pi-GAN: Periodic Implicit Generative Adversarial Networks for 3D-Aware Image Synthesis

Eric R. Chan* Marco Monteiro* Petr Kellnhofer Jiajun Wu Gordon Wetzstein
Stanford University
{erchan,pkellnho,jiajunw,gordon.wetzstein}@stanford.edu, monteiro.marcoa@gmail.com

Abstract

We have witnessed rapid progress on 3D-aware image synthesis, leveraging recent advances in generative visual models and neural rendering. Existing approaches however fall short in two ways: first, they may lack an underlying 3D representation or rely on view-inconsistent rendering, hence synthesizing images that are not multi-view consistent; second, they often depend upon representation network architectures that are not expressive enough, and their results thus lack in image quality. We propose a novel generative model, named Periodic Implicit Generative Adversarial Networks (π-GAN or pi-GAN), for high-quality 3D-aware image synthesis. π-GAN leverages neural representations with periodic activation functions and volumetric rendering to represent scenes as view-consistent radiance fields. The proposed approach obtains state-of-the-art results for 3D-aware image synthesis with multiple real and synthetic datasets.

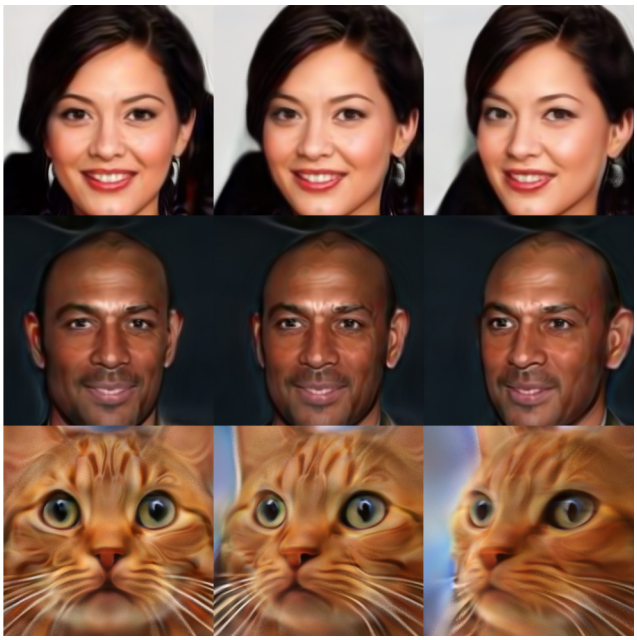

Figure 1: Selected examples synthesized by π-GAN with CelebA [35] and Cats [70] datasets.

1. Introduction

Generative Adversarial Networks (GANs) are capable of generating high-resolution, photorealistic images [25, 26, 27]. However, these GANs are often confined to two dimensions because of a lack of photorealistic 3D training data; therefore, they cannot support tasks such as synthesizing multiple views of a single object. 3D-aware image synthesis offers to learn neural scene representations unsupervised from 2D images. The learned representations can be used to render view-consistent images from new camera poses [44, 57, 19].

Current solutions have achieved impressive results in decoupling identity from structure, allowing for the rendering of a single instance from multiple poses. Nevertheless, these approaches either lack multi-view consistency or fine detail. Voxel-based approaches [19] generate interpretable, true 3D representations, but are limited by computational complexity to low resolutions and coarse detail. Convolutional approaches with deep-voxel representations [44, 45] take advantage of recent progress in convolutional GANs and can create finely detailed images. However, because of their reliance on learned black-box rendering, these approaches fail to guarantee multi-view consistency and cannot easily generalize beyond the training distribution of camera poses at inference. Recent approaches that leverage neural implicit representations [57] incorporate representations based on neural network–parameterized radiance fields that ensure multi-view consistency and explicit camera control. Nonetheless, the implicit representations used by these approaches have so far been unable to effectively express fine details, leading to compromised image quality.

We propose Periodic Implicit Generative Adversar-

*These authors contributed equally to this work. Project page: https://marcoamonteiro.github.io/pi-GAN-website/

ial Networks (π-GAN), a generative adversarial approach to unsupervised 3D representation learning from images. Given input noise, π-GAN conditions an implicit radiance field represented by a SIREN network [59], a fully-connected network with periodic activation functions. The conditioned radiance field maps a 3D location and 2D viewing direction to a view-dependent radiance and view-independent volume density [23, 38]. Using a differentiable volume rendering approach that relies on classical volume rendering techniques, we can render the radiance field from arbitrary camera poses [42].

π-GAN improves upon the image quality and view-consistency of previous approaches to 3D-aware image synthesis, as shown in Figure 1. The proposed method utilizes a SIREN-based neural radiance field representation to encourage multi-view consistency, allowing rendering from a wide range of camera poses and providing an interpretable 3D structure. The SIREN implicit scene representation, which makes use of periodic activation functions, is more capable than ReLU implicit representations at representing fine details and enables π-GAN to render sharper images than previous works.

Beyond introducing π-GAN, we make two additional technical contributions. First, we observe that while existing work has conditioned ReLU-based radiance fields through concatenation of the input noise to one or more layers, conditioning-by-concatenation is sub-optimal for implicit neural representations with period activations (SIRENs). We instead propose to use a mapping network to condition layers in the SIREN through feature-wise linear modulation (FiLM) [51, 9]. This contribution can more generally be applied to SIREN architectures beyond GANs. Second, we introduce a progressive growing strategy, inspired by previous successes in 2D convolutional GANs [25], to accelerate training and offset the increased computational complexity of 3D GANs.

We obtain state-of-the-art 3D-aware image synthesis results on real-world and synthetic datasets, demonstrate that our method generalizes to new viewpoints, and has applications to novel view synthesis. Moreover, the 5D spatio-angular radiance field representation used by π-GAN allows for an interpretable 3D proxy shape to be extracted via the marching cubes algorithm [36]. While these proxy shapes may not be as high quality as those estimated by single-view shape reconstruction methods tailored to this task [68], they often end up resulting in a fair approximation, all without explicit supervision.

Our contributions in this paper include the following:

- We introduce SIREN-based implicit GANs as a viable alternative to convolution GAN architectures.
- We propose a mapping network with FiLM conditioning and a progressive growing discriminator as key components to achieve high quality results with our novel SIREN-based implicit GAN.
- We demonstrate view consistency and explicit camera control as advantages of approaches that rely on an underlying neural radiance field representation and classical rendering.
- We achieve state-of-the-art results on 3D-aware image synthesis from unsupervised 2D data on the CelebA [35], Cats [70], and CARLA [8, 57] datasets.

2. Related Work

Neural representations and rendering. Emerging neural implicit scene representations promise 3D-structure-aware, continuous, memory-efficient representations for parts [13, 12], objects [49, 41, 1, 16, 69, 7, 5], or scenes [10, 60, 21, 50, 59]. These can be supervised with 3D data, such as point clouds, and optimized as either signed distance functions [49, 41, 1, 16, 60, 21, 50, 58, 28] or occupancy networks [40, 6]. Using neural rendering [63], implicit neural representations can also be trained using multiview 2D images [54, 60, 48, 47, 42, 69, 32, 22, 34]. Temporally aware extensions [46] and multimodal variants with part-level semantic segmentation [29] have also been proposed.

Among these approaches, sinusoidal representation networks (SIREN) [59] and neural radiance fields (NeRF) [42] are most closely related to our work. Specifically, we use SIREN as the representation network architecture of our framework combined with a neural rendering technique inspired by NeRF. Both SIREN and NeRF, however, have only been explored in the context of overfitting to individual objects or scenes, whereas we study the combination of aspects of these seminal works for applications in 3D GANs. Exploring the unique challenges of training a neural implicit GAN supervised by natural 2D data is one of the core contributions of our work.

Generative 3D-aware image synthesis. Generative Adversarial Nets (GANs) [15], or more generally the paradigm of adversarial learning, have led to significant progress in various image synthesis tasks, including image generation [52, 25, 26, 27], image-to-image translation [71], interactive image editing [66], and learning from partial and noisy observations [3]. These methods operate on the 2D space of pixels, ignoring the 3D nature of our physical world. This has limited the application of these generative models in tasks such as view synthesis.

