ViewNeRF: Unsupervised Viewpoint Estimation Using Category-Level Neural Radiance Fields

Octave Mariotti
omariott@ed.ac.uk
Oisin Mac Aodha
oisin.macaodha@ed.ac.uk
Hakan Bilen
hbilen@ed.ac.uk

School of informatics
University of Edinburgh
Edinburgh, UK

Abstract

We introduce ViewNeRF, a Neural Radiance Field-based viewpoint estimation method that learns to predict category-level viewpoints directly from images during training. While NeRF is usually trained with ground-truth camera poses, multiple extensions have been proposed to reduce the need for this expensive supervision. Nonetheless, most of these methods still struggle in complex settings with large camera movements, and are restricted to single scenes, *i.e*. they cannot be trained on a collection of scenes depicting the same object category. To address these issues, our method uses an analysis by synthesis approach, combining a conditional NeRF with a viewpoint predictor and a scene encoder in order to produce self-supervised reconstructions for whole object categories. Rather than focusing on high fidelity reconstruction, we target efficient and accurate viewpoint prediction in complex scenarios, *e.g*. 360° rotation on real data. Our model shows competitive results on synthetic and real datasets, both for single scenes and multi-instance collections.

1 Introduction

Understanding the 3D world from images is a longstanding goal in computer vision. Recovering camera viewpoint (*i.e*. pose) is a key step in establishing the correspondence between 2D images and the 3D world. Camera viewpoint is typically defined relative to a coordinate system in a single 3D scene or to a canonical frame in an object category. The former case often involves analysis from multiple views of the same scene (*e.g*. Structure from Motion (SfM) [22]) where recovered viewpoints are tied to a single scene. The latter case aims to estimate viewpoint relative to unseen object instances from a specific category (*e.g*. [10, 23]). It is often more challenging, as it requires invariance to not only view-dependent changes but also to shape and texture variations among instances of an object category. In this paper, we focus on the latter case, *category-level viewpoint estimation*.

A notable difficulty in learning to estimate viewpoint is obtaining reliable ground truth poses. This process is typically complex, requiring either time-consuming manual annotations or calibrated laboratory setups, which limits the applicability of supervised learning. To this end, multiple approaches have been proposed to learn category-level poses from image collections *without* ground-truth supervision [11, 25, 30, 45]. These works broadly use an analysis-by-synthesis approach by disentangling viewpoint and 3D appearance from images and render objects from new viewpoints, hence enabling the learning of object poses without external pose supervision. As the supervision from the reconstruction relies on projection of the 3D scene, it is essential that these methods are able to model 3D information as accurately as possible. However, prior work either builds on simplistic pseudo-rendering methods through neural decoders [24, 30, 31] that do not preserve the 3D geometry in rendering, or on classical 3D representations such as voxels [25, 45] or point clouds [11] that fail to produce high-quality reconstructions.

Recently, neural radiance fields (NeRF) [29] have achieved unprecedented quality in the 3D reconstruction and rendering of scenes. Deviating from the traditional geometrically-explicit representations, NeRF belongs to the family of implicit 3D representations [4, 21, 28, 33, 40], encoding 3D data in the weights of a neural network which allows them to operate at continuous 3D coordinates and hence at high resolution. In addition, their viewpoint-dependent rendering enables the modeling of material properties such as reflections. Despite the recent progress [2, 20, 26, 41, 51, 53], they still typically require accurate camera poses during training, and are also limited to modeling a single scene at a time. This restricts their application to controlled settings, with labeled poses, or to scenes where many high-quality camera poses can be obtained via SfM. Recent works, which aim to alleviate these problems, can either operate only on simple forward-facing scenes without accurate camera pose initialization or only work on a single scene or object instance [13, 19, 27, 47]. Those that model multiple instances, require ground-truth camera poses and/or expensive test-time optimization in order to synthesize novel views [12, 52] (see Table 1). Hence, these models cannot be trivially used to learn category-level poses without training pose supervision.

Motivated by these limitations, we propose an analysis-by-synthesis approach that leverages the powerful 3D modeling ability of NeRFs for unsupervised category-level viewpoint estimation. Specifically, our model, *ViewNeRF* (i) simultaneously learns to estimate shape, appearance, and pose of object instances, enables single-shot pose prediction on unseen views more efficiently than the gradient descent-based pose estimation used in methods like [12, 13, 19, 47, 50], (ii) works on multiple instances of an object category via a conditional NeRF model and extends the classic single-scene setting used in pose-free NeRFs [27], and (iii) obtains significantly more accurate pose predictions compared to existing unsupervised method [25] across multiple benchmarks.

