ADen: Adaptive Density Representations for Sparse-view Camera Pose Estimation

Hao Tang, Weiyao Wang, Pierre Gleize, and Matt Feiszli

FAIR at Meta
{haotang, weiyaowang, gleize, mdf}@meta.com

Abstract. Recovering camera poses from a set of images is a foundational task in 3D computer vision, which powers key applications such as 3D scene/object reconstructions. Classic methods often depend on feature correspondence, such as keypoints, which require the input images to have large overlap and small viewpoint changes. Such requirements present considerable challenges in scenarios with sparse views. Recent data-driven approaches aim to directly output camera poses, either through regressing the 6DoF camera poses or formulating rotation as a probability distribution. However, each approach has its limitations. On one hand, directly regressing the camera poses can be ill-posed, since it assumes a single mode, which is not true under symmetry and leads to sub-optimal solutions. On the other hand, probabilistic approaches are capable of modeling the symmetry ambiguity, yet they sample the entire space of rotation uniformly by brute-force. This leads to an inevitable trade-off between high sample density, which improves model precision, and sample efficiency that determines the runtime. In this paper, we propose ADen to unify the two frameworks by employing a generator and a discriminator: the generator is trained to output multiple hypotheses of 6DoF camera pose to represent a distribution and handle multi-mode ambiguity, and the discriminator is trained to identify the hypothesis that best explains the data. This allows ADen to combine the best of both worlds, achieving substantially higher precision as well as lower runtime than previous methods in empirical evaluations.

1 Introduction

Understanding 3D structure from 2D image observations of objects or scenes is an important task in computer vision. Recent advances in Neural Radiance Field (NeRF) [26] and Gaussian Splatting [20] enable high quality 3D reconstruction and novel view synthesis, initially from densely posed images. At the core of these methods, Structure-from-Motion (SfM) plays an important role to extract camera poses from the input images.

Motivated by key real-world applications such as online marketplaces and casual captures by everyday users [43], there is also growing interest in bringing these methods to sparse-view images [28, 30, 42], where only a handful of images (*e.g.* 3 to 5) are available, each covers a different viewpoint. Similar to their

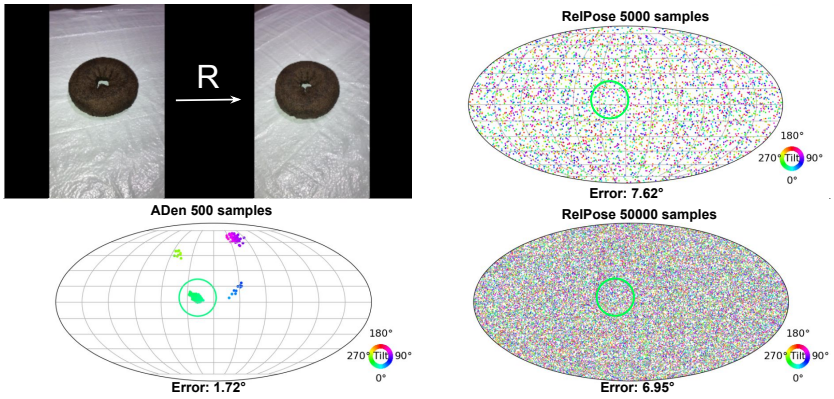


Fig. 1: Ambiguity in wide baseline images. Implicit-PDF/RelPose models rotation as a probability distribution using an energy-based method, which requires evaluating densely sampled rotation hypotheses. To achieve high accuracy, RelPose requires assessing 500k rotations for each image pair, incurring significant computational costs. In contrast, ADen outputs 500 high accuracy hypotheses directly, avoiding the constraints imposed by grid resolution. Filled circles are samples while unfilled circles are the ground truth relative rotation.

dense-view counterparts, these methods often assume known input camera poses; yet in practice, the default geometric-based SfM pipeline may fail due to minimal view overlap in the context of sparse-view images. This highlights the need for sparse pose estimations and inspires a new stream of research taking data-driven approach that learns to predict poses from large-scale object-centric dataset with superior performance than geometric-based approach [22, 27, 35, 41, 43].