Visual Object Networks [72] and PrGANs [11] learn to synthesize 2D images by first generating a voxelized 3D shape using a 3D-GAN [67] and then projecting it into 2D. HoloGAN [44] and BlockGAN [45] have extended the system by incorporating a volumetric but implicit 3D representation. While these methods attempt to model the 3D structure of the object in the synthesized image, the use of

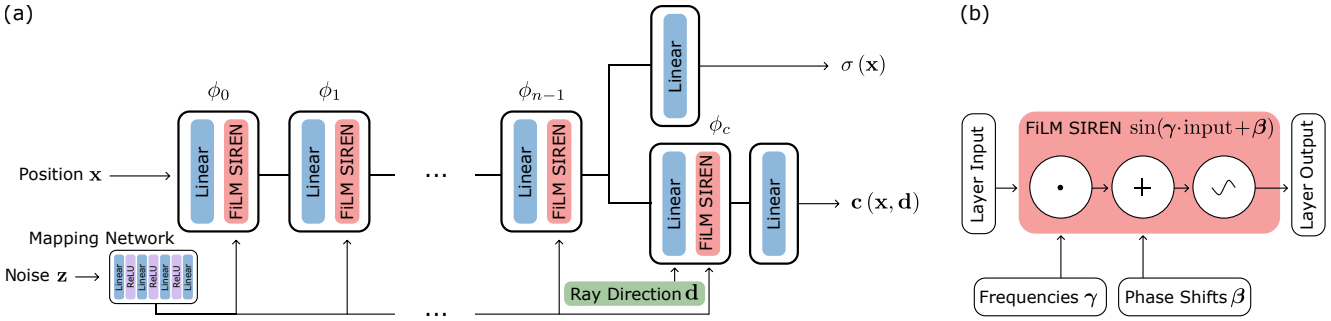

Figure 2: The π-GAN generator architecture.

an explicit volume representation has constrained their resolution [37]. Szabó *et al*. [61] and Liao *et al*. [31] instead proposed to model 3D shapes as meshes and collections of primitives for image synthesis, respectively. However, these representations lack the expressiveness needed to synthesize high-fidelity pictures.

The work most similar to ours is GRAF [57], which learns a generative model for implicit radiance fields for 3D-aware image synthesis. Although π-GAN operates in a similar setting, its network architecture and training strategy differ from GRAF in several ways. First, we use SIREN rather than a positionally encoded ReLU MLP as a choice of neural implicit representation. Second, GRAF conditioned its MLP generator on both a shape noise code and an appearance noise code by concatenation; in contrast, we leverage a StyleGAN-inspired mapping network, which conditions the entire MLP on a single input noise vector through FiLM conditioning. Third, we utilize a progressive growing strategy during training. Finally, we did not employ a patch-based discriminator, as used by GRAF, as SIREN is prone to local overfitting to the last batch if sufficient coverage of the space is not maintained. Our experiments demonstrate that all of our innovations are critical to high-quality image synthesis results.

Beyond unconditional 3D-aware image generation, there is an orthogonal line of work on conditional reconstruction of 3D shape and texture from partial observations. These reconstructions can later be used for novel view synthesis. Various 3D representations have been considered for the task, including voxels [19, 65], meshes [24, 6, 14, 18, 43], point clouds [62], a depth map [68], and implicit functions [53, 64]. Some of these methods are also grounded in adversarial training. While these methods focus on 3D reconstruction, π-GAN aims to learn an unconditional generative model of radiance fields.

3. Methods

π-GAN is a generative approach to learning radiance field representations from unlabeled 2D images, with the goal of synthesizing high-quality view consistent images. Traditional 2D GANs, such as StyleGAN [26], take in a latent vector $\mathbf{z} \sim p_{\mathbf{z}}$ and directly produce a 2D image. Instead of directly generating a 2D image from the input noise, $\mathbf{z}$, our generator $G_{\theta_G}(\mathbf{z}, \xi)$ produces an implicit radiance field conditioned on $\mathbf{z}$. This radiance field is rendered using volume rendering to produce a 2D image from some camera pose ξ.

At training time, the generated images are directed to a traditional convolutional discriminator for adversarial training. At test time, the radiance field can be rendered from arbitrary camera poses to produce view-consistent images.

3.1. SIREN-Based Implicit Radiance Field

We represent 3D objects implicitly with a neural radiance field, which is parameterized as a multilayer perceptron (MLP) that takes as input a 3D coordinate in space $\mathbf{x} = (x, y, z)$ and the viewing direction $\mathbf{d}$. The neural radiance field outputs both the spatially varying density $\sigma(\mathbf{x}) : \mathbb{R}^3 \rightarrow \mathbb{R}$ and the view-dependent color $(r, g, b) = \mathbf{c}(\mathbf{x}, \mathbf{d}) : \mathbb{R}^5 \rightarrow \mathbb{R}^3$. Moreover, we leverage a StyleGAN-inspired mapping network to condition the SIREN on a noise vector $\mathbf{z}$ through FiLM conditioning [51, 9].

As shown in Figure 2a, we formalize the FiLM-ed SIREN backbone of our representation as

$$\Phi(\mathbf{x}) = \phi_{n-1} \circ \phi_{n-2} \circ \ldots \circ \phi_0(\mathbf{x}), \quad (1)$$

$$\phi_i(\mathbf{x}_i) = \sin(\boldsymbol{\gamma}_i \cdot (\mathbf{W}_i \mathbf{x}_i + \mathbf{b}_i) + \boldsymbol{\beta}_i), \quad (2)$$

where $\phi_i : \mathbb{R}^{M_i} \mapsto \mathbb{R}^{N_i}$ is the i^{th} layer of an MLP. It consists of an affine transform defined by the weight matrix $\mathbf{W}_i \in \mathbb{R}^{N_i \times M_i}$ and the biases $\mathbf{b}_i \in \mathbb{R}^{N_i}$ applied on the input $\mathbf{x}_i \in \mathbb{R}^{M_i}$, followed by the sine nonlinearity applied to each component of the resulting vector (Figure 2b). Our mapping network is a simple ReLU MLP, which takes as input a noise vector $\mathbf{z}$ and outputs the frequencies $\boldsymbol{\gamma}_i$ and phase shifts $\boldsymbol{\beta}_i$, which condition each layer of the SIREN.

We found this mapping network to be more expressive than concatenation-based conditioning. It yielded image-quality improvements, both for conditioning ReLU-based

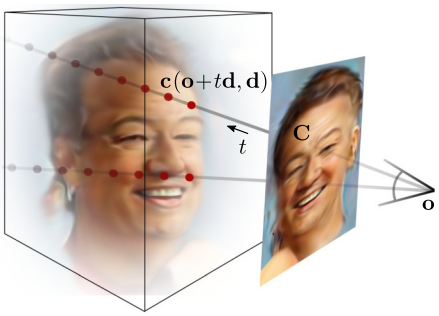

Figure 3: A visualization of our neural volume rendering procedure. Given a conditioned radiance field, we cast rays from the camera origin $\mathbf{o}$, sample density σ and color $\mathbf{c}$ values along each ray, and calculate pixel color $\mathbf{C}$ using Eq. 5.

and SIREN-based neural implicit representations. The ablation studies shown in Sec. 4.3 give further insight into these conditioning methods.

Both density and color of our implicit volume are then defined as

$$\sigma\left(\mathbf{x}\right) = \mathbf{W}_{\sigma}\Phi\left(\mathbf{x}\right) + \mathbf{b}_{\sigma}, \tag{3}$$

$$\mathbf{c}\left(\mathbf{x}, \mathbf{d}\right) = \mathbf{W}_{c}\phi_{c}\left(\left[\Phi\left(\mathbf{x}\right), \mathbf{d}\right]^{T}\right) + \mathbf{b}_{c}, \tag{4}$$

where $\mathbf{W}_{\sigma/c}$ and $\mathbf{b}_{\sigma/c}$ are additional weight and bias parameters.

