Learning 3D-aware Image Synthesis with Unknown Pose Distribution

Zifan Shi†1 Yujun Shen†2 Yinghao Xu*3 Sida Peng4 Yiyi Liao4 Sheng Guo2
Qifeng Chen1 Dit-Yan Yeung1

1HKUST 2Ant Group 3CUHK 4Zhejiang University

Abstract

Existing methods for 3D-aware image synthesis largely depend on the 3D pose distribution pre-estimated on the training set. An inaccurate estimation may mislead the model into learning faulty geometry. This work proposes PoF3D that frees generative radiance fields from the requirements of 3D pose priors. We first equip the generator with an efficient pose learner, which is able to infer a pose from a latent code, to approximate the underlying true pose distribution automatically. We then assign the discriminator a task to learn pose distribution under the supervision of the generator and to differentiate real and synthesized images with the predicted pose as the condition. The pose-free generator and the pose-aware discriminator are jointly trained in an adversarial manner. Extensive results on a couple of datasets confirm that the performance of our approach, regarding both image quality and geometry quality, is on par with state of the art. To our best knowledge, PoF3D demonstrates the feasibility of learning high-quality 3D-aware image synthesis without using 3D pose priors for the first time. Project page can be found here.

1. Introduction

3D-aware image generation has recently received growing attention due to its potential applications [3, 4, 9, 21, 30, 34, 47]. Compared with 2D synthesis, 3D-aware image synthesis requires the understanding of the geometry underlying 2D images, which is commonly achieved by incorporating 3D representations, such as neural radiance fields (NeRF) [2, 16, 17, 24, 25, 50], into generative models like generative adversarial networks (GANs) [8]. Such a formulation allows explicit camera control over the synthesized results, which fits better with our 3D world.

To enable 3D-aware image synthesis from 2D image collections, existing attempts usually rely on adequate camera pose priors [3, 4, 9, 20, 21, 30, 47] for training.

† indicates equal contribution.
* This work was done during an internship at Ant Group.
In this work, we present a new approach for 3D-aware image synthesis, which removes the requirements for pose priors. Typically, a latent code is bound to the 3D content alone, where the camera pose is independently sampled from a manually designed distribution. Our method, however, maps a latent code to an image, which is implicitly composed of a 3D representation (i.e., neural radiance field) and a camera pose that can render that image. In this way, the camera pose is directly inferred from the latent code and jointly learned with the content, simplifying the input requirement. To facilitate the pose-free generator better capturing the underlying pose distribution, we re-design the discriminator to make it pose-aware. Concretely, we tailor the discriminator with a pose branch which is required to predict a camera pose from a given image. Then, the estimated pose is further treated as the conditional pseudo label when performing real/fake discrimination. The pose branch in the discriminator learns from the synthesized data and its corresponding pose that is encoded in the latent code, and in turn use the discrimination score to update the pose branch in the generator. With such a loop-back optimization process, two pose branches can align the fake data with the distribution of the dataset.

We evaluate our pose-free method, which we call PoF3D for short, on various datasets, including FFHQ [12], Cats [51], and Shapenet Cars [5]. Both qualitative and quantitative results demonstrate that PoF3D frees 3D-aware image synthesis from hyper-parameter tuning on pose distributions and dataset labeling, and achieves on par performance with state-of-the-art in terms of image quality and geometry quality.

2. Related Work

3D-aware Image Synthesis. 3D-aware image synthesis has achieved remarkable success recently [33, 43] and benefits lots of 3D-aware applications [39, 44, 49, 55] rather than the conventional 2D applications [1, 26, 27, 38, 40, 41, 54]. Different from 2D synthesis [11–13, 48], some works [18, 19, 56] propose to adopt voxels as 3D representation for image rendering but suffer from the poor image quality and consistency due to the voxel resolution restriction. Then a series of works [4, 6, 23, 30, 45] introduce neural implicit function [16, 17, 24] as the underlying 3D representation for image rendering. However, rendering high-resolution images with direct volume rendering is very heavy. Lots of works [3, 9, 21, 22, 31, 37, 46, 47, 52] resort to either 2D convolutional upsamplers [3, 9, 21, 22, 47], multi-plane image rendering [52], sparse-voxel [31] inference, patch-based training [37] for acceleration. Besides, some works [23, 35] focus on the improvement of geometry quality. Although these methods are able to synthesize high-quality 3D-aware images, they are restricted to strong pose prior, i.e., manually tuned pose distribution, or well-annotated camera poses [3]. Our work instead naturally enables 3D-aware image synthesis without any pose prior via a pose-free generator, which gives a fruitful avenue for future work.