The data-driven approaches for recovering relative camera poses from sparse-view images can be broadly classified into two categories. One approach is to directly regress the 6DoF camera parameters rotation R and translation T [19, 35]. However, in the sparse view setting, ambiguity arises, parituclarly for objects or scenes with symmetry (Figure 1). Pose regression assumes a single mode in the data, which can result in suboptimal solutions when trained with data exhibiting a multi-mode distribution.

A different approach is to model the rotation as a probability distribution [22, 27, 43]. Implicit-PDF [27] first introduces a method to predict arbitrary, non-parametric probability distributions over the rotation manifold to address the symmetry issue. It densely samples rotation hypotheses from SO(3), and predicts the probability for each given image features. It naturally accounts for the uncertainty in the symmetric case, allowing the model to output multiple modes, thus improving accuracy over pose regression.

While powerful, the primary drawback of this brute-force energy-based approach lies in the requirement to densely sample from the entire parameter space. To achieve high accuracy, a dense grid must be sampled from the parameter space. For example, RelPose [43] needs to sample 500,000 rotation matrices at inference time to generate a dense enough grid for good accuracy, in particu-

lar at lower error thresholds. However, evaluating 500,000 rotation hypotheses for one pair of images is computationally expensive. Furthermore, this method suffers from the curse of dimensionality: it is only practical in low dimensions (*e.g.* SO(3) is three dimensional) and becomes prohibitively expensive in higher dimensions. Moving from 3 to 6 dimensions to jointly model rotation and translation [27] without reducing sampling granularity implies a 250-billion-sized grid, yet reducing the sampling granularity implies inferior results.

How do we benefit from both approaches? Our key observation is that using uniform grid to represent pose distributions is inefficient. In real-world, the distribution of poses is highly skewed, with a few isolated modes dominating the distribution. In other words, the real-world distribution of poses sit in between the regression (one mode) [19,35] and the uniform distribution [22,27,43]. Based on this observation, we focus on a generator-discriminator framework: given 2 or more images, the generator learns to produce samples from the conditional distribution of relative poses, while the discriminator ranks them. We find that this requires only a few hundred samples to cover all possible modes of the distribution. The adaptive nature also eliminates any fundamental lower bound on accuracy imposed by a grid resolution (either a fixed spacing for fixed grids, or expected spacing for random grids).

We name this work ADen; as shown in Figure 1, ADen only needs 500 samples to outperform methods that sample 500K locations at inference time; ADen is not constrained by any grid resolution and can output samples arbitrarily close to the true mode. The generated samples clearly learn to follow the multi-modal distribution, capturing the uncertainty of poses. Importantly, by eliminating the need for dense sampling from the parameter space, ADen is not limited to model rotation alone; the generator easily outputs joint rotation and translation [$\mathbf{R}$, $\mathbf{t}$] pairs simply by predicting particles in the product space, *without* increasing their number.

In summary, our contributions are as follows.

- We propose ADen, a method for learning and sampling from the conditional distribution of relative pose from images using an efficient, adaptive generator-discriminator framework.
- ADen extends naturally to high-dimensional spaces without requiring exhaustive sampling of the entire space; it adapts to the complexity of the distribution and not the ambient space.
- Experiments show ADen outperforms SoTA methods by a large margin, especially at low error thresholds. Moreover, ADen runs much faster than previous methods, achieving real-time inference speed.

2 Related work

2.1 Structure-from-Motion (SfM)

SfM aims at recovering 3D geometry and camera poses from multi-view images set. This classic problems has been extensively studied in the past [15,29], which

typically includes computing image features (typically key points) [3, 23], establishing feature correspondence [24], computing camera poses using five-point or eight-point algorithms [15, 16, 21] with RANSAC [11], and verifying epipolar geometry with bundle adjustments [37]. Each component in this pipeline highly depends their previous steps and needs careful tuning to be robust to scale to hundreds or thousands of images [12,32]. Among these algorithms, COLMAP [34] is an open-source implementation of this entire pipeline and has been widely used by the community.