3.2. Neural Rendering

We render a neural radiance field from arbitrary camera poses ξ using neural volume rendering. For this purpose, we employ a pinhole camera model and cast rays from the camera origin $\mathbf{o}$ to compute the integrals along each ray through the volume. At every sample, our generator predicts the volume density σ and color $\mathbf{c}$. The pixel color $\mathbf{C}$ for a camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$ with near and far bounds t_n and t_f is then calculated using the volume rendering equation [38]:

$$\begin{aligned} \mathbf{C}(\mathbf{r}) &= \int_{t_n}^{t_f} T(t)\sigma\left(\mathbf{r}(t)\right)\mathbf{c}\left(\mathbf{r}(t), \mathbf{d}\right)dt, \\ \text{where} \quad T(t) &= \exp\left(-\int_{t_n}^{t}\sigma(\mathbf{r}(s))ds\right). \end{aligned} \tag{5}$$

Our approach implements a discretized form of this equation using the stratified and hierarchical sampling approach introduced by NeRF [42] (see Figure 3).

This neural rendering approach, which is also adopted by GRAF [57], has several advantages over previous 3D-to-2D projections. Neural rendering allows for explicit control over camera pose, focal length, aspect ratio, and other parameters, while simple projections, such as those used by HoloGAN [44], are restricted to representing poses in the training dataset.

3.3. Discriminator

Following ProgressiveGAN [25], we use a convolutional discriminator D_{θ_D} with parameters θ_D that grows progressively. We begin training at low resolutions and high batch sizes, during which the generator can focus on producing coarse shapes. As training progresses, we increase the image resolution and add new layers to the discriminator to handle the higher resolutions and discriminate fine details. For most experiments, we begin training at 32×32 and double the resolution twice during training, up to 128×128. In practice, we found this progressive growing strategy to allow for larger batch sizes at the beginning of training, which helped to stabilize and speed training (see Sec. 4.3). Final results are rendered by sampling 512×512 pixels.

Unlike ProgressiveGAN [25], our generator architecture does not grow; instead, we increase the resolution of the generator by sampling rays more densely from the same implicit representation.

3.4. Training Details

At training time, we randomly sample camera poses ξ from a distribution p_{ξ}. The pose distributions for each dataset are known a priori and approximated as either Gaussian, for CelebA and Cats, or uniform, for CARLA (see supplement for details). In our experiments, we constrained camera positions to the surface of a unit sphere and directed the camera to point towards the origin. At training time, pitch and yaw along the sphere were sampled from a distribution that was tuned according to the dataset. Real images I are sampled from the training set with distribution p_I. We use the non-saturating GAN loss with R1 regularization [39]:

$$\begin{aligned} \mathcal{L}(\theta, \phi) &= \mathbf{E}_{\mathbf{z}\sim p_z, \xi\sim p_{\xi}}[f(D_{\theta_D}(G_{\theta_G}(\mathbf{z}, \xi)))] \\ &\quad + \mathbf{E}_{I\sim p_{\mathcal{D}}}[f(-D_{\theta_D}(I)) + \lambda|\nabla D_{\theta_D}(I)|^2], \\ &\text{where} \quad f(u) = -\log(1 + \exp(-u)). \end{aligned} \tag{6}$$

We train π-GAN in a generative adversarial framework in which a generator and discriminator compete in a zero sum game. Our generator tries to minimize Equation 6, while the discriminator simultaneously tries to maximize Equation 6. We use the Adam optimizer with $\beta_1 = 0$, $\beta_2 = 0.9$. We initialize learning rates to 5×10^{-5} for the generator and 4×10^{-4} for the discriminator, decayed over training to 1×10^{-5} and 1×10^{-4} respectively. Further training and implementation details can be found in the supplemental materials.

4. Experiments and Analysis

In this section, we first evaluate the quality of images generated by π-GAN. We then demonstrate that it learns 3D representations that enables synthesizing images at unseen

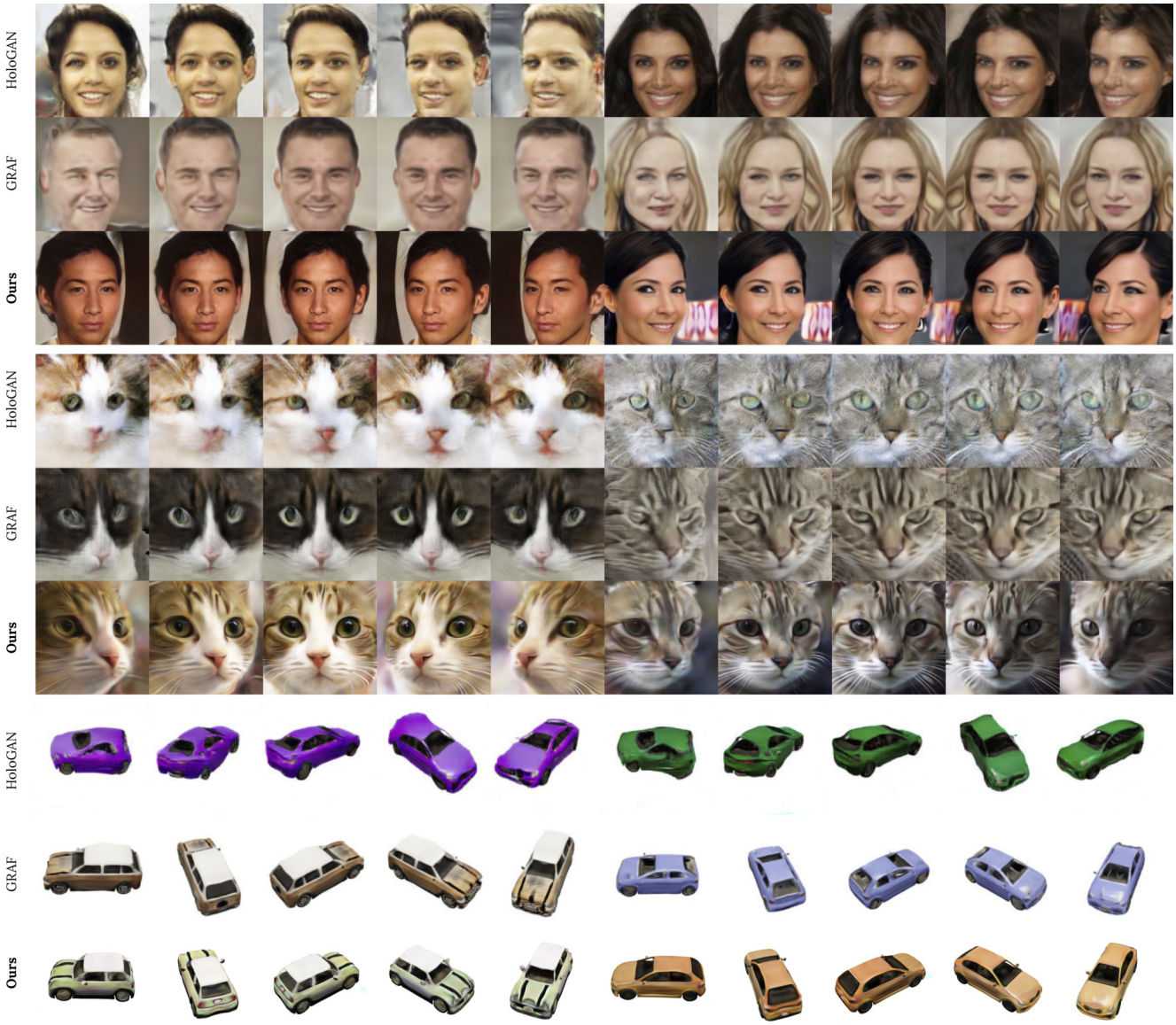

Figure 4: Qualitative comparison on CelebA, Cats, and CARLA.

poses. We also include ablation studies to justify our use of sinusoidal activations and mapping network conditioning.

4.1. Evaluating Image Quality

Datasets. We evaluate π-GAN on the real-world CelebA [35] and Cats [70] datasets, as well as the synthetic CARLA [8, 57] dataset. CelebA contains 200,000 high-resolution face images of 10,000 different celebrities. We crop the images from the top of the hair to the bottom of the chin. The Cats dataset contains 6,444 128×128 images of cat heads. The CARLA dataset contains 10k images of 16 car models with random texture and color properties, rendered with the Carla Driving simulator. We train and evaluate at 128×128 resolution for all datasets and models. We evaluate all models using a moving average of parameters.

Baselines. We compare against two previous approaches to 3D-aware image synthesis: HoloGAN [44] and Generative Radiance Fields (GRAF) [57]. Baseline models were obtained as pre-trained checkpoints directly from the authors or trained until convergence using the recommended hyperparameters.