2 Related work

Unsupervised viewpoint estimation. Despite the large body of supervised methods [6, 18, 35, 43], viewpoint estimation is still a challenging task due to the cost of building large labeled datasets. Hence a growing number of methods attempts to limit the amount of supervision needed at training time. SfM approaches such as COLMAP [36] use multi-view geometry to infer camera poses using only images, but are limited to single scenes, require many views, and are thus unsuitable for estimating poses across category-centric datasets. Recently several deep learning pose-estimation methods have been proposed that utilize var-

	Pose-free 360° training	Real data 360° training	One shot pose on new views	Multiple scenes
NeRF [30]				
INeRF [60]				
NeRF– [47]†				
BARF [19]				
SCNeRF [13]		✓		
GaRF [6]†				
GNeRF [27]	✓		✓‡	
CodeNeRF [12]				✓
ViewNeRF (Ours)	✓	✓	✓	✓

Table 1: Comparison of pose-free NeRF methods on 360°scenes. Most of these approaches require ground-truth poses or initial estimates.†Untested on 360°scenes. ‡Contains a pose estimator but it is not used during evaluation.

ious amounts of pose supervision including semi-supervised [24, 46], few/zero-shot learning [1, 7, 48], and unsupervised methods [11, 25, 30, 45]. Most related to us, unsupervised methods typically learn to disentangle category-level pose and appearance using an analysis-by-synthesis pipeline. ViewNet [25] generates a voxel-based reconstruction of a specific object instance and renders it from the predicted viewpoint. This approach is limited by the spatial resolution of the voxel grid, fails to reconstruct fine details and viewpoint dependent illumination effects. SSV [30] adopts a generative approach, using 3D latent feature maps to represent the scene. However, it does not enforce geometric consistency between decoded viewpoints, resulting in noisy viewpoint estimation due to its decoder's flexibility and ability to overfit to geometrically implausible poses.

Neural Radiance Fields (NeRFs). 3D data is traditionally represented using meshes [16, 17], voxels [39, 44, 45, 49], or point clouds [11]. Recently, implicit representations [4, 21, 28, 33, 40] have emerged as an effective tool in 3D modeling. They represent 3D data implicitly in the parameters of a fully connected neural network that takes 3D coordinates as input and predicts properties such as their occupancy and color of the imaged scene at the specified 3D location. NeRF [29] models 3D scenes by mapping both 3D coordinates and the viewing direction to RGBA space, achieving breakthrough performances in novel view synthesis. Multiple works have extend the NeRF paradigm targeting higher quality reconstruction [2, 54] and faster runtime [20, 51, 53]. Two directions, particularly related to object pose estimation, is the extension of NeRF beyond the strict single-scene setting and the removal on the dependency for training time poses.

Growing out of the fixed-scene setting, NeRF in the Wild [26] learns to aggregate views of the same scene taken in different settings by learning image-specific embeddings, while Nerfies [34] learn to deform rays to represent deformable objects. PixelNeRF [52] learns scene-based dense embeddings to represent multiple scenes. CodeNeRF [12] disentangles shape and texture across instances from the same category. These methods, however, require ground-truth camera poses during training. Though several generative methods [8, 32, 37] have been proposed to model object categories, they are unable to estimate poses, and employ neural rendering that can violate the scene geometry as in SSV [30]. Unlike them, through the use of an implicit 3D representation and analytical rendering, our reconstructions are 3D consistent.

Multiple pose-free NeRF methods [6, 13, 19, 47] have been proposed that attempt to learn camera poses during training by refining initial pose estimates through gradients coming from the NeRF model itself. However, these methods are only pose-free on forward-facing scenes, needing COLMAP as initialization for 360° scenes, if they even are evaluated

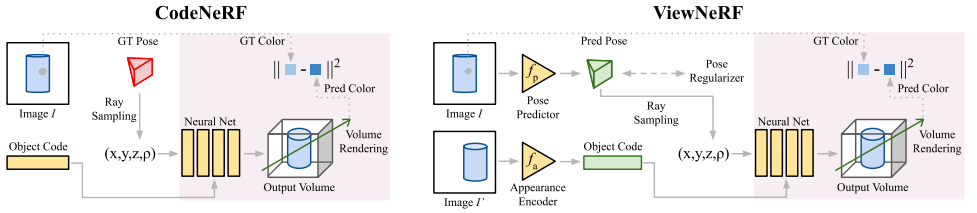

Figure 1: (Left) CodeNeRF [17] requires ground-truth pose information at training time and performs expensive direct optimization for each object appearance code. In practice, CodeNeRF also enforces that the object code for distinct views of the same object are the same which provides some multi-view signal. (Right) In contrast, our ViewNeRF approach is fully self-supervised by making use of a separate pose predictor f_p and appearance encoder f_a that can be applied to any image in a single-shot fashion.

on such challenging settings. While NeRF methods can be used to retrieve the pose of new images under certain condition [17, 50], they require expensive test-time optimization. By comparing image reconstructions based on initial noisy pose estimates to the target image, they perform many gradient descent steps on the camera parameters to gradually align the two images. In addition to being a slow process, this approach can get trapped in local minima in the multi-object setting. In comparison, our model can predict the pose of unseen instances in a single forward pass. A notable exception from other pose-free NeRFs is GNeRF [27], which can operate without initialization thanks to its adversarial training. However, it is limited to single scenes and is slow to train as a result of the additional GAN-based objective. A comparison to related NeRF-based approaches is shown in Table 1.