Camera Learning from 2D Images. Learning neural radiance fields (NeRF) [17] requires accurate pose annotations. To overcome this, some works [14, 42] try to optimize the annotated camera parameters or directly estimate camera poses with a small network, which is conceptually comparable to ours. On the other hand, our work differs from theirs in that our method focuses on learning pose distribution of a real dataset rather than exact camera poses. CAMPARI [20] and 3DG [36] investigate a similar problem, aiming to estimate the pose distribution of an actual dataset. However, both of them need a manually designed prior, i.e., uniform or Gaussian, and fail to learn appropriate 3D geometry when the prior is very distant from the native distribution. In contrast, our work abandons all the manually designed prior and allows the generator to learn pose distribution automatically via adversarial training.

3. Method

Our proposed method for 3D-aware image synthesis can be freed from the requirement of pose priors. To achieve this goal, we propose some new designs in both the generator and the discriminator in conventional 3D-aware GANs [3, 4, 9, 30]. Specifically, the pose-free generator is equipped with a pose learner that infers the camera pose from the latent code. The pose-aware discriminator extracts a pose from the given image and uses it as the conditional label when performing real/fake classification. The framework is shown in Fig. 2. Before going into details, we first briefly introduce the generative neural radiance field, which plays a crucial role in 3D-aware image synthesis.

3.1. Preliminary

A neural radiance field (NeRF) [17], \( F(x, d) \rightarrow (c, \sigma) \), provides color \( c \in \mathbb{R}^3 \) and volume density \( \sigma \in \mathbb{R} \) from a coordinate \( x \in \mathbb{R}^3 \) and a viewing direction \( d \in S^2 \), typically parameterized with multi-layer perceptron (MLP) networks. Then, the pixel values are accumulated from the colors and densities of the points on the sampled rays. However, NeRF is highly dependent on multi-view supervision, leading to the inability to learn from a single-view image collection. To enable random sampling from single-view captures, recent attempts propose to condition NeRF with a latent code \( z \), resulting in their generative forms [4, 30], \( G(x, d, z) \rightarrow (c, \sigma) \), to achieve diverse 3D-aware generation.

3.2. Pose-free Generator

Different from NeRF, the camera pose \( \xi \) deriving the spatial point \( x \) and view direction \( d \) in conventional 3D-aware
generators is randomly sampled from a prior distribution $p_\xi$ rather than well annotated by MVS [32] and SfM [29]. Such a prior distribution requires the knowledge of the pose distribution, and it should be tailored for different datasets, which introduces non-trivial hyperparameter tuning for model training. This generation process can be formulated as

$$G(z, \xi) = I_f \sim p_\theta(I_f|z, \xi),$$

where the generator $G$ synthesizes the image $I_f$ by modeling the conditional probability given the latent code and the camera pose. Since the camera pose is independent of the latent code, we can rewrite Eq. (1) in the following form:

$$p_\theta(I_f|z, \xi) = \frac{p_\theta(I_f|z, \xi)}{p(z, \xi)} = \frac{p_\theta(I_f, \xi|z)}{p(\xi)} = \frac{p_\theta(I_f, \xi|z)}{p_\xi(\xi)},$$

(2)

Obviously, if the pose prior does not align well with the real distribution, it is hard to correctly estimate the image distribution $p_\theta(I_f|z, \xi)$. As noticed by [9, 47], it always makes the training diverge if the prior does not align well with the real distribution. To free the generator from sampling poses from a prior distribution, we follow the formulation of conventional 2D GANs that synthesizes images from the latent code only:

$$G(z, \Psi(z)) = I_f \sim p_\theta(I_f|z, \Psi(z)),$$

(3)

where the camera pose $\xi$ is parameterized with a nonlinear function $\Psi(\cdot)$ that takes $z$ as input. Concretely, we implement $\Psi(\cdot)$ by introducing an additional pose branch on the top of the generator. The estimated camera pose is further fed into a generative radiance field to render a 2D image. Compared with previous solutions, our generator aims at approximating the conditional probability only from the latent observation, which is prone to simulating the native data distribution without taking the pose prior into account.

### 3.3. Pose-aware Discriminator

As mentioned above, we remove the pose prior of a 3D-aware generator by parameterizing camera pose $\xi$ from a latent code $z$. The 3D representation leveraged in the generator is helpful in disentangling the camera factor. However, the conventional discriminator remains to differentiate real and fake images from 2D space, leading to inadequate supervision to factorize camera poses from latent codes. It easily makes the generator synthesize flat shapes and learn invalid camera distribution. Therefore, we propose to make the discriminator pose-aware.