Qualitative results. Figure 4 compares images generated by π-GAN, HoloGAN, and GRAF on three datasets.

Qualitatively, HoloGAN achieves good image quality but suffers from multi-view inconsistency. Although it gen-

Figure 5: Uncurated generated faces, corresponding to the first 30 random seeds.

(a) CelebA @ 128×128

	FID ↓	KID ↓	IS ↑
HoloGAN	39.7	2.91	1.89
GRAF	41.1	2.29	2.34
π-GAN	**14.7**	**0.39**	**2.62**

(b) Cats @ 128×128

	FID ↓	KID ↓	IS ↑
HoloGAN	40.4	3.30	2.03
GRAF	28.9	1.43	1.66
π-GAN	**16.8**	**0.92**	**2.06**

(c) CARLA @ 128×128

	FID ↓	KID ↓	IS ↑
HoloGAN	67.5	3.95	3.52
GRAF	41.7	2.43	3.70
π-GAN	**29.2**	**1.36**	**4.27**

Table 1: FID, KID mean$\times$100, and IS for CelebA, Cats, and CARLA datasets.

erally produces sharp images, identity shift is visible across rotations, particularly at the edges of the training distribution. HoloGAN struggled on the synthetic CARLA dataset, which featured much larger variations in viewpoint than CelebA or Cats. Previous papers were also unable to obtain consistent HoloGAN baselines on this dataset [57].

GRAF, which allows for explicit camera control, is more capable than HoloGAN at recovering wide viewing angles. Because it utilizes a 3D representation, it renders different views of the same scene with less identity shift than HoloGAN. However, GRAF is less capable than HoloGAN at rendering fine details such as hair and teeth, and generally produces images that are more cartoon-ish and less lifelike than HoloGAN.

Our π-GAN combines fine details with the ability to represent a wide range of camera angles. Compared with HoloGAN and GRAF, it better recreates details such as individual teeth (CelebA) and whiskers (Cats). Because we represent each instance with a radiance field, π-GAN generates images that are inherently view consistent, have minimal identity shift, and that recover a wide range of angles.

Quantitative results. We evaluate image quality using Frechet Inception Distance (FID) [20], Kernel Inception Distance (KID) [2], and Inception Score [56]. Tables 1a, 1b, and 1c show a quantitative comparison on CelebA, Cats, and CARLA, respectively. We show significant improvements in image quality metrics compared with baselines, particularly on real-world datasets with fine details. Additional results, including precision-recall plots [55], are provided in the supplemental material.

Our evaluation was consistently performed across all models for Table 1. Note that specific experiment parameters, such as image crop, may differ from those used by other authors.

4.2. Generating Approximate 3D Representations

A key advantage of our approach over previous CNN attempts at 3D representation learning is that by generating an implicit radiance field, our model learns an underlying 3D-structure-aware representation. This representation allows for explicit camera control, naturally lends itself to rendering poses that were uncommon or unseen at training time, and is interpretable.

Extrapolation to rare or unseen camera poses. π-GAN relies on an underlying 3D structural representation and offers explicit camera control. Like previous methods that offer explicit camera control (e.g., [57]), it more readily renders views and poses outside of the training dataset distribution than previous methods that rely on black-box repre-

Figure 6: π-GAN is capable of rendering views from steep angles, producing reasonable results even beyond two standard deviations of camera yaw on CelebA. Face yaw on CelebA is approximately zero-centered Gaussian, with a standard deviation of 17^o from the centerline.

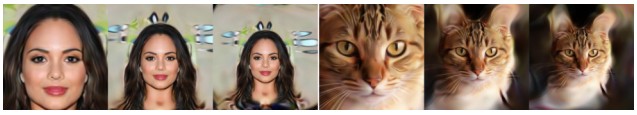

Figure 7: Explicit camera control at inference enables rendering views completely absent from the training distribution of camera poses. Although π-GAN was trained only on close-up images, it extrapolates to zoomed-out poses.

Figure 8: Linearly interpolating between two latent codes.

sentations or projections (e.g., [44]).

Figure 6 shows that the explicit camera control and representation naturally generalizes to rendering views even from steep angles, although visual artifacts are stronger at the edges of the camera distribution. This is a consequence of the distribution of CelebA images being imbalanced towards front-facing images. As shown in Figure 4, CARLA, which features uniformly distributed poses, did not suffer from this issue.

Figure 7 illustrates that, despite only training on tightly cropped images, the radiance field extrapolates when we zoom out the camera. Because the radiance field may be rendered from any of a wide variety of angles at training time, the generator is encouraged to produce a radiance field that represents the entire scene, even if only a small portion will be visible in any single image.

To demonstrate that the latent space learned by π-GAN is semantically meaningful, we show the results of interpolating between two latent codes in Figure 8.

Interpreting the 3D representation. Although the color output of the implicit representation depends on ray direction to allow for view-dependent effects, such as specularities, the density output σ is completely view independent, resulting in a view-consistent 3D structure that represents a proxy shape of the scene. This 3D structure can be extracted and visualized using the marching cubes algorithm [36] on the density output of the conditioned radiance field to produce a surface mesh. Figure 9 shows 3D models extracted from the 3D representation.

Figure 9: We can extract a proxy 3D representation as a mesh, either by projecting a depth-map (CelebA, Cats), or through marching cubes (CARLA).

Conditioning	Architecture	
	ReLU P.E.	Sine
Concatenation	32.0	21.6
Mapping Network	26.8	**5.15**

Table 2: FID scores on CelebA @ 64×64, when comparing network architectures with different activation functions and conditioning methods.

4.3. Ablations

We ablate sinusoidal activations and mapping network conditioning to better understand their individual contributions. We compare radiance fields with sinusoidal activations against radiance fields with ReLU activations and positional encodings (P.E.) [42]. Moreover, we evaluate radiance fields conditioned with a mapping network and FiLM conditioning against radiance fields conditioned via concatenation [57]. Table 2 summarizes the results of these experiments. Ablations were conducted at 64×64 in order to save computational resources. Sinusoidal activations and mapping network conditioning each yielded improvements against their respective baselines. However, the combined model, with both sinusoidal activations and a mapping network, was more effective than the sum of its parts.

Figure 10 compares early training steps for a model trained with progressive growing against a model initialized to the full 128×128 image resolution. Because computational complexity grows quadratically with image size, pro-

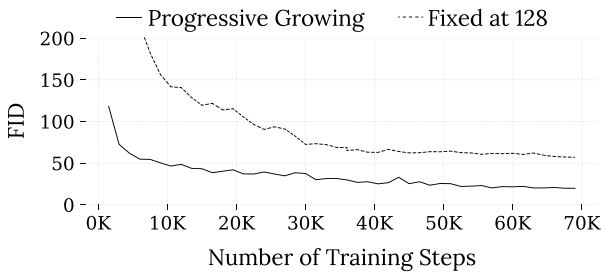

Figure 10: Ablation study for training π-GAN with and without progressive growing on CelebA @ 128×128

Figure 11: Using a trained π-GAN generator, we can optimize a radiance field to fit an input image and synthesize novel views from arbitrary camera poses.

gressive growing, which begins at low resolutions, allows for the use of much larger batch sizes at the start of training. The large batch sizes are helpful in stabilizing training, while also allowing for a higher throughput in images per iteration. As others have found before us [25], progressive growing, and the larger batch sizes it enables, helped ensure quality and diversity for generated images.

5. Discussion

Applications to novel view synthesis. Figure 11 demonstrates that it is possible to use a trained generator, without modifications, to perform single-view reconstruction using the procedure described by Karras et al. [27]. For this purpose, we freeze the parameters of our implicit representation and seek the frequencies γ_i and phase shifts β_i for each MLP layer i which produce a radiance field that, when rendered, best matches the target image. Additional details are found in the supplement.

Failure modes, limitations, and future work. While π-GAN has demonstrated considerable improvements to image quality for 3D-aware image synthesis, there remain a plethora of avenues for future work.

Figure 12: In a failure case reminiscent of the hollow-face illusion, our model sometimes generates objects with inverted sections.

Although the unsupervised learning of 3D shapes was not the focus of this work, π-GAN nevertheless produces interpretable and view-consistent 3D representations that capture the 3D structures of objects. Future work could focus on refining the quality of extracted meshes, with π-GAN as a viable solution to learning shapes from unposed images.