3 Method

Here we outline the main components and training procedure for our ViewNeRF model. Unlike prior methods, ViewNeRF is capable of estimating the pose for held out images and can be trained on images of multiple instances (*i.e.* from different scenes) of the same object category *without* requiring any ground-truth pose supervision.

Our goal is to estimate the viewpoint (*i.e.* camera pose) of an unseen object instance from a known category (*e.g.* car, chair, *etc.*) in an image. To this end, we wish to learn a function, f_p that takes an image I as input and outputs the corresponding viewpoint represented as the rotation and translation, *i.e.* $p = f_p(I)$. As ground-truth viewpoints are not available for training, we treat learning pose prediction as an image reconstruction problem. In the same vein as [11, 12, 25, 45, 52], we exploit multi-view information in the form of image pairs. Specifically, given N unlabeled image pairs $\{I_n, I'_n\}_{n=1}^{N}$, where each pair contains source and target images of the same object instance which differ only in their unknown viewpoints, our objective is to reconstruct the target image I from the source image I'.

Clearly, one needs a good estimate of the viewpoint of I in order to reconstruct it from I'. During training we reconstruct the target image I using its estimated viewpoint $f_p(I)$ and the appearance information extracted from our viewpoint-independent appearance encoder f_a for the source image I'. For rendering, we pass the pose and appearance information to a NeRF-based decoder f_r, to reconstruct the target image I, and minimize an image reconstruction loss to simultaneously learn the weights of f_p, f_a, and f_r which are instantiated as

neural networks,

$$\min_{f_p, f_a, f_r} \frac{1}{N} \sum_{n=1}^{N} \mathcal{L}(f_r(f_a(I'_n), f_p(I_n)), I_n) + \lambda \mathcal{L}_{\text{reg}}(f_p(I_n)). \quad (1)$$

$\mathcal{L}$ is a loss function that measures the difference between the reconstructed and target images, and $\mathcal{L}_{\text{reg}}$ is a pose regularization term applied to the viewpoint predictions, which is weighted by a scalar λ. The training pipeline for our model, ViewNeRF, is shown in Fig. 1 (right).

Clearly the target image cannot be successfully reconstructed without its viewpoint information. However, in the case of an arbitrary decoder, where the appearance and viewpoint encodings are passed through multiple arbitrary nonlinear transformations, there are at least two challenges: it is not guaranteed that (i) the estimated viewpoints are disentangled from the appearance and, if they are, (ii) the estimated viewpoints are geometrically meaningful transformations. Hence, it is crucial that the decoder utilizes the estimated viewpoint in a geometrically consistent way.

3.1 NeRF decoder - f_r

NeRFs are originally formulated as 3D models that learn to reconstruct what a scene looks like when observed from a specified viewpoint p. Formally, they combine a volume rendering operation [15] with a neural network that learns to map an input $(\mathbf{x}, \rho) \in \mathrm{R}^3 \times \mathcal{S}^2$, consisting of a 3D coordinate and a viewing direction, to a 4D vector (r, g, b, σ) in RGBA space. To reconstruct an image, rays corresponding to each pixel are cast from a camera p, and the network is queried multiple times along each ray to estimate the color and occupancy at those locations. Then, values along the ray are integrated according to their density to form a single pixel value. An important property of NeRF is that the predicted occupancy σ only depends on the input coordinates $\mathbf{x}$. This coupled with its analytical rendering, grants them relatively strong 3D consistency (although not perfect [54]). Along with its high resolution, this property makes it particularly suitable as a decoder in our pose estimation pipeline.

Object appearance conditioned on a NeRF decoder. Standard NeRFs are trained to model a single 3D scene, effectively memorizing its shape and appearance from multiple viewpoints. In a category-based setting, this would mean training an individual model for each object instance, which would be very time consuming and no information across instances would be shared. Hence, a better approach is to implement a conditioning mechanism to allow the NeRF decoder f_r to reconstruct specific instances. This conditioning should be pose-agnostic in order to let the pose predictor f_p learn to disentangle pose from appearance. Therefore, we adopt a strategy inspired by ViewNet [25] and CodeNeRF [12] where object instances are fully described by a latent object code $\mathbf{a}$. Similar to [12], $\mathbf{a}$ is mapped at different depths of the NeRF model to condition its activations, and similar to [25], $\mathbf{a}$ is predicted by an appearance network f_a that learns a global latent space shared over all object instances.

