv preprint arXiv:2012.05903.

Gropp, A., Yariv, L., Haim, N., Atzmon, M., & Lipman, Y. (2020). Implicit geometric regularization for learning shapes. In *ICML*

Guan, S., Tai, Y., Ni, B., Zhu, F., Huang, F., Yang, X. (2020). Collaborative learning for faster styleGAN embedding. arXiv preprint arXiv:2007.01758.

Jeong, Y., & Ahn, S. (2021). *Christopher Choy*. Anima Anandkumar: Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. In ICCV.

Karras, T., Laine, S., Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *CVPR*.

Karras, T., Laine, S., Aittala, M., Hellsten, M., Lehtinen, J., & Aila, T. (2020). Analyzing and improving the image quality of StyleGAN. In *CVPR*.

Ke, Z., Sun, J., Li, K., Yan, Q., Lau, R. W.H. (2022). MODNet: Real-time trimap-free portrait matting via objective decomposition. In *AAAI*.

Ko, J., Cho, K., Choi, D., Ryoo, K., & Kim, S. (2023). 3D GAN inversion with pose optimization. In *WACV*.

Lin, C. Z., Lindell, D. B., Chan, E. R.,& Wetzstein, G. (2022). 3D GAN inversion for controllable portrait image animation. In *ECCVW*.

Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.-L, Yong, M. G., Lee, Juhyun, et al. (2019). Mediapipe: A framework for building perception pipelines. arXiv preprint arXiv:1906.08172.

Mescheder, L., Oechsle, M., Niemeyer, M., Nowozin, S., Geiger, A. (2019). Occupancy networks: Learning 3D reconstruction in function space. In *CVPR*.

Meuleman, A., Liu, Y.-L., Gao, C., Huang, J.-B., & Kim, C., M.H., Kim, and Johannes K. (2023). In CVPR: Progressively optimized local radiance fields for robust view synthesis.

Michalkiewicz, M., Pontes,JK. Jack, D., Baktashmotlagh, M. and Eriksson, Anders. (2019). In ICCV: Implicit surface representations as layers in neural networks.

Mildenhall, B.,Srinivasan, P. P., Tancik, M., Barron, J. T., Ramamoorthi, R., & Ng, R. (2020). NeRF: Representing scenes as neural radiance fields for view synthesis. In *ECCV*.

Nagano, K., Luo, H., Wang, Z., Seo, J., Xing, J., Hu, L., Wei, L., & Li, H. (2019). *Deep face normalization. ACM TOG, 38*(6), 1–16.

Niemeyer, M., Geiger, A. (2021). Giraffe: Representing scenes as compositional generative neural feature fields. In *CVPR*.

Or-El, R., Luo, X., Shan, M., Shechtman, E., Park, J. J., Kemelmacher-Shlizerman, I. (2022). StyleSDF: High-resolution 3D-consistent image and geometry generation. In *CVPR*.

Park, J. J.,, Florence, P., Straub, J., Newcombe, R., & Lovegrove, S. (2019). Deepsdf: Learning continuous signed distance functions for shape representation. In *CVPR*.

Peng, S., Niemeyer, M., & Lars, M. (2020). *Mescheder: Marc Pollefeys, and Andreas Geiger*. In ECCV: Convolutional occupancy networks.

Pérez, P., Gangnet, M., & Blake, A. (2003). Poisson image editing. In *SIGGRAPH*.

Ponimatkin, G., Labbé, Y., Russell, B., Aubry, M., & Sivic, J. (2022). Focal length and object pose estimation via render and compare. In *CVPR*.

Ranftl, R., Lasinger, K., Hafner, D., Schindler, K., & Koltun, V. (2020). Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 44*(3), 1623–1637.

Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., Cohen-Or, D. (2021). Encoding in style: A styleGAN encoder for image-to-image translation. In *CVPR*.

Roich, D., & Mokady, R.,A. H., Bermano & Daniel C.-O. (2021). ACM TOG: Pivotal tuning for latent-based editing of real images.

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. In *CVPR*.