In certain cases, π-GAN can generate a radiance field that creates viable images when rendered from each direction but nonetheless fails to conform to the 3D shape that we would expect. As Figure 12 demonstrates, a concave face is a valid geometric solution, given the constrained range of poses the discriminator sees at training. Further investigation may reveal insights that could resolve such ambiguities.

While π-GAN has made strides in improving image quality for 3D-aware image synthesis, much work remains before implicit GANs can match the image quality of state-of-the-art 2D-convolutional GANs [27, 4, 25]. Future work may produce solutions to remaining visual artifacts and further improve image quality. π-GAN is computationally expensive compared to traditional 2D GANs because the complexity of training the generator scales not only with image size but also with depth along each ray. More efficient render techniques could lower the computational barrier and allow for larger, sharper images.

Ethical considerations. While our inverse rendering results only reconstruct static images, the method could be extended to generate fake photos or videos of real people (DeepFakes). DeepFakes pose a societal threat, and we do not condone using our work to generate fake images or videos of any person with the intent of spreading misinformation or tarnishing their reputation. We also recognize a lack of diversity in our faces results, stemming from the implicit bias in the CelebA dataset.

Conclusion. Photorealistic 3D-aware image synthesis has many exciting applications in vision and graphics. With our work, we take a significant step towards this goal.

Acknowledgements. Thanks to Matthew Chan for fruitful discussions and to Stanford HAI for AWS Cloud Credits. J.W. was supported by the Samsung Global Research Award and Autodesk. G.W. was supported by an NSF CAREER Award (IIS 1553333) and a PECASE from the ARO.

References

[1] Matan Atzmon and Yaron Lipman. SAL: Sign agnostic learning of shapes from raw data. In *Proc. CVPR*, 2020. 2

[2] Mikołaj Bińkowski, Dougal J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying mmd gans. In *Proc. ICLR*, 2018. 6

[3] Ashish Bora, Eric Price, and Alexandros G. Dimakis. AmbientGAN: Generative models from lossy measurements. In *Proc. ICLR*, 2018. 2

[4] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. In *Proc. ICLR*, 2019. 8

[5] Rohan Chabra, Jan Eric Lenssen, Eddy Ilg, Tanner Schmidt, Julian Straub, Steven Lovegrove, and Richard Newcombe. Deep local shapes: Learning local sdf priors for detailed 3d reconstruction. In *Proc. ECCV*, 2020. 2

[6] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Proc. CVPR*, 2019. 2, 3

[7] Thomas Davies, Derek Nowrouzezahrai, and Alec Jacobson. Overfit neural networks as a compact shape representation. *arXiv preprint arXiv:2009.09808*, 2020. 2

[8] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. Carla: An open urban driving simulator. In *Proc. CoRL*, 2017. 2, 5, 16

[9] Vincent Dumoulin, Ethan Perez, Nathan Schucher, Florian Strub, Harm de Vries, Aaron Courville, and Yoshua Bengio. Feature-wise transformations. *Distill*, 3(7):e11, 2018. 2, 3

[10] SM Ali Eslami, Danilo Jimenez Rezende, Frederic Besse, Fabio Viola, Ari S Morcos, Marta Garnelo, Avraham Ruderman, Andrei A Rusu, Ivo Danihelka, Karol Gregor, et al. Neural scene representation and rendering. *Science*, 360(6394):1204–1210, 2018. 2

[11] Matheus Gadelha, Subhransu Maji, and Rui Wang. 3d shape induction from 2d views of multiple objects. In *Proc. 3DV*, 2017. 2

[12] Kyle Genova, Forrester Cole, Avneesh Sud, Aaron Sarna, and Thomas Funkhouser. Local deep implicit functions for 3d shape. In *Proc. CVPR*, 2020. 2

[13] Kyle Genova, Forrester Cole, Daniel Vlasic, Aaron Sarna, William T Freeman, and Thomas Funkhouser. Learning shape templates with structured implicit functions. In *Proc. ICCV*, 2019. 2

[14] Shubham Goel, Angjoo Kanazawa, and Jitendra Malik. Shape and viewpoint without keypoints. In *Proc. ECCV*, 2020. 3

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Proc. NeurIPS*, 2014. 2

[16] Amos Gropp, Lior Yariv, Niv Haim, Matan Atzmon, and Yaron Lipman. Implicit geometric regularization for learning shapes. In *Proc. ICML*, 2020. 2

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proc. CVPR*, 2016. 12

[18] Paul Henderson, Vagia Tsiminaki, and Christoph H Lampert. Leveraging 2d data to learn textured 3d mesh generation. In *Proc. CVPR*, 2020. 3

[19] Philipp Henzler, Niloy J Mitra, and Tobias Ritschel. Escaping plato's cave: 3d shape from adversarial rendering. In *Proc. ICCV*, 2019. 1, 3

[20] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Günter Klambauer, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a nash equilibrium. In *Proc. NeurIPS*, 2017. 6

[21] Chiyu Jiang, Avneesh Sud, Ameesh Makadia, Jingwei Huang, Matthias Nießner, and Thomas Funkhouser. Local implicit grid representations for 3d scenes. In *Proc. CVPR*, 2020. 2

[22] Yue Jiang, Dantong Ji, Zhizhong Han, and Matthias Zwicker. Sdfdiff: Differentiable rendering of signed distance fields for 3d shape optimization. In *Proc. CVPR*, 2020. 2

[23] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. In *Proc. SIGGRAPH*, 1984. 2

[24] Angjoo Kanazawa, Shubham Tulsiani, Alexei A. Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *Proc. ECCV*, 2018. 3

[25] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *Proc. ICLR*, 2018. 1, 2, 4, 8

[26] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proc. CVPR*, 2019. 1, 2, 3, 13

[27] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. In *Proc. CVPR*, 2020. 1, 2, 8, 12

[28] Petr Kellnhofer, Lars C. Jebe, Andrew Jones, Ryan Spicer, Kari Pulli, and Gordon Wetzstein. Neural lumigraph rendering. In *Proc. CVPR*, 2021. 2

[29] Amit Kohli, Vincent Sitzmann, and Gordon Wetzstein. Semantic implicit neural scene representations with semi-supervised training. *Proc. 3DV*, 2020. 2

[30] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. In *Proc. NeurIPS*, 2019. 13

[31] Yiyi Liao, Katja Schwarz, Lars Mescheder, and Andreas Geiger. Towards unsupervised learning of generative models for 3d controllable image synthesis. In *Proc. CVPR*, 2020. 3

[32] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. In *Proc. NeurIPS*, 2020. 2

[33] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. In *Proc. NeurIPS*, 2018. 12

[34] Shaohui Liu, Yinda Zhang, Songyou Peng, Boxin Shi, Marc Pollefeys, and Zhaopeng Cui. Dist: Rendering deep implicit signed distance function with differentiable sphere tracing. In *Proc. CVPR*, 2020. 2

[35] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proc. ICCV*, 2015. 1, 2, 5, 14

[36] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3d surface construction algorithm. *ACM TOG*, 21(4):163–169, 1987. 2, 7

[37] Sebastian Lunz, Yingzhen Li, Andrew Fitzgibbon, and Nate Kushman. Inverse graphics gan: Learning to generate 3d shapes from unstructured 2d data. *arXiv preprint arXiv:2002.12674*, 2020. 3

[38] N. Max. Optical models for direct volume rendering. *IEEE TVCG*, 1(2):99–108, 1995. 2, 4

[39] Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *Proc. ICML*, 2018. 4

[40] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proc. CVPR*, 2019. 2

[41] Mateusz Michalkiewicz, Jhony K Pontes, Dominic Jack, Mahsa Baktashmotlagh, and Anders Eriksson. Implicit surface representations as layers in neural networks. In *Proc. ICCV*, 2019. 2

[42] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *Proc. ECCV*, 2020. 2, 4, 7

[43] Siva Karthik Mustikovela, Varun Jampani, Shalini De Mello, Sifei Liu, Umar Iqbal, Carsten Rother, and Jan Kautz. Self-supervised viewpoint learning from image collections. In *Proc. CVPR*, 2020. 3

[44] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *Proc. ICCV*, 2019. 1, 2, 4, 5, 7