3.2 Pose estimator - f_p

Conventional NeRF methods requires ground-truth camera poses during training. Recent extensions [6, 13, 19, 47, 50] allows for estimating camera poses with NeRF by letting reconstruction gradients flow to the camera parameters. However, this approach suffers from two issues: i) it is computationally expensive, requiring hundreds of steps to converge, if at all, and ii) it can get stuck in local minima, limiting its operation to forward-facing scenes when a reasonable pose initialization is not available. Following the recent advances in viewpoint estimation [11, 25, 30, 45], we posit that a better solution is to estimate poses

directly from images using a pose predictor f_p. This enables fast prediction during inference and generalizes efficiently to new object instances. However, f_p is still subject to local minima and may not receive meaningful gradients from f_r, as reconstruction errors can arise either from the reconstruction process or an incorrect pose prediction. This can lead to the collapse of pose predictions and to degenerate solutions where the model relies only on the appearance encoding $\mathbf{a}$ to reconstruct I'. Next we introduce two mechanisms to prevent this.

Multi-hypothesis predictions. We supply f_p with a multi-head predictor as used in [11, 25]. During training, we let the pose estimator output multiple hypotheses $f_p(I) = p_1, \ldots, p_K$, and each of them is fed to f_r to produce a low resolution reconstruction. These are then compared to the target, and the pose p^* resulting in the best reconstruction is selected. p^* is then passed again to the NeRF decoder, this time at full resolution. For inference, a student head is jointly trained to predict p^*, removing the need for multiple outputs [25].

Pose regularization. Multiple pose predictions alone are not always sufficient to prevent training collapse or instability as all heads can still predict the same pose. Inspired by generative models like [8, 32] that sample poses during training, we encourage the predicted pose distribution to follow a prior distribution $\mathcal{P}$. As generative models do not aim to reconstruct images from a specific viewpoint, they can directly sample a pose from the prior $p \sim \mathcal{P}$ and use it to generate an image. However, this is unfeasible in our case, as a random pose would not match that of the specific image that we would like to reconstruct. Instead, we attempt to match batch-wise distributions, following the assumption that a batch of predicted poses should closely follow the pose prior $\mathcal{P}$.

Specifically, given a batch of B predicted poses $p^*_{1,\ldots,B}$, we sample K pseudo-targets $p'_{1,\ldots,K} \sim \mathcal{P}$ and compute for each p'_i its closest match p^*_j in the batch. Finally, the distance $||p'_i - p^*_j||^2$ is added to the loss, *i.e.* $\mathcal{L}_{\text{reg}} = \frac{1}{K}\sum_{i=1}^{K} \min_{j=1\ldots B} ||p'_i - p^*_j||^2$. This prevents the collapse of all predictions to a single point while being very cost-efficient. To prevent unnecessary noise, the regularization weight λ in Eq. (1) is progressively tuned down during training. Further details are provided in the supplementary.

3.3 Reconstruction objective

Reconstructing full resolution images requires millions of queries to the NeRF decoder and hence is expensive, so NeRFs are usually trained by sampling a subset of pixels per image per iteration. This strategy works well when using ground-truth camera poses, as each sampled pixel corresponds to one exact ray that will stay constant during training. However, when jointly estimating poses and training the NeRF model, it can introduce a significant amount of noise, as the randomly selected pixels might not contain enough relevant information to recover incorrectly estimated poses. In particular, some object categories such as cars can exhibit symmetries that can only be broken by focusing on fine details (*e.g.* color of the headlights).

To this end, we deviate from the standard NeRF training procedure and instead use reconstructions of the entire image, however, in low-resolution coupled with a perceptual loss [14] (denoted as $\mathcal{L}$ in Eq. (1)) that provides a finer structure-preserving objective. As the perceptual loss aims to match activation maps from a pretrained network, it is more sensitive to salient image features like edges that would be missing or misplaced under wrongly estimated poses.

	Supervised				Unsupervised			
	CodeNeRF, w/o init		CodeNeRF, w/ init		ViewNet		Ours	
	car	chair	car	chair	car	chair	car	chair
Accuracy at 10°(%, ↑)	08.5	03.4	**82.1**	**60.2**	61.2	76.7	**70.0**	**82.8**
Median rotation error (°, ↓)	115	108	**3.53**	**7.70**	6.54	4.25	**5.71**	**4.18**
Median translation error (%, ↓)	139	134	**5.9**	**13.9**	n/a	n/a	**8.0**	**6.4**

Table 2: Multi-instance results on ShapeNet-SRN. CodeNeRF pretrained models were kindly provided by the authors. When initialized, pose estimates were randomly drawn within 30°of the ground-truth, where **bold** results indicate the best model per category.

4 Experiments

4.1 Implementation details

We use EfficientNet [42] backbones for our pose and appearance encoders f_p and f_a, and a compact NeRF model with two fully connected layers with 128 dimensions for the decoder. Note that our decoder is designed to be significantly smaller than the one used in [27] and [12] to discourage unnecessarily complex mappings and to make it more efficient. Computational efficiency is particularly important in our experiments, as we use a perceptual loss for reconstruction error which requires generating a full image.