Recent progress in deep neural nets and large-scale image datasets, various methods have been proposed to improve the performance of different components of the SfM pipeline with better feature descriptors [8,10,13,31], improved feature matching [33,36,38]. Although SfM works well with multiple views, it often fails in the wide baseline setting , due to insufficient overlap between images, occlusion and failure to establish correspondence.

2.2 Data-driven pose estimation

The key step in the above-mentioned SfM pipeline is to establish correspondence between images. However, given only sparse views or wide baselines, it struggles to find matches [7]. As the result of advances in network architecture and availability of large-scale dataset with posed images [9, 30], learning-based methods has been proposed recently to directly estimate camera poses between images using a top-down approach. Unlike previous bottom-up approach, PoseNet [19] uses a network to directly regress the 6DoF camera parameters from a single RGB image. Implicit-PDF [27] is the first to propose a new framework for single image pose estimation which represents the rotation distribution implicitly, with a neural network to assign probability given input images and a candidate pose and demonstrated strong performance in handling symmetry. RelPose [43] finds the widespread presence of symmetry in real-world dataset under sparse-view setting and extends the Implicit-PDF framework to estimate camera rotations based on intial pairwise prediction. RelPose++ [22] further extends RelPose by adding a transformer to process multi-view images jointly and also report camera translation by defining a new camera coordinate system. SparsePose [35] proposes to learn an iterative refinement step to regress initial pose estimation. PoseDiff [41] proposes to formulate the SfM problem as a probabilistic diffusion framework which mainly mirrors the iterative procedure of bundle adjustment.

Similar to PoseNet and SparsePose, our approach involves the direct regression of poses to the ground truth. However, we distinguish our method by employing a generator that produces multiple pose hypotheses. Our strategy differs from the conventional approach of regressing all predictions to a single ground truth. Instead, we utilize losses designed to guide the mode towards the ground truth, leveraging the diversity provided by the generator. This approach enables the model to explore various modes in the dataset without being constrained to learn a single mode. Unlike Implicit-PDF and RelPose which require evaluating hundreds of thousands of pose hypotheses, ADen only generates a few hundred

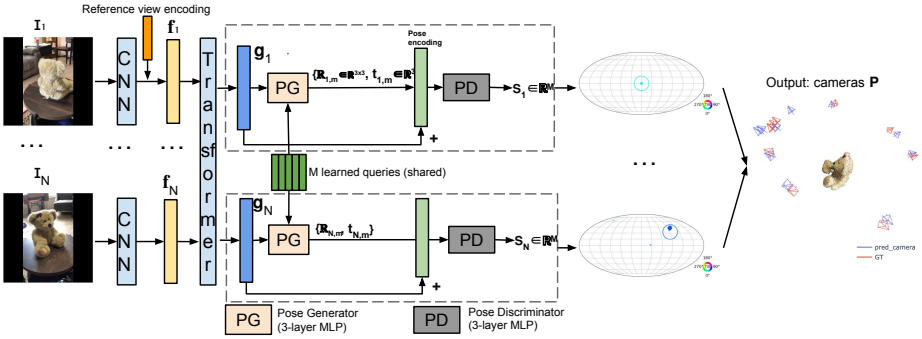


Fig. 2: ADen overview. ADen is a novel method for recovering camera poses from sparse-view RGB images. ADen starts by extracting per-image features using the ResNet backbone, then uses a transformer to fuse features from all images and propagate information globally. ADen predicts a non-uniform distribution over camera poses for each image by first applying a pose generator head on fused features to produce a support set of M camera poses, then using a pose discriminator with fused features to predict probability on each generated pose.

hypotheses and uses a discriminator to pick the best one thus can achieve real-time inference.