[45] Thu Nguyen-Phuoc, Christian Richardt, Long Mai, Yong-Liang Yang, and Niloy Mitra. Blockgan: Learning 3d object-aware scene representations from unlabelled images. In *Proc. NeurIPS*, 2020. 1, 2

[46] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Occupancy flow: 4d reconstruction by learning particle dynamics. In *Proc. ICCV*, 2019. 2

[47] Michael Niemeyer, Lars Mescheder, Michael Oechsle, and Andreas Geiger. Differentiable volumetric rendering: Learning implicit 3d representations without 3d supervision. In *Proc. CVPR*, 2020. 2

[48] Michael Oechsle, Lars Mescheder, Michael Niemeyer, Thilo Strauss, and Andreas Geiger. Texture fields: Learning texture representations in function space. In *Proc. ICCV*, 2019. 2

[49] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proc. CVPR*, 2019. 2

[50] Songyou Peng, Michael Niemeyer, Lars Mescheder, Marc Pollefeys, and Andreas Geiger. Convolutional occupancy networks. In *Proc. ECCV*, 2020. 2

[51] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Proc. AAAI*, 2018. 2, 3

[52] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. In *Proc. ICLR*, 2016. 2

[53] Sai Rajeswar, Fahim Mannan, Florian Golemo, Jérôme Parent-Lévesque, David Vazquez, Derek Nowrouzezahrai, and Aaron Courville. Pix2shape: Towards unsupervised learning of 3d scenes from images using a view-based representation. *IJCV*, 128(10):2478–2493, 2020. 3

[54] Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proc. ICCV*, 2019. 2

[55] Mehdi S. M. Sajjadi, Olivier Bachem, Mario Lučić, Olivier Bousquet, and Sylvain Gelly. Assessing Generative Models via Precision and Recall. In *Proc. NeurIPS*, 2018. 6, 13

[56] Tim Salimans, Ian J. Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Proc. NeurIPS*, 2016. 6

[57] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. In *Proc. NeurIPS*, 2020. 1, 2, 3, 4, 5, 6, 7

[58] Vincent Sitzmann, Eric R. Chan, Richard Tucker, Noah Snavely, and Gordon Wetzstein. Metasdf: Meta-learning signed distance functions. In *Proc. NeurIPS*, 2020. 2

[59] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit neural representations with periodic activation functions. In *Proc. NeurIPS*, 2020. 2, 12

[60] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. In *Proc. NeurIPS 2019*, 2019. 2

[61] Attila Szabó, Givi Meishvili, and Paolo Favaro. Unsupervised generative 3d shape learning from natural images. *arXiv preprint arXiv:1910.00287*, 2019. 3

[62] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Multi-view 3D models from single images with a convolutional network. In *Proc. ECCV*, 2016. 3

[63] Ayush Tewari, Ohad Fried, Justus Thies, Vincent Sitzmann, Stephen Lombardi, Kalyan Sunkavalli, Ricardo Martin-Brualla, Tomas Simon, Jason Saragih, Matthias Nießner, et al. State of the art on neural rendering. *Proc. Eurographics*, 2020. 2

[64] Shubham Tulsiani, Nilesh Kulkarni, and Abhinav Gupta. Implicit mesh reconstruction from unannotated image collections. *arXiv preprint arXiv:2007.08504*, 2020. 3

[65] Shubham Tulsiani, Hao Su, Leonidas J. Guibas, Alexei A. Efros, and Jitendra Malik. Learning shape abstractions by assembling volumetric primitives. In *Proc. CVPR*, 2017. 3

[66] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional GANs. In *Proc. CVPR*, 2018. 2

[67] Jiajun Wu, Chengkai Zhang, Tianfan Xue, William T. Freeman, and Joshua B. Tenenbaum. Learning a probabilistic latent space of object shapes via 3D generative-adversarial modeling. In *Proc. NeurIPS*, 2016. 2

[68] Shangzhe Wu, Christian Rupprecht, and Andrea Vedaldi. Unsupervised learning of probably symmetric deformable 3d objects from images in the wild. In *Proc. CVPR*, 2020. 2, 3

[69] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Ronen Basri, and Yaron Lipman. Multiview neural surface reconstruction by disentangling geometry and appearance. In *Proc. NeurIPS*, 2020. 2

[70] Weiwei Zhang, Jian Sun, and Xiaoou Tang. Cat head detection - how to effectively exploit shape and texture features. In *Proc. ECCV*, 2008. 1, 2, 5, 15

[71] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-Consistent adversarial networks. In *Proc. ICCV*, 2017. 2

[72] Jun-Yan Zhu, Zhoutong Zhang, Chengkai Zhang, Jiajun Wu, Antonio Torralba, Joshua B. Tenenbaum, and William T. Freeman. Visual object networks: Image generation with disentangled 3D representations. In *Proc. NeurIPS*, 2018. 2

A. Novel View Synthesis Details

We demonstrate a potential application of π-GAN: we can use a trained generator, without modifications, to perform single-view reconstruction. We base our method on the inverse projection procedure outlined by Karras et al. [27].

We freeze the parameters of our implicit representation and seek the frequencies $\boldsymbol{\gamma}_i$ and phase shifts $\boldsymbol{\beta}_i$ for each MLP layer i which produce a radiance field that, when rendered, best matches the target image. We initialize $\boldsymbol{\gamma}_i$ and $\boldsymbol{\beta}_i$ to $\bar{\boldsymbol{\gamma}}_i$ and $\bar{\boldsymbol{\beta}}_i$, the center of mass of frequencies and phase shifts for each layer. We calculate $\bar{\boldsymbol{\gamma}}_i$ and $\bar{\boldsymbol{\beta}}_i$ simply by averaging the frequencies and phase shifts of ten thousand random noise vector inputs. We then run gradient descent to minimize the mean-squared-error image reconstruction loss. We additionally introduce an $\mathcal{L}_2$ penalty with a weight of 0.1 during the optimization process to prevent $\boldsymbol{\gamma}_i$ and $\boldsymbol{\beta}_i$ from straying too far from $\bar{\boldsymbol{\gamma}}_i$ and $\bar{\boldsymbol{\beta}}_i$. We optimize the frequencies and phase shifts with the Adam optimizer over 700 iterations. We initialize the learning rate to 0.01, decaying by a factor of 0.5 every 200 iterations.

B. Model Details

Mapping Network. The mapping network is parameterized as an MLP with three hidden layers of 256 units each. The mapping network uses leaky-ReLU activations with a negative slope of 0.2.

SIREN-based Implicit Radiance Field. The FiLMed-SIREN [59] backbone of the generator is parameterized as an MLP with eight FiLMed-SIREN hidden layers of 256 units each.

Discriminator. Table 3 shows the architecture of the progressive discriminator. We begin training at low resolutions and progressively add discriminator stages while upsampling image size. In order to smooth transitions between upsamples, we fade in the contributions of new layers over ten-thousand iterations. We utilized CoordConv layers [33] and residual connections [17] throughout the discriminator.We considered using a patch discriminator similar to GRAF, but found it leads to uneven image quality as SIREN is prone to local overfitting to the last batch if sufficient coverage of the space is not maintained.

C. Additional Training Details

We train the majority of our models across two RTX 6000 GPUs or a single RTX 8000 GPU. We begin training at a resolution of 32×32, with an initial batch size of 120. At each upsample, we drop the batch size by a factor of four to keep the models and generated images in memory.

Figure 13: COLMAP reconstructions for models trained on CelebA, obtained by running COLMAP with default parameters and no known camera poses; GRAF's results were from their supplement.