Camera pose prediction. To link the predictions of f_p to real world poses, we need to ensure it can be interpreted as such. Similar to ViewNet and GNeRF, we formulate pose as a point on the 3D unit sphere $\mathcal{S}^2$ from which we derive a camera matrix using a Gram-Schmidt orthogonalization process. While other representations like quaternions are possible, this provides a simple way to enforce the necessary constraints and exhibits favorable properties for optimization[55]. For synthetic datasets, we follow GNeRF's assumption that the object is located at the center of the scene where the camera is pointed to, and that the camera is held upright, *i.e.* it is aligned with the z vector in world coordinates. On the Freiburg Cars dataset [38], this assumption is not valid as data was hand-recorded. Therefore, we additionally allow our viewpoint estimator to predict a camera distance, target point, *i.e.* a point along the camera principal axis, and an upwards direction to account for in-plane rotation. Instead of hard constraints, we set soft target for the first 10 epochs of training. Additional implementation details can be found in the supplementary material.

As poses are predicted up to an arbitrary rotation, following the evaluation in [25], we align our estimated camera poses with the ground-truth labels by solving an orthogonal Procrustes problem. As our main goal is to estimate viewpoint from single images rather than high-fidelity reconstruction, we evaluate our method in terms of viewpoint accuracy, reporting rotation and translation errors. The other approaches we compare to, [12, 25, 27], each use their own sets of metrics making direct comparison difficult. Hence, for fair comparison, we re-evaluate their models and report viewpoint accuracy with a 10° threshold, along with median rotation and translation error for each method in all experiments. Taking the median instead of the average provides a less noisy estimate in the presence of strong symmetries [43]. Translations errors are normalized by the camera distance to the origin to account for scaling differences.

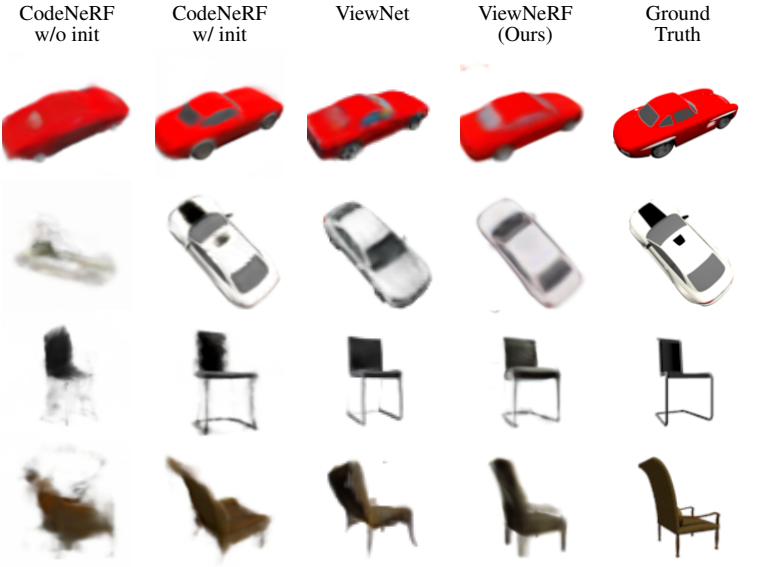

Figure 2: Comparison of reconstructions from the supervised CodeNeRF, unsupervised ViewNet, and our unsupervised ViewNeRF.

4.2 Multi-instance results

In Table 2 we first evaluate our ViewNeRF approach on the ShapeNet-SRN dataset [40] which contains renderings of ShapeNet [3] cars and chairs. We compare to CodeNeRF [42], a *supervised* NeRF-based model and the *unsupervised* voxel-based ViewNet [25]. While ViewNet reports results on ShapeNet, it is only trained on a limited set of viewpoints, *i.e.* the elevation of views only spans $[-20°, 40°]$, instead of the full range in ShapeNet-SRN. Hence, we retrain it using code from the authors on this new data split.

CodeNeRF requires expensive test-time optimization to perform pose estimation and only reports results for a *single* object instance in their paper, *i.e.* not multiple instances from the same category, starting from hand-selected poses[1]. Therefore, we re-evaluated it on each test instance, under two settings, a realistic one where the starting pose is uniformly sampled according to the training distribution ('w/o init'), and an easier setting, in which the initial pose is chosen to be within 30° of the ground-truth ('w/ init').

The results in Table 2 illustrate that CodeNeRF, despite being trained with ground-truth pose, is unable to properly estimate pose when the initial estimate is noisy. Since both pose and object embeddings have to be jointly optimized, the process can converge to a degenerate solution, relying mostly on the appearance embeddings rather than pose to minimize the reconstruction error. Test-time reconstructions shown in Fig. 2 confirm this. While reconstruction with good initializations are accurate, noisy initialization results in poor reconstructions. Finally, compared with ViewNet, our approach reaches higher pose prediction performance by making fewer gross pose errors, *e.g.* ViewNet predicts in the wrong orientation for the car in the second row of Fig. 2.

[1] Confirmed via correspondence with the authors.

Figure 3: Comparison of reconstructions from the unsupervised ViewNet and ViewNeRF on held-out test instances from the Freiburg Cars dataset.