2.3 Generative learning and contrastive learning

Modeling pose distribution using generative models has been successful recently [5, 41]. GANs [14] are generative models trained adversarially. Similar to GANs, we do use a generator and a discriminator, but with a shared backbone (as opposed to having two fully independent models). Unlike GANs, our objective is not adversarial. The discriminator is trained to score various samples, not competing with the generator. The discriminator is trained using the contrastive loss [6] to model the pose distribution, by rewarding the positive samples and penalizing negative samples. Unlike previous methods [6, 17], we do not aim to learn visual representations but to score samples from a distribution.

3 Method

Given a set of N sparse-view images of an object $\{\mathbf{I}_1, ..., \mathbf{I}_N\}$, ADen aims to recover the camera extrinsics $\{[\mathbf{R}_1, \mathbf{t}_1], ..., [\mathbf{R}_N, \mathbf{t}_N]\}$, where $\mathbf{R}_i \in \mathrm{SO}(3)$ is the rotation and $\mathbf{t}_i \in \mathbb{R}^3$ is the translation for image $\mathbf{I}_i$. The model is trained to first draw samples $[\mathbf{R}_{n,k}, \mathbf{t}_{n,k}]$, from the distribution of all possible poses given the input images $p(\mathbf{R}_n, \mathbf{t}_n | \mathbf{I}_1, ..., \mathbf{I}_N)$, where n is the index for the image and k is the index for the sample. And subsequently, it scores these samples and selects the most probable one as its prediction $[\hat{\mathbf{R}}_n, \hat{\mathbf{t}}_n]$ for each image. Our proposed method consists of a shared feature backbone, and two heads: pose generator and pose discriminator (Figure 2). We use the camera pose for the first frame as canonical reference to infer camera poses of other frames.

3.1 Multi-view feature extraction

ADen begins by extracting per-image features using ResNet [18], denoted as $\mathbf{F}_i = f(\mathbf{I}_i) \in \mathbb{R}^d$, where d is the feature space dimension. To reason relative poses across multiple input views, ADen concatenates the features and uses transformers [40] t to fuse them: $(\mathbf{G_1}, ..., \mathbf{G}_N) = t(\mathbf{F_1}, ..., \mathbf{F_N})$. An additional reference image encoding is added to $\mathbf{F_1}$ to designate it as representing the canonical pose. The fused features $\mathbf{G}$ are then fed into a pose generator and a pose discriminator to predict relative poses.

3.2 Pose generator

The pose generator takes inputs from the fused features and predicts multiple pose hypotheses for each input image. These hypotheses aim at capturing a distribution over poses under natural ambiguities such as rotation symmetry. However, the training data only contains a single groundtruth pose. Training all pose proposals to match the ground truth will result in a mode collapse: the generator will output only one pose and loses its ability to output a pose distribution to model these natural ambiguities. This has also been verified through empirical experiments (Table 5). To avoid such mode collapse, we allow the model to explore multiple modes by only regressing one pose candidate closest to the ground truth; no loss is computed to penalize other poses predictions.

We use an MLP to generate M pose hypotheses $Q_{i,m} \in \mathbb{R}^7$ which contains 4 elements for the quaternion and 3 for the translation. The choice to represent the rotation matrix with a quaternion is due to the ease of normalizing the model's output in quaternion space. We then transform the quaternion representation of the rotation component of $Q_{i,m}$ back to 3×3 rotation matrix. This yields $P_{i,m} \in \mathbb{R}^{12}$. The pose generator takes inputs from the fused features G and M randomly initialized learnable queries $\{e_{i,m}\}_{m=1}^{M} \in \mathbb{R}^{256}$. An MLP is used to map the queries to the feature space. Note that these randomly initialized queries specialize in distinct partition of the pose space, similar to learnable queries in object detections [4], enabling the model to fully cover the space.

$$P_{i,m} = \mathbf{MLP}(\mathbf{G}_i + \mathbf{MLP}(e_{i,m}))$$

3.3 Pose discriminator

Given the generated pose hypotheses from the pose generator head, the pose discriminator evaluates the probability of each hypothesis being correct. At inference, the hypothesis with the highest probability is used as the output.