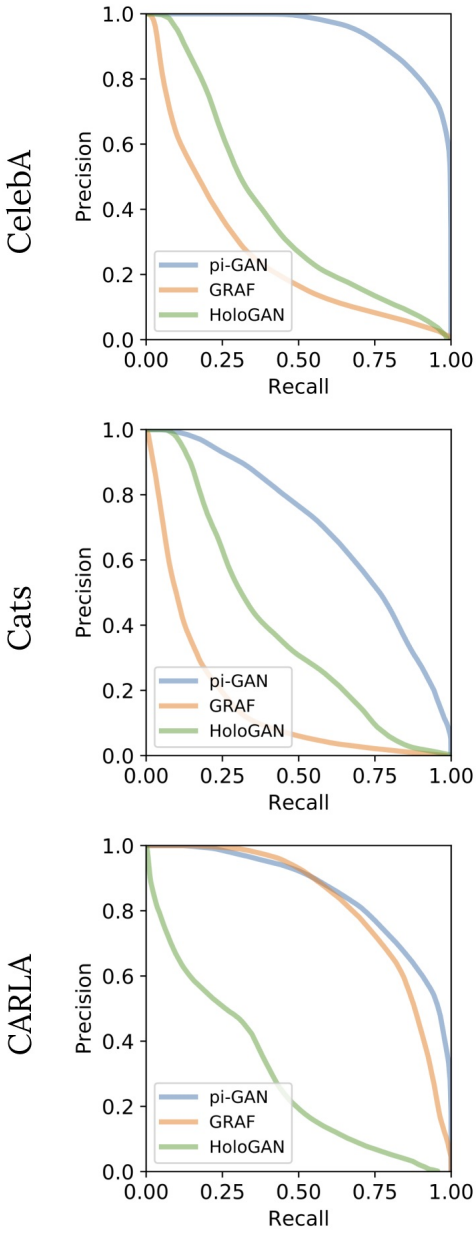

Figure 14: Precision-recall plots for π-GAN, GRAF, and HoloGAN on CelebA, Cats, and CARLA.

Table 3: Discriminator architecture, showing progressive growing stages.

	Activation	Output Shape
Input Image	-	3×128×128
Adapter Block (1×1)	LeakyReLU (0.2)	64×128×128
Coord Conv 1 (3×3)	LeakyReLU (0.2)	128×128×128
Coord Conv 2 (3×3)	LeakyReLU (0.2)	128×128×128
Avg Pool Downsample	-	128×64×64
Coord Conv 1 (3×3)	LeakyReLU (0.2)	256×64×64
Coord Conv 2 (3×3)	LeakyReLU (0.2)	256×64×64
Avg Pool Downsample	-	256×32×32
Coord Conv 1 (3×3)	LeakyReLU (0.2)	400×32×32
Coord Conv 2 (3×3)	LeakyReLU (0.2)	400×32×32
Avg Pool Downsample	-	400×16×16
Coord Conv 1 (3×3)	LeakyReLU (0.2)	400×16×16
Coord Conv 2 (3x3)	LeakyReLU (0.2)	400×16×16
Avg Pool Downsample	-	400×8×8
Coord Conv 1 (3×3)	LeakyReLU (0.2)	400×4×4
Coord Conv 2 (3×3)	LeakyReLU (0.2)	400×4×4
Avg Pool Downsample	-	400×2×2
Conv 2d (2×2)		1×1×1

Table 4: FID, KID mean × 100, and IS for π-GAN on CelebA, Cats, and CARLA datasets.

	FID ↓	KID ↓	IS ↑
CelebA @ 64×64	5.15	0.09	2.28
Cats @ 64×64	7.36	0.23	2.07
CARLA @ 64×64	13.59	0.34	3.85

At higher resolutions, we aggregate across mini-batches to keep an effective batch size at or above 12, given our GPU constraints. To further reduce memory usage, we used PyTorch's Automatic Mixed Precision (AMP). π-GAN trained for 10 hours at 32×32, 10 hours at 64×64, and 36 hours at 128×128. Certain rendering and camera parameters were tuned according to the dataset. We use the true pose distribution when it is known, e.g. for synthetic datasets, otherwise we make a guess and tune the distribution as a hyperparameter. We sample camera poses for CelebA from a normal distribution, with a vertical standard deviation of 0.15 radians and a horizontal standard deviation of 0.3 radians. We sample camera poses for Cats from a uniform distribution, with horizontal range $(-0.75, 0.75)$ and vertical range $(-0.4, 0.4)$. We sample poses for CARLA uniformly from the upper hemisphere. We tune the number of samples along each ray to balance memory consumption and depth resolution. We use 24 samples per ray for CelebA and Cats and 64 samples per ray for CARLA. We utilize a pinhole perspective camera with a field of view of 12° for CelebA, 12° for Cats, and 30° for CARLA.

D. π-GAN results @ 64×64

Table 4 includes additional quantitative results, evaluated at 64×64, in order to allow for comparisons of π-GAN against models evaluated at lower resolutions.

E. Additional Visual Results

We include additional visual results to show the image quality and view consistency of π-GAN. Figures 16 and 17 demonstrate the wide range of camera poses supported by π-GAN for generated faces and cats. Figure 15 shows the fine detail that π-GAN renders on larger images. Figure 18 shows additional cars with varying elevation and rotation. We include several videos of faces and cats with the camera following an elliptical trajectory in our supplementary video.

F. COLMAP Reconstruction

In order to demonstrate the images from π-GAN are multi-view consistent, we include a COLMAP reconstruction in Figure 13. We observe that proxy shapes extracted from pi-GAN lead to more pleasing novel views when projected to novel camera poses than those from GRAF.

G. Interpolation and Truncation

Following the method of StyleGAN [26] we can smoothly interpolate between two generated samples by linearly interpolating between the frequencies and phase shifts corresponding to the two latent codes. We include a result in Figure 8 in the paper. Along similar lines, it is also possible to trade off fidelity and diversity at test time following the method proposed in StyleGAN [26]. Because truncation reduced the diversity of generated images, we provided all evaluation metrics without truncation.

H. Precision and Recall

Recent work in generative models have investigated alternative metrics in order to independently evaluate fidelity and diversity [55, 30]. Figure 14 provides precision-recall plots on CelebA, Cats, and CARLA, comparing π-GAN to GRAF and HoloGAN.