**Learning Pose-aware Discriminator.** To let the discriminator become aware of poses, we assign a pose estimation task on fake images beyond bi-class domain classification, which is to derive the pose information from the given images. We introduce a pose branch $\Phi(\cdot)$ on top of the discriminator and optimize it with the following objective:

$$\hat{\xi} = \Phi(I_f),$$  

(4)

$$L_{\text{pose}} = l_2(\xi, \hat{\xi}),$$  

(5)

where $l_2(\cdot)$ denotes the function measuring $l_2$ distance between poses. $\xi$ is the pose that is inferred from the same latent code as $I_f$ by the generator.

**Performing Pose-aware Discrimination.** With the help of the pose branch, our discriminator can be adopted to infer camera poses for the given images. Even though the pose branch training is done on fake images, it is also generalizable on real images $I_r$ from the dataset. In such cases, we can leverage the pose-aware discriminator to perform discrimination on real or fake images as well as differentiate whether the image is attached with an accurate pose or not. Concretely, we first ask the discriminator to extract a pose from the input image, and then the pose is treated as a pseudo label to perform conditional bi-class classification. Here, we take the real image part of the discriminator loss as an example:

$$L_{D}^{I_r} = -E[\log(D(I_r|\Phi(I_r))].$$  

(6)

In this way, we facilitate the discriminator with awareness of pose cues, leading to better pose alignment across various synthesized samples. As a result, this alignment also
implicitly prevents the generator from degenerate solutions where a flat shape is usually generated with huge artifacts. It is worth noting that the discriminator learns poses from the generator and, in turn, uses the conditional discrimination score to update the pose branch in the generator. Therefore, the poses are learned in a loop-back manner without any annotations.

3.4. Training Objectives

Adversarial Loss. We use the standard adversarial loss for training following [8],

\[ L_D = -\mathbb{E}[\log (1 - D(I_f|\Phi(I_f)))] - \mathbb{E}[\log (D(I_r|\Phi(I_r)))] + \lambda \mathbb{E}[\|\nabla I_r \cdot D(I_r)\|^2_2], \]

\[ L_G = -\mathbb{E}[\log (D(I_f|\Phi(I_f)))], \]  \hspace{1cm} (7)

where \( I_r \) and \( I_f \) are real data and generated data, respectively. The third term in Eq. (7) is the gradient penalty, and \( \lambda \) denotes the weight for this term.

Symmetry Loss. Planar underlying shapes are easily generated when single-view images are synthesized for training. To avoid the trivial solution, we ask the network to synthesize the second image under another view, which we choose the symmetrical camera view \( \xi' \) regarding the \( yz \)-plane,

\[ I_f' = G(z, \xi'). \]  \hspace{1cm} (9)

The novel view image will be used to calculate the adversarial loss as in Eq. (7) and Eq. (8), and get \( L_{D'} \) and \( L_{G'} \).

Pose Loss. As stated in Sec. 3.3, we also attach an auxiliary pose branch \( \Phi(I_f) \) in the discriminator to perform pose estimation on the given images. We use Eq. (5) for pose branch training.

Full Objectives. In summary, the pose-free generator and the pose-aware discriminator are jointly optimized with

\[ L = L_G + L_{G'} + L_D + L_{D'} + \gamma L_{\text{pose}}, \]  \hspace{1cm} (10)

where \( \gamma \) is the weight of pose loss.

3.5. Implementation Details

We build our PoF3D on the architecture of EG3D [3]. For the pose-aware generator, we do not use pose-conditioned generation. Instead, we Instantiate the pose branch with two linear layers and a leaky ReLU activation in between. This branch takes in latent codes from \( w \) space and outputs camera poses. The camera pose consists of an azimuth angle and an elevation angle, which indicates the camera position for rendering. The triplane resolution is \( 256 \times 256 \), and the rendering in the neural radiance field is conducted on \( 64 \times 64 \) resolution. Both the feature map and the image are rendered. A super-resolution module then transforms the feature map into a high-resolution image. The pose-aware discriminator is inherited from the dual discriminator in EG3D [3]. We add the pose branch on the features before the last two fully-connected layers that output the realness score. The pose branch is composed of two fully-connected layers with a leaky ReLU as the activation function in the middle. The learning rate of the generator is \( 2.5e^{-3} \) while that of its pose branch is set to \( 2.5e^{-5} \). The discriminator’s learning rate is \( 2e^{-3} \). More details are available in the Supplementary Material.