The pose discriminator first embeds the generated hypotheses to the same dimension as feature dimension and then uses an MLP to predict logits $x_{i,m}$.

$$x_{i,m} = \mathbf{MLP}(\mathbf{G}_i + \mathbf{MLP}(P_{i,m}))$$

For any pose and logit pair $(\mathbf{P}, x)$ we then have probability:

$$\hat{p}(\mathbf{P}|\mathbf{I}_1, ..., \mathbf{I}_N; i) = \frac{e^x}{e^{x_i^*} + \sum_m e^{x_{i,m}}} \tag{1}$$

where $(\mathbf{P}_i^*, x_i^*)$ is the ground truth pose during training. At inference, we omit this term since ground truth is not available.

3.4 Model training details

Pose generator To prevent mode collapse, which occurs when all proposals are trained to match the ground truth as discussed in Section 3.2, we only regress the one pose proposal that is closest to the ground truth pose: $P_i^* = [\mathbf{R}_i^*, \mathbf{t}_i^*]$ and no loss is computed to penalize other poses proposals. We define the "closest" camera pose as the one with the smallest geodesic distance:

$$\hat{P}_i = [\hat{\mathbf{R}}_i, \hat{\mathbf{t}}_i] = \arg\min_{P_{i,m}} d(P_{i,m}, P_i^*), m \in [1, \mathrm{M}]$$

The loss function then is the geodesic distance of predicted rotation to ground truth and the L2 loss of the predicted translation.

$$\mathcal{L}_g = \sum_i^N \arccos(\mathrm{Tr}(\mathbf{R}_i^* \hat{\mathbf{R}}_i)) + \sum_i^N ||\mathbf{t}_i^* - \hat{\mathbf{t}}_i|| \tag{2}$$

Pose discriminator During training, we also add the ground truth to the network so $x_i^* = \mathbf{MLP}(\mathbf{g}_i, \mathbf{MLP}(p_i^*))$. The loss function for the discriminator then is the contrastive negative log-likelihood loss to differentiate ground truth from the rest.

$$\mathcal{L}_d = -\sum_i^N \log \hat{p}(P_i^* | \mathbf{I}_1, ..., \mathbf{I}_N, i) \tag{3}$$

with $\hat{p}$ given by (1). The final loss is the sum of pose generator loss and pose discriminator loss $\mathcal{L} = \mathcal{L}_g + \mathcal{L}_d$.

Joint training Training instability is well-studied in generative adversarial models (GAN) [14]. In particular, as the generator improves its ability to generate accurate camera poses, the discriminator's task becomes more difficult, and its gradients may effectively become noise; this can negatively impact model performance. We observed similar behavior during our training. There are many techniques to address this for GANs, but in our setting a very simple approach works well. Specifically, we add random Gaussian noise into the query embedding $\mathbf{e_i}$ to generate negative examples for the discriminator. In our ablation study, we explore the impact of this noise on the final performance of the model Section 4.3.

In our setup, the pose generator produces samples from the current estimate for the pose distribution. While we design our system as a generator-discriminator pair with a contrastive loss, we observe that if the set of samples is

sufficient to approximate the distribution, then this is in fact simply a maximum-likelihood objective. This suggests classical models like kernel density estimates and mixture models may also be effective here, and indeed this is the case (see Supplemental), although our contrastive method provides stronger results.

3.5 Implementation details

Since the pose coordinate is ill-defined in an absolute coordinate system, we set the first frame as the canonical frame and set its rotation to identity. A reference frame encoding is added to the first frame to inform the model. For translation, we use a similar object-centric coordinate system as [22]. We normalize the translation of the canonical frame to be [0, 0, 1] and normalize all translations according to the distance from the canonical camera to the object center. This can be achieved by computing the center of mass of the object point cloud.