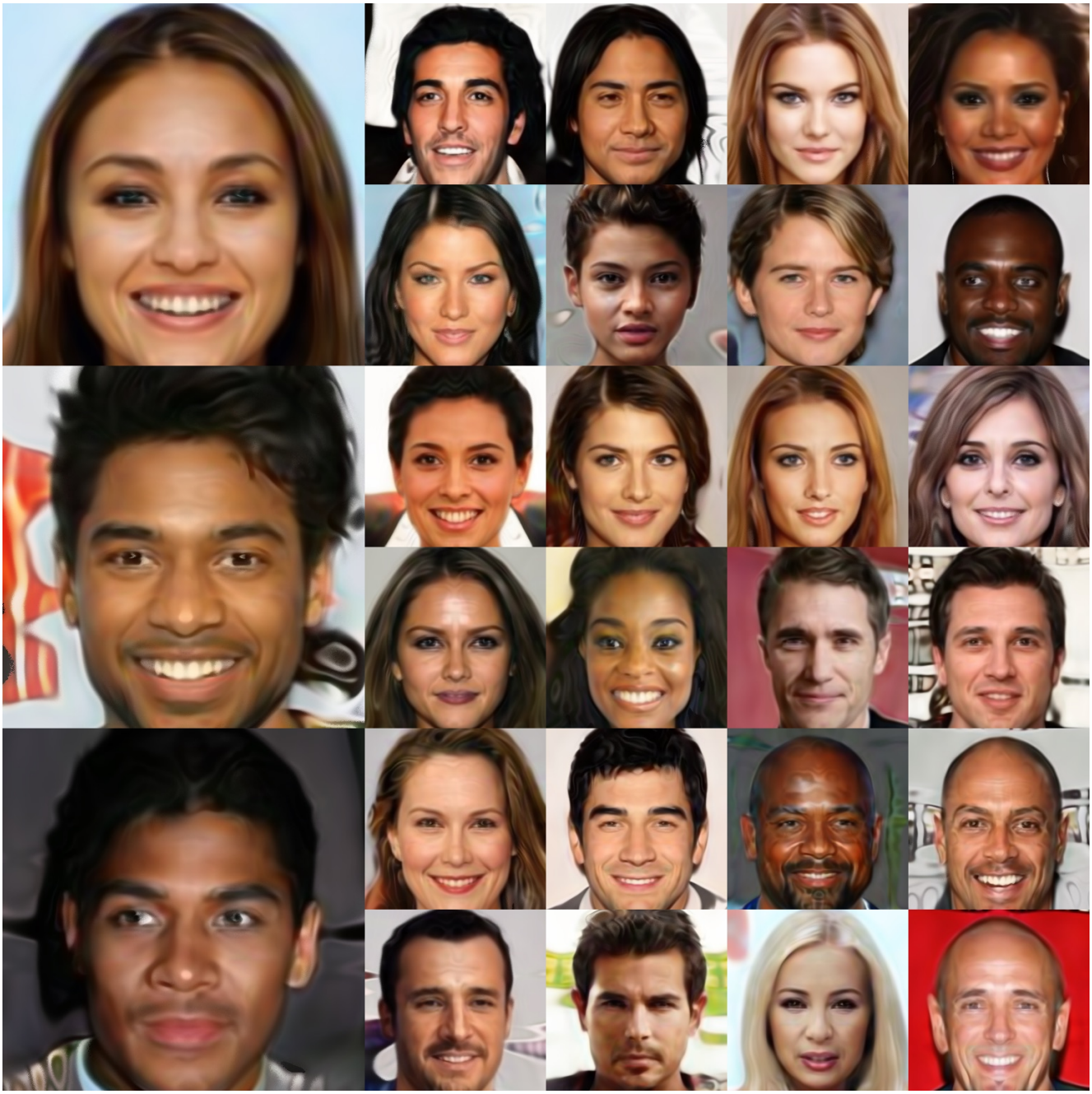

Figure 15: Curated examples from our model trained with CelebA [35].

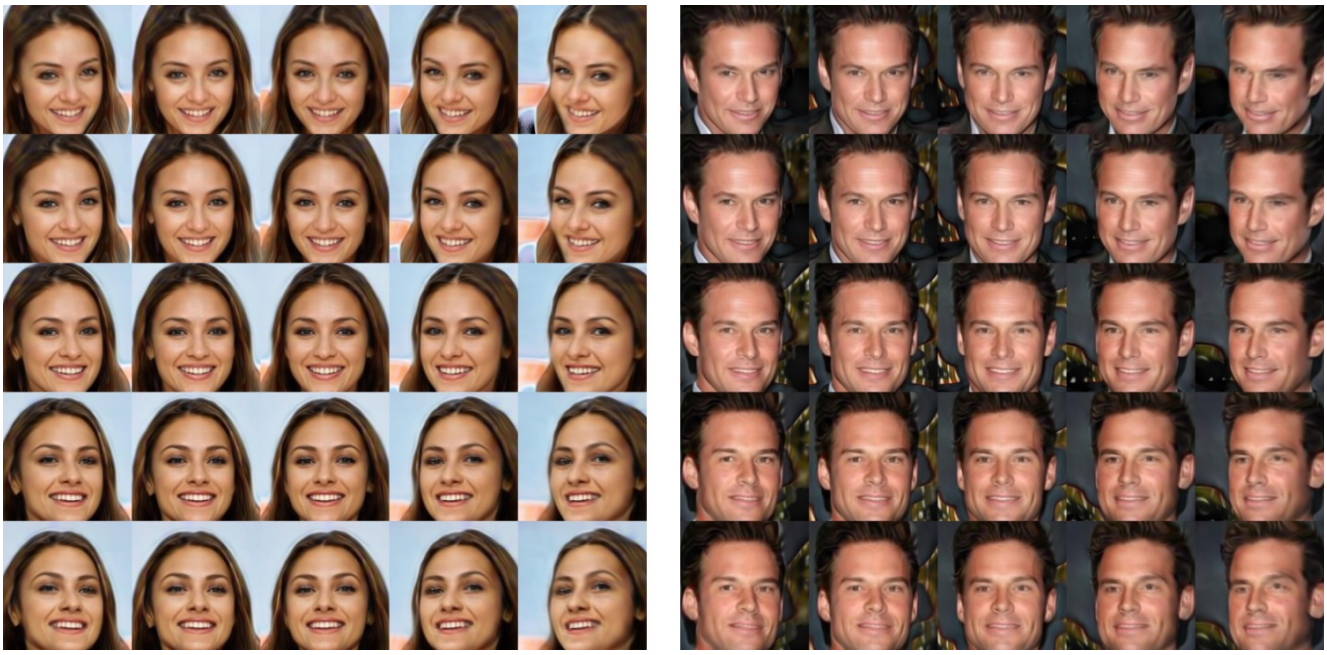

Figure 16: Curated examples from our model trained with CelebA, displayed from multiple viewing angles.

Figure 17: Curated examples from our model trained with Cats [70], displayed from multiple viewing angles.

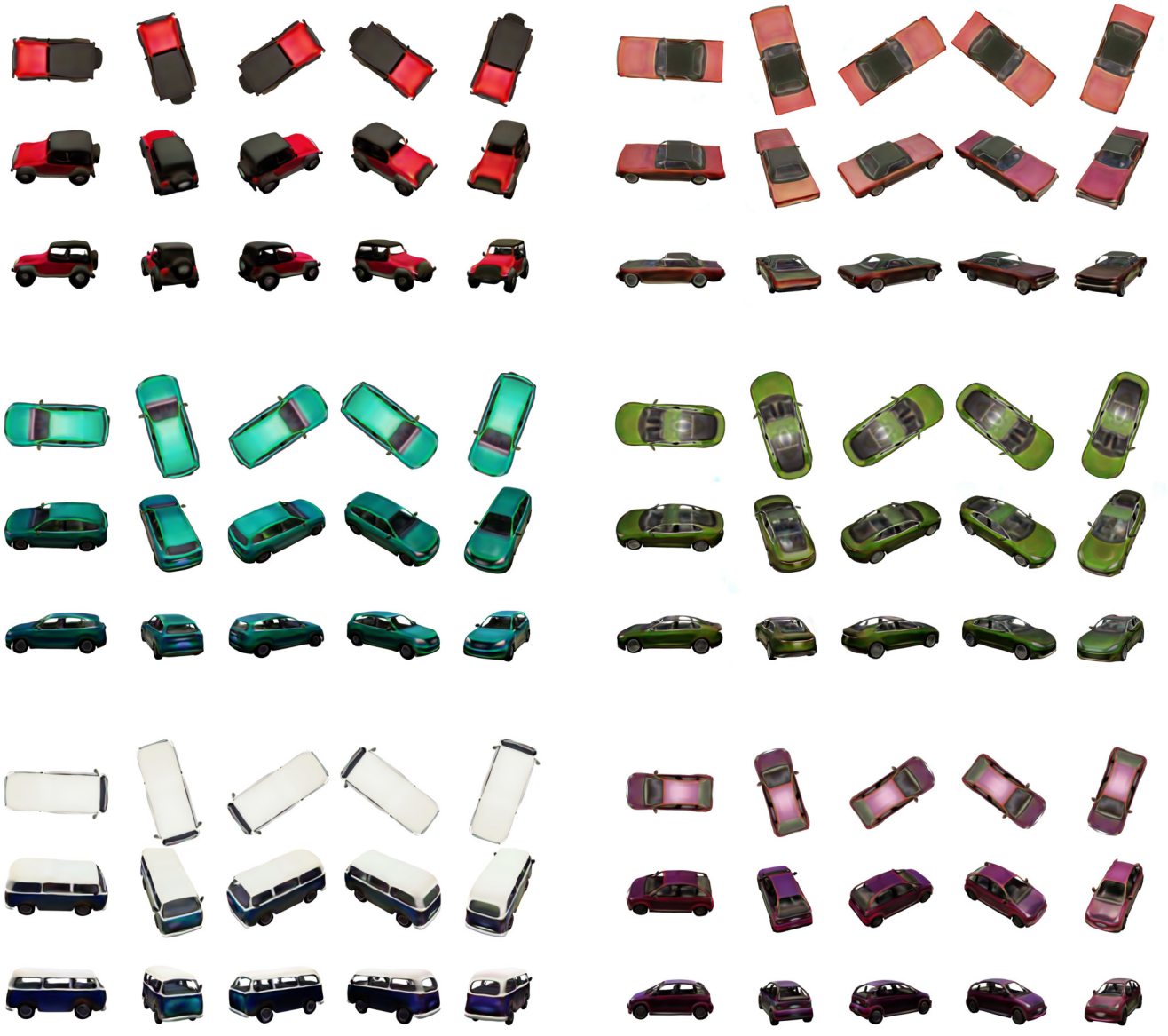

Figure 18: Curated examples from our model trained with CARLA [8], displayed from multiple viewing angles.