4. Experiments

4.1. Experimental Settings

Datasets. We evaluate PoF3D on three datasets, including FFHQ [12], Cats [51], and Shapenet Cars [5]. FFHQ is a real-world dataset that contains unique 70K high-quality images of human faces. We follow [3] to align and crop the data. Cats includes 10K real-world cat images of various resolutions. The data is preprocessed following [6]. Shapenet Cars [5] is a synthetic dataset that contains different car models. We use the dataset rendered from [3] that has around 530K images. Unlike the face-forward datasets, the camera poses of it span the entire \( 360^\circ \) azimuth and \( 180^\circ \) elevation distributions. In the experiments, we use the resolution of \( 256 \times 256 \) for FFHQ and Cats, and \( 128 \times 128 \) for Shapenet Cars.

Baselines. We compare our approach against two methods: CAMPARI [20], the state-of-the-art pose learning method in 3D-aware image synthesis, and EG3D [3], the state-of-the-art in 3D-aware image synthesis. We also build a baseline method which is a combination of them, where we incorporate the pose learning method in CAMPARI into the framework of EG3D. More details regarding the baselines can be found in Supplementary Material.

Metrics. We use five metrics to evaluate the performance, including Fréchet Inception Distance (FID) [10], Depth Error [3], Pose Error [47], Reprojection Error (RE) [47], and Jensen–Shannon Divergence (JS). FID is measured between 50K generated images and all real images. Depth Error is used to assess the quality of geometry. We follow [3] and calculate the mean squared error (MSE) against pseudo-ground-truth depth estimated by [7] on 10K synthesized samples. We evaluate the pose accuracy on 10K generated samples as well. Given the synthesized images, we leverage the head pose estimator [53] and report the L1 distance between the estimated poses and the poses inferred from the generator. Note that we subtract the respective means of estimated and inferred pose distribution to compensate the canonical shift problem (see Fig. 4). Inter-view consistency is measured by the re-projection error following [47]. We render 5 views by sampling azimuth uniformly in the range of \([-23^\circ, 23^\circ]\) and warp two consecutive views to each other.
Figure 3. **Qualitative comparison between our approach and baselines.** CAMPARI [20] struggles to generate reasonable results. “CAMPARI + EG3D” suffers from mode collapse on the FFHQ and Cats datasets. EG3D [3] achieves impressive rendering and reconstruction results when camera poses of training images are given. In contrast, our approach can generate high-quality rendering and geometry without any pose priors.

to report the MSE. The images are normalized in the range of [-1, 1] for evaluation. Since the azimuth range learned by CAMPARI is much smaller, we use the learned range for sampling in the measurement of CAMPARI. To measure the quality of the learned pose distribution, we report Jensen–Shannon Divergence on 50K synthesized samples and all real images by averaging divergence values over azimuth and elevation.

### 4.2. Main Results

**Qualitative Comparison.** Fig. 3 shows the qualitative comparison against the baselines. CAMPARI learns faulty and very sharp shapes that span the entire space along the ray direction (the same phenomenon as observed in Fig. 1d). Therefore, even if the object is rotated with an extremely small angle (e.g., 2.5°), the caused visual effect is equivalent to that of the normal rotation of a decent shape with a much larger angle (e.g., 23°). We visualize CAMPARI in its own valid range of the horizontal angle (around [-2.5°, 2.5°] for FFHQ, [-5°, 5°] for Cats, and [-30°, 30°] for Shapenet Cars). For others, visualizations on FFHQ and Cats are conducted on the angle range [-23°, 23°]. Poses for cars are randomly sampled from the entire 360° azimuth and 180° elevation distributions. EG3D leverages ground-
truth camera poses for training and generates high-quality images and underlying geometry simultaneously. When incorporated with the pose learning method in CAMPARI, EG3D suffers from pose collapse and converges to one pose on FFHQ and Cats. Hence, novel views are not available in such cases, and the 3D-aware image synthesis problem is degraded to 2D image synthesis. On Shapenet Cars, the generated image and its corresponding shape are visually reasonable under a specific view but become completely unnatural when deviating from that view. Our method, without any pose priors, can generate high-fidelity images and shapes on par with EG3D.

Quantitative Comparison. We report the quantitative evaluation of baselines and our method in Tab. 1. CAMPARI fails to capture the underlying pose distribution and gets high Jensen–Shannon Divergence scores and pose errors. Consequently, the quality of images and shapes is unsatisfying as well, reflected by high FID scores and depth errors. CAMPARI+EG3D generates planar shapes and collapses to one pose or a small range of poses, which simplifies 3D-aware image synthesis to 2D image synthesis, resulting in lower FID scores but higher depth error, pose error and Jensen–Shannon Divergence score. EG3D, the method that uses pose annotations, exhibits good performance in terms of image quality and geometry quality. Our method yields results on par with EG3D and learns much better underlying pose distribution than the baselines. The quality of the geometry affects the multi-view consistency of the learned neural fields. Thus, EG3D and ours, which synthesize better shapes, can get better scores of reprojection errors. Note that our pose error is higher than that of EG3D, one of the reasons is that there exists the canonical view shift. As we do not provide any information on what the canonical view will be like, the model tends to learn a canonical view that leads to the best of its performance, which could be different from the standard. More discussions are in Sec. 4.4.