We use the same Anti-alias Res50 backbone as [43] and a transformer with six layers of multi-head attention. The ResNet backbone is pretrained on ImageNet while the transformer is trained from scratch. We use Adam as the optimizer with learning rate set to $1e^{-4}$ and the model is trained for 2000 epochs. During training, we randomly sampled different numbers of images ranging from 2-10 for each batch.

4 Experiments

4.1 Experiment setup

Dataset We use CO3D V2 [30] for training and testing ADen, which encompasses 51 object-centric sequences.The ground truth pose for each sequence is determined by employing [34]. In line with [41,43], ADen is trained on 41 object categories and subsequently evaluated on both the test set of these 41 seen categories and a separate test set of 10 unseen categories. We follow the same test set frame sampling method of [22] that resamples N images from a sequence 5 times and reports the average as the final accuracy.

To further evaluate the generalization ability of method, we test ADen on two additional datasets in zero-shot setting (without any finetuning): Objectron [1] and Niantic Map Free Relocalization (NMFR) [2]. Objectron consists of short, object-centric video clips captured for AR use cases; NMFR captures diverse outdoor scenes and structures including sculptures, murals and foundations. Following [22], all methods were trained using CO3D [30] and tested zero-shot (no finetuning). For the Objectron dataset, we used the test set of four classes (Camera, Chair, Cup, Shoe), following the same protocol as Relpose++ [22]. For the Niantic Map Free dataset, we used the validation set, which contains 65 scans.

Evaluation metrics For evaluating the rotation accuracy, we measure the predicted pairwise relative rotation against ground truth rotations and report the proportion of rotation errors that are less than 15 degrees. To show accuracy at a tighter error threshold, we also report rotation errors that are less than 5 and

10 degrees. For evaluating the translation, we follow [22] to first apply an optimal similarity transform to align the predicted centers with ground truth [39]. We then report the accuracy as the proportion of translation errors that are less than 10% of the scene scale. The scene scale is determined by the distance from the centroid of all ground truth cameras to the furthest camera.

4.2 Comparing with SoTA

Baselines We compare the proposed method with the following state-of-the-art (SoTA) classic correspondence-based and data-driven approaches:

COLMAP (SP + SG) [34]. This represents the SoTA SfM pipeline using the open-source COLMAP implementation, with SuperPoint as key-point features [8] and SuperGlue for key-point matching [33].

SparsePose [35] SparesePose directly regresses poses and uses an iterative refinement strategy to further refine the initial poses.

PoseDiff [41]. PoseDiff formulates pose estimation inside a probabilistic diffusion framework, which mirrors the iterative procedure of bundle adjustment.

RelPose and RelPose++ [22, 43]. RelPose is the first work that models rotation as a probability distribution and RelPose++ is an extension that adds a transformer module and also reports translation error. RelPose++ represents the SoTA learning-based method on the Co3D dataset.

Pose Regression [22]. Pose Regression directly regresses poses given sparse view images. We use the one reported in [22].

As shown in Table 1 and Table 3, ADen achieves SoTA performance on both rotation and translation accuracy compared to previous methods. ADen consistently outperforms other baselines with different numbers of images as input. This performance gain is more prominent at tighter rotation accuracy threshold Table 2. Compared with Pose Regression and SparsePose, our method allows the model to explore different modes during training to better learn the underlying ambiguous multi-modal distribution, so it performs significantly better. In contrast to RelPose/RelPose++, which also predicts a probabilistic distribution for poses, our approach is not constrained by the resolution of samples from the SO(3) space and can generate samples that closely match the ground truth camera pose. This improvement is more apparent at tighter rotation error thresholds (Table 2), underscoring the precision of the samples generated by our pose generator. Additionally, our method achieves SoTA results in camera translation errors, primarily due to enhanced rotation accuracy. On the two additional datasets, ADen demonstrates strong generalization, significantly surpassing prior methods and achieving SOTA performance Table 4a and Table 4b.