	ViewNet	Ours	Ours, no predictor	Ours, MSE	Ours, singlehead	Ours, subsampled	Ours, no reg.
Accuracy at 10°(%, ↑)	50.0	**73.5**	00.8	04.1	11.4	13.5	54.5
Median rotation error (°, ↓)	9.99	**8.05**	90.8	91.0	67.4	93.1	9.09

Table 3: Comparison to ViewNet and ablated versions of our ViewNeRF method on the Freiburg Cars dataset.

4.3 Real scenes results

Here we demonstrate ViewNeRF's ability to work on real images using the Freiburg Cars dataset [38]. The target car instance in each image is first segmented using MaskRCNN [9], and out of the 48 scenes, the first 40 are used for training, the next three for validation, and the remaining five for testing. As the data is only labeled with weak viewing direction information, we only report rotation-base metrics. The results in Table 3 illustrate a large gap between the performances of our approach and ViewNet. We mostly attribute it to ViewNet's inability to model the complex illumination patterns (*e.g.* reflections) on real cars. Qualitative results in Fig. 3 further illustrate this, *i.e.* reconstructions from ViewNet, while having sharper colors are very noisy. In addition, it seems that ViewNet is unable to differentiate the front from the back of the red car.

Ablated models. To validate our design choices, we also evaluate multiple ablated versions of our model on the Freiburg Car dataset. We evaluate four variations: (i) removing estimators and trying to learn poses directly with backpropagation, *i.e.* standard pose-free NeRF training, (ii) using a MSE loss instead of the perceptual loss for reconstruction, (iii) using a single pose hypothesis, (iv) subsampling 1024 random pixels per image instead of using low resolution full reconstructions, and (v) removing pose regularization $\mathcal{L}_{\text{reg}}$. All ablations produce worse performance, with the first three resulting in catastrophic failure.

4.4 Single instance results

Finally we evaluate ViewNeRF in the single-scene setting with full 360° rotations on the synthetic-NeRF datasets [29] used in GNeRF [27]. GNeRF is closely related to our model

	GNeRF			Ours		
	Acc (%,↑)	MR (°,↓)	MT (%,↓)	Acc (%,↑)	MR (°,↓)	MT (%,↓)
Chair	**100**	**2.645**	**4.401**	**100**	3.012	4.680
Drums	**98.5**	**3.307**	**5.489**	80.5	5.212	8.356
Hotdog	74.5	7.120	11.12	**96.0**	**2.412**	**3.898**
Lego	**91.5**	5.153	8.313	87.0	**4.659**	**7.571**
Mic	**97.5**	**3.022**	**4.787**	93.5	4.169	6.823
Ship	15.0	28.23	43.56	**69.5**	**6.674**	**9.946**

Table 4: NERF synthetic single scenes. Acc: Accuracy at 10°, MR: Median rotation error, MT: Median translation error. GNeRF models were retrained using published code.

as it can estimate pose with a simple single forward pass. However, the pose results reported in the original GNeRF paper are from the training split of the data, where they are learned using a mixture of gradient descent optimization and soft-labeling. Similarly, COLMAP [36] is evaluated directly over the training images. In Table 4 we instead evaluate the GNeRF pose predictor on the test split in order to perform a fair comparison with our model. We observe that in spite of its strong performance on the training split and the much larger model it uses, GNeRF results are broadly comparable to ours during inference. This can be explained by the size of the training set, that only contains 100 samples, hinting towards overfitting. The ship scene exhibits a strong rotational symmetry, and is thus particularly challenging for both methods.

4.5 Limitations

Although our model outperforms prior works in category-based viewpoint estimation, it also has certain limitations that hinders its applicability on more complex scenarios. While the requirement for multiple views at training time is common, it limits our approach to multi-view datasets. It would be desirable to have a model that can learn object categories from different instances without requiring multi-view data. While generative methods (*e.g.* [8, 30, 31, 32]) possibly possess this ability, they also employ neural-based decoders that hurt 3D consistency. Another limitation is the need for segmenting foreground object from cluttered background. We observed that when using unsegmented views, complex backgrounds that form the majority of an image prevented NeRF from paying enough attention to the object to capture the details needed for estimating category-level pose. Forcing our model to focus less on the background during training by using two separate NeRFs as in [32] could be a potential solution.

5 Conclusion

We presented ViewNeRF, a conditional NeRF-based method for accurate category-centric pose estimation that is trained from self-supervision alone. Through careful design choices, our ViewNeRF model manages to predict accurate viewpoints during training and testing across a wide variety of real and synthetic datasets, going beyond what previous NeRF-based models are capable of. Through extensive evaluation, we show the pitfalls of the gradient descent-based pose recovery that is used in many NeRF pipelines. We compare our method with other related self-supervised approaches and illustrate the benefits of NeRF over explicit 3D modeling for the challenging task of single image pose estimation.

Acknowledgment. HB is supported by the EPSRC Visual AI grant EP/T028572/1.

References

[1] Mohamed El Banani, Jason J Corso, and David F Fouhey. Novel object viewpoint estimation through reconstruction alignment. In *Computer Vision and Pattern Recognition*, pages 3113–3122, 2020.