4.3. Ablation Study

We conduct ablation studies on FFHQ 256 × 256 to analyze the effectiveness of PoF3D. Evaluation results are shown in Tab. 2.
alignment.

**Learning Rate of Pose Learner.** Learning rate of the pose branch in the generator matters a lot. With a large learning rate (e.g., $2.5e^{-4}$ in Tab. 2), the generator always tries with new poses of large variation at the beginning and leaves the content learning behind, resulting in unstable training and higher tendency of mode collapse. As reported in Tab. 2, all the evaluation metrics experience a significant drop. On the contrary, if the learning rate is too small (e.g., $2.5e^{-6}$ in Tab. 2), the generator cannot explore the pose distribution enough and easily gets stuck in a small range of pose distribution, putting more focus on the content learning. Consequently, the model fails to capture and understand the pose distribution, leading to worse shapes. A proper learning rate should balance the exploration of pose distribution and the learning of the content.

### 4.4. Pose Analysis

In this section, we analyze the pose distributions that are learned by our pose-free generator and pose-aware discriminator.

**Poses in Pose-free Generator.** To visualize the pose
distribution learned in the pose-free generator, we randomly sample 50K latent codes and infer the poses from them. The pose distributions from the dataset and learned in the generator are shown in Fig. 4. Without any pose priors, our method can capture the pose range and distribution of the dataset in general. As observed in the figure for elevation distributions on the FFHQ dataset, there exists a distribution shift. The reason is that we do not release any information about the canonical view to the model, and therefore, the model can interpret the canonical view freely with different angles which might be different from the one defined in the dataset. Generally, the model will choose the one that leads to the best performance as its canonical view.

**Poses in Pose-aware Discriminator.** Since our discriminator is equipped with a pose predictor, we hire it to predict the poses for all data in the dataset and draw the distributions accordingly in Fig. 5. For FFHQ and Cats datasets, the distributions are well approximated, and the canonical view shift problem also exists in the discriminator. On Shapenet Cars, the distribution of azimuth is biased to one side. We conjecture the reason is that the model has difficulty distinguishing the front and rear of the car as they look very similar. As a result, the learned distribution shifts towards a hemisphere instead of the entire sphere.

**4.5. Applications**

**Linearity in Latent Space.** To demonstrate the latent space learned by PoF3D is semantically meaningful, we randomly sample two latent codes and linearly interpolate between them. The interpolation results are shown in Fig. 6. We can see that both the appearance and the underlying geometry are changing smoothly. Besides, as we infer the pose from the latent code, poses undergo smooth changes as well.

**Real Image Inversion.** One of the potential applications of PoF3D is to reconstruct the geometry from a real image and enable novel view synthesis. To achieve this goal, PTI [28], a GAN inversion method, is adopted to perform on our pose-free generator. In the previous methods [3] that demonstrate GAN inversion as an application, an off-the-shelf pose predictor is required to obtain the pose of the given image prior to performing GAN inversion. Ours, however, benefits from the changed formulation in which the camera pose is encoded in the latent space. Therefore, we can get not only the inverted neural radiance field but also the camera pose under which the given image is captured, when performing GAN inversion. The result is shown in Fig. 7. Our method can successfully invert the image and synthesize vivid images under novel views.

**5. Conclusion**

This work presents PoF3D, which learns 3D-aware image synthesis without using any pose priors. A pose-free generator is designed to infer camera poses directly from the latent space so that the requirement for pose priors in the previous works is removed. Besides, to help the generator better capture the underlying distribution, we make the discriminator pose-aware by asking it to first predict the camera pose and then use it as a condition when performing conditional real/fake discrimination. With such a design, experimental results demonstrate that our approach is able to synthesize high-quality images and high-fidelity shapes that are on par with state-of-the-art 3D synthesis methods without the need for manual pose tuning or dataset labeling.

**Acknowledgement.** This research has been made possible by funding support from the Research Grants Council of Hong Kong under the Research Impact Fund project R6003-21.
References


[38] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for styleGAN image manipulation. *ACM Trans. Graph.*, 2021. 2