4.3 Ablation

For ablation study, we by default use 5 input images to train and test ADen on the seen categories.

How to train the generator.

	# of Images	2	3	4	5	6	7	8
Seen	COLMAP(SP+SG)	30.7	28.4	26.5	26.8	27.0	28.1	30.6
	RelPose	56.0	56.5	57.0	57.2	57.2	57.3	57.2
	Pose Regression	49.1	50.7	53.0	54.6	55.7	56.1	56.5
	RelPose++	81.8	82.8	84.1	84.7	84.9	85.3	85.5
	PoseDiff	76.0	76.7	77.2	77.7	78.3	78.5	78.5
	Ours	**84.3**	**85.5**	**86.0**	**86.5**	**87.0**	**87.1**	**87.3**
Unseen	COLMAP(SP+SG)	34.5	31.8	31.0	31.7	32.7	35.0	38.5
	RelPose	48.6	47.5	48.1	48.3	48.4	48.4	48.3
	Pose Regression	42.7	43.8	46.3	47.7	48.4	48.9	48.9
	SparsePose	-	65.0	68.0	70.0	67.0	72.0	72.0
	RelPose++	69.8	71.1	71.9	72.8	73.8	74.4	74.9
	PoseDiff	60.3	64.0	64.9	65.6	66.4	67.1	67.5
	Ours	**78.6**	**79.0**	**80.1**	**80.9**	**81.1**	**81.4**	**82.1**

Table 1: Pairwise relative rotation accuracy @ 15. We measure the relative angular errors between relative predicted and ground truth rotation, and report the accuracy as the proportion of errors less than 15 degrees. ADen consistently outperforms SoTA methods using different number of images as input.

Acc@5	# of Images	2	3	4	5	6	7	8
Seen	RelPose++	42.1	43.6	44.4	44.7	45.0	45.1	45.4
	PoseDiff	49.8	51.0	50.6	51.7	52.5	53.0	54.4
	Ours	**53.5**	**55.0**	**55.8**	**56.0**	**56.7**	**56.6**	**56.9**
Unseen	RelPose++	30.7	31.9	32.8	33.5	33.9	34.0	33.9
	PoseDiff	38.0	34.9	37.2	39.5	41.2	42.4	42.6
	Ours	**45.3**	**46.4**	**47.5**	**48.2**	**48.2**	**48.5**	**48.7**
Acc@10	**# of Images**	2	3	4	5	6	7	8
Seen	RelPose++	70.5	72.3	73.6	74.2	74.7	75.2	75.2
	PoseDiff	68.4	70.1	70.2	71.1	72.2	72.3	72.4
	Ours	**77.7**	**79.0**	**79.4**	**79.9**	**80.4**	**80.5**	**80.7**
Unseen	RelPose++	57.7	58.7	60.4	61.2	61.8	62.0	62.5
	PoseDiff	50.6	55.7	57.2	57.9	58.3	60.0	60.8
	Ours	**69.2**	**70.6**	**72.0**	**72.9**	**73.5**	**73.7**	**74.1**

Table 2: Pairwise relative rotation accuray @ 5° and 10°. We report Acc@5 and Acc@10 for ADen, PoseDiff and RelPose++. ADen is not constrained by the accuracy of the sampled grid on SO3, thereby achieving greater gains at tighter accuracy thresholds.

	# of Images	3	4	5	6	7	8
Seen	COLMAP(SP+SG)	35.8	26.1	21.6	18.9	18.3	19.2
	Pose Regression	87.6	81.2	77.6	75.8	74.5	73.6
	RelPose++	92.3	89.1	87.5	86.4	85.9	85.5
	PoseDiff	78.3	78.1	79.1	79.4	79.4	79.4
	Ours	**92.5**	**89.4**	**88.1**	**86.7**	**86.4**	**85.9**
Unseen	COLMAP(SP+SG)	37.9	29.3	24.7	23.1	23.5	25.3
	Pose Regression	82.8	74.0	70.0	67.8	65.8	65.3
	RelPose++	82.5	75.6	71.9	69.9	68.5	67.5
	PoseDiff	61.8	61.9	63.1	63.1	62.7	63.3
	Ours	**85.4**	**78.9**	**75.7**	**73.0**	**71.5**	**70.7**

Table 3: Translation accuracy @ 0.2 We measure the accuracy as the proportion of predicted camera translations that are within 20% of the scene scale of each sequence.