[2] Jonathan T Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields. In *International Conference on Computer Vision*, pages 5855–5864, 2021.

[3] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015.

[4] Zhiqin Chen and Hao Zhang. Learning implicit fields for generative shape modeling. In *Computer Vision and Pattern Recognition*, pages 5939–5948, 2019.

[5] Shin-Fang Chng, Sameera Ramasinghe, Jamie Sherrah, and Simon Lucey. Garf: Gaussian activated radiance fields for high fidelity reconstruction and pose estimation. *arXiv e-prints*, pages arXiv–2204, 2022.

[6] Christopher B Choy, Danfei Xu, JunYoung Gwak, Kevin Chen, and Silvio Savarese. 3d-r2n2: A unified approach for single and multi-view 3d object reconstruction. In *European conference on computer vision*, pages 628–644. Springer, 2016.

[7] Walter Goodwin, Sagar Vaze, Ioannis Havoutis, and Ingmar Posner. Zero-shot category-level object pose estimation. *arXiv preprint arXiv:2204.03635*, 2022.

[8] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. Stylenerf: A style-based 3d-aware generator for high-resolution image synthesis. In *ICLR*, 2021.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *International Conference on Computer Vision*, pages 2961–2969, 2017.

[10] Daniel P Huttenlocher and Shimon Ullman. Object recognition using alignment. In *Proceedings of the DARPA Image Understanding Workshop*, pages 370–380, 1987.

[11] Eldar Insafutdinov and Alexey Dosovitskiy. Unsupervised learning of shape and pose with differentiable point clouds. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2802–2812, 2018.

[12] Wonbong Jang and Lourdes Agapito. Codenerf: Disentangled neural radiance fields for object categories. In *International Conference on Computer Vision*, pages 12949–12958, 2021.

[13] Yoonwoo Jeong, Seokjun Ahn, Christopher Choy, Anima Anandkumar, Minsu Cho, and Jaesik Park. Self-calibrating neural radiance fields. In *International Conference on Computer Vision*, pages 5846–5854, 2021.

[14] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *International Conference on Computer Vision (ECCV)*, pages 694–711, 2016.

[15] James T Kajiya and Brian P Von Herzen. Ray tracing volume densities. *ACM SIGGRAPH computer graphics*, 18(3):165–174, 1984.

[16] Angjoo Kanazawa, Shubham Tulsiani, Alexei A Efros, and Jitendra Malik. Learning category-specific mesh reconstruction from image collections. In *European Conference on Computer Vision*, pages 371–386, 2018.

[17] Hiroharu Kato, Yoshitaka Ushiku, and Tatsuya Harada. Neural 3d mesh renderer. In *Computer Vision and Pattern Recognition*, pages 3907–3916, 2018.

[18] Wadim Kehl, Fabian Manhardt, Federico Tombari, Slobodan Ilic, and Nassir Navab. Ssd-6d: Making rgb-based 3d detection and 6d pose estimation great again. In *International Conference on Computer Vision*, pages 1521–1529, 2017.

[19] Chen-Hsuan Lin, Wei-Chiu Ma, Antonio Torralba, and Simon Lucey. Barf: Bundle-adjusting neural radiance fields. In *International Conference on Computer Vision*, pages 5741–5751, 2021.

[20] Lingjie Liu, Jiatao Gu, Kyaw Zaw Lin, Tat-Seng Chua, and Christian Theobalt. Neural sparse voxel fields. *Advances in Neural Information Processing Systems*, 33:15651–15663, 2020.

[21] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751*, 2019.

[22] H Christopher Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293(5828):133–135, 1981.

[23] David G Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial intelligence*, 31(3):355–395, 1987.

[24] Octave Mariotti and Hakan Bilen. Semi-supervised viewpoint estimation with geometry-aware conditional generation. In *European Conference on Computer Vision*, pages 631–647. Springer, 2020.

[25] Octave Mariotti, Oisin Mac Aodha, and Hakan Bilen. Viewnet: Unsupervised viewpoint estimation from conditional generation. In *International Conference on Computer Vision*, pages 10418–10428, 2021.

[26] Ricardo Martin-Brualla, Noha Radwan, Mehdi SM Sajjadi, Jonathan T Barron, Alexey Dosovitskiy, and Daniel Duckworth. Nerf in the wild: Neural radiance fields for unconstrained photo collections. In *Computer Vision and Pattern Recognition*, pages 7210–7219, 2021.

[27] Quan Meng, Anpei Chen, Haimin Luo, Minye Wu, Hao Su, Lan Xu, Xuming He, and Jingyi Yu. Gnerf: Gan-based neural radiance field without posed camera. In *International Conference on Computer Vision*, pages 6351–6361, 2021.

[28] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Computer Vision and Pattern Recognition*, pages 4460–4470, 2019.

[29] 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 *European conference on computer vision*, pages 405–421. Springer, 2020.