Shih, M.-L., Su, S.-Yang, K., Johannes, H., Jia-Bin. (2020). 3D photography using context-aware layered depth inpainting. In *CVPR*.

Shih, Y., Lai, W.-S., & Liang, C.-K. (2019). Distortion-free wide-angle portraits on camera phones. *ACM Transactions on Graphics (TOG), 38*(4), 1–12.

Sitzmann, V., Zollhöfer, M., Wetzstein, G. (2019). Scene representation networks: Continuous 3D-structure-aware neural scene representations. In *NeurIPS*.

Sun, J., Wang, X., Shi, Y., Wang, L., Wang, J., Liu, Y. (2022). IDE-3D: Interactive disentangled editing for high-resolution 3D-aware portrait synthesis. *ACM TOG (Proc. SIGGRAPH Asia)*.

Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., & Cohen-Or, D. (2021). Designing an encoder for styleGAN image manipulation. *ACM TOG (Proc. SIGGRAPH)*.

Trevithick, A., Chan, M., Stengel, M., Chan, E., Liu, C., Yu, Z., Khamis, S., Chandraker, M., Ramamoorthi, R., & Nagano, K. (2023). Real-time radiance fields for single-image portrait view synthesis. *ACM TOG, 42*(4), 1–15.

Tzaban, R., Mokady, R., Gal, R., Bermano, A., & Cohen-Or, D. (2022). Stitch it in time: GAN-based facial editing of real videos. In *SIGGRAPH Asia*.

Valente, J., Soatto, S. (2015). Perspective distortion modeling, learning and compensation. In *CVPRW*.

Wang, P., Liu, L., Liu, Y., Theobalt, C., Komura, T., Wang, W. (2021). NeuS: Learning neural implicit surfaces by volume rendering for multi-view reconstruction. In *NeurIPS*.

Wang, Y., Xu, T., Wu, Y., Li, M., Chen, W., Xu, L., Yu, J. (2022). Narrate: A normal assisted free-view portrait stylizer. arXiv preprint arXiv:2207.00974.

Wang, Z., Bovik, A. C., Sheikh, H. R., Simoncelli, E. P. (2004). Image quality assessment: from error visibility to structural similarity. *IEEE TIP*.

Ward, B., Ward, M., Fried, O., & Paskhover, B. (2018). Nasal distortion in short-distance photographs: The selfie effect. *JAMA Facial Plastic Surgery, 20*, 333–335.

Williams, K. K., & Motta, R. Camera field of view effects based on device orientation and scene content, Jul 18 2017. US Patent 9,712,751.

Xie, J., Ouyang, H., Piao, J., Lei, C., Chen, Q. (2023). High-fidelity 3D GAN inversion by pseudo-multi-view optimization. In *CVPR*.

Xu, Y., AlBahar, B., Huang, J.-B. (2022). Temporally consistent semantic video editing. In *ECCV*.

Xu, Y., Shu, Z., Smith, C., Huang, J.-B., & Oh, S. W. (2023). In-n-out: Face video inversion and editing with volumetric decomposition. arXiv preprint arXiv:2302.04871.

Yuan, Z., Zhu, Y., Li, Y., Liu, H., & Yuan, C. (2023). Make encoder great again in 3D GAN inversion through geometry and occlusion-aware encoding. In *ICCV*.

Zhang, R., & Isola, P.,Efros, A. A. & Shechtman, E. and Oliver W. (2018). In CVPR: The unreasonable effectiveness of deep features as a perceptual metric.

Zhao, Y., Huang, Z., Li, T., & Chen, W., Chloe L., Xinglei R., Ari S., and Hao L. (2019). Learning perspective undistortion of portraits. In *ICCV*.

Zhou, P., Xie, L., Ni, B., & Tian, Q. (2021). CIPS-3D: A 3D-aware generator of GANs based on conditionally-independent pixel synthesis. arXiv preprint arXiv:2110.09788.

Zhu, P. J-Y., Eli S., Krähenbühl, & Efros, A. A. (2016). Generative visual manipulation on the natural image manifold. In *ECCV*.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.