[30] 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 *Computer Vision and Pattern Recognition*, 2020.

[31] Thu Nguyen-Phuoc, Chuan Li, Lucas Theis, Christian Richardt, and Yong-Liang Yang. Hologan: Unsupervised learning of 3d representations from natural images. In *International Conference on Computer Vision*, pages 7588–7597, 2019.

[32] Michael Niemeyer and Andreas Geiger. Giraffe: Representing scenes as compositional generative neural feature fields. In *Computer Vision and Pattern Recognition*, pages 11453–11464, 2021.

[33] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Computer Vision and Pattern Recognition*, pages 165–174, 2019.

[34] Keunhong Park, Utkarsh Sinha, Jonathan T Barron, Sofien Bouaziz, Dan B Goldman, Steven M Seitz, and Ricardo Martin-Brualla. Nerfies: Deformable neural radiance fields. In *International Conference on Computer Vision*, pages 5865–5874, 2021.

[35] Mahdi Rad and Vincent Lepetit. Bb8: A scalable, accurate, robust to partial occlusion method for predicting the 3d poses of challenging objects without using depth. In *International Conference on Computer Vision*, pages 3828–3836, 2017.

[36] Johannes L Schonberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Computer Vision and Pattern Recognition*, pages 4104–4113, 2016.

[37] Katja Schwarz, Yiyi Liao, Michael Niemeyer, and Andreas Geiger. Graf: Generative radiance fields for 3d-aware image synthesis. *Advances in Neural Information Processing Systems*, 33:20154–20166, 2020.

[38] Nima Sedaghat and Thomas Brox. Unsupervised generation of a viewpoint annotated car dataset from videos. In *International Conference on Computer Vision*, pages 1314–1322, 2015.

[39] Vincent Sitzmann, Justus Thies, Felix Heide, Matthias Nießner, Gordon Wetzstein, and Michael Zollhofer. Deepvoxels: Learning persistent 3d feature embeddings. In *Computer Vision and Pattern Recognition*, 2019.

[40] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019.

[41] Vincent Sitzmann, Semon Rezchikov, Bill Freeman, Josh Tenenbaum, and Fredo Durand. Light field networks: Neural scene representations with single-evaluation rendering. *Advances in Neural Information Processing Systems*, 34, 2021.

[42] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

[43] Shubham Tulsiani and Jitendra Malik. Viewpoints and keypoints. In *Computer Vision and Pattern Recognition*, pages 1510–1519, 2015.

[44] Shubham Tulsiani, Tinghui Zhou, Alexei A Efros, and Jitendra Malik. Multi-view supervision for single-view reconstruction via differentiable ray consistency. In *Computer Vision and Pattern Recognition*, pages 2626–2634, 2017.

[45] Shubham Tulsiani, Alexei A Efros, and Jitendra Malik. Multi-view consistency as supervisory signal for learning shape and pose prediction. In *Computer Vision and Pattern Recognition*, pages 2897–2905, 2018.

[46] Angtian Wang, Shenxiao Mei, Alan L Yuille, and Adam Kortylewski. Neural view synthesis and matching for semi-supervised few-shot learning of 3d pose. *Advances in Neural Information Processing Systems*, 34, 2021.

[47] Zirui Wang, Shangzhe Wu, Weidi Xie, Min Chen, and Victor Adrian Prisacariu. Nerf–: Neural radiance fields without known camera parameters. *arXiv preprint arXiv:2102.07064*, 2021.

[48] Yang Xiao and Renaud Marlet. Few-shot object detection and viewpoint estimation for objects in the wild. In *European conference on computer vision*, pages 192–210. Springer, 2020.

[49] Xinchen Yan, Jimei Yang, Ersin Yumer, Yijie Guo, and Honglak Lee. Perspective transformer nets: Learning single-view 3d object reconstruction without 3d supervision. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1696–1704, 2016.

[50] Lin Yen-Chen, Pete Florence, Jonathan T Barron, Alberto Rodriguez, Phillip Isola, and Tsung-Yi Lin. iNeRF: Inverting neural radiance fields for pose estimation. In *IROS*, 2021.

[51] Alex Yu, Ruilong Li, Matthew Tancik, Hao Li, Ren Ng, and Angjoo Kanazawa. Plenoctrees for real-time rendering of neural radiance fields. In *International Conference on Computer Vision*, pages 5752–5761, 2021.

[52] Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Computer Vision and Pattern Recognition*, pages 4578–4587, 2021.

[53] Alex Yu, Sara Fridovich-Keil, Matthew Tancik, Qinhong Chen, Benjamin Recht, and Angjoo Kanazawa. Plenoxels: Radiance fields without neural networks. In *Computer Vision and Pattern Recognition*, 2022.

[54] Kai Zhang, Gernot Riegler, Noah Snavely, and Vladlen Koltun. Nerf++: Analyzing and improving neural radiance fields. *arXiv preprint arXiv:2010.07492*, 2020.

[55] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Computer Vision and Pattern Recognition*, pages 5745–5753, 2019.