	Rotation			Cam. Cen.		
# of Images	3	5	8	3	5	8
MediaPipe [25]	52.3	52.8	52.7	74.5	59.1	49.9
PoseDiff [41]	69.2	68.0	70.0	87.2	73.8	67.2
RelPose++ [22]	75.8	76.6	77.0	91.6	83.9	77.6
Ours	**86.0**	**85.8**	**85.7**	**93.1**	**85.0**	**80.0**

(a) Objectron [1]

	Rotation			Cam. Cen.		
# of Images	3	5	8	3	5	8
PoseDiff [41]	39.5	43.2	42.3	70.4	48.9	42.8
RelPose++ [22]	45.2	44.8	46.2	73.9	50.4	43.2
Ours	**55.3**	**52.6**	**52.8**	**74.5**	**51.5**	**44.4**

(b) Niantic Map-Free Relocalization [2]

Table 4: Zero-shot evaluation on two additional datasets on rotation (@ 15°) and camera center (@ 0.2) Accuracy.

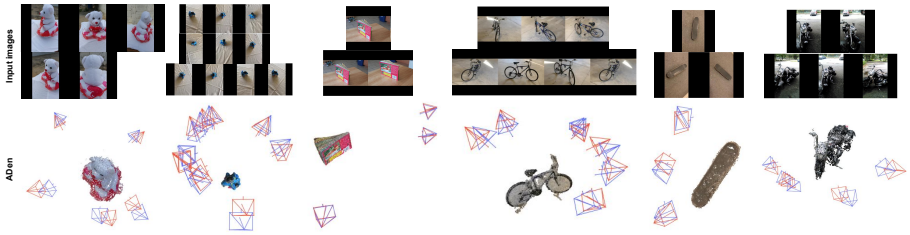


Fig. 3: Camera pose prediction of ADen on CO3D examples.

Training generator directly controls the quality of the generated sample distribution and thus is critical to the performance of the model. By default we only apply $\mathcal{L}_g$ to the camera pose that is closest to the ground truth. This allows the model to move samples closer to the ground truth, while simultaneously grants the flexibility for other samples to explore different modes without penalizing them. We compare training the generator by pulling 2% (10), 10% (50), and all samples towards the ground truth. As shown in Table 5, it indicates that applying regression loss to all samples may hinder the model's ability to explore diverse modes during training, leading to poorer performance. Furthermore, the generator's sensitivity to the exact number of samples used for training is low, provided there is sufficient leeway for other samples to explore.

How to train the discriminator.

Training the discriminator is critical to learn good features that are informative in differentiating different camera poses. Our observations indicate that

Setting		Acc@5	Acc@10	Acc@15
How to train generator				
	Top 1	55.7	79.9	86.4
	Top 10	54.5	79.8	86.8
	Top 50	56.0	80.0	87.0
	All	48.2	70.5	77.3
How to train discriminator				
(1)	500 randomly sampled	38.2	72.0	83.4
	5k randomly sampled	50.0	78.9	87.6
	50k randomly sampled	54.2	80.2	87.4
(2)	500 generated w/o noise	42.5	67.3	77.0
	2x training	48.0	71.7	79.6
(3)	500 generated w/ noise	55.7	79.9	86.4

Table 5: Ablation on training the generator and discriminator. Training the generator with regression loss applied only to the nearest few samples allows the model to explore alternative modes, leading to improved outcomes. Additionally, training the discriminator with generated noisy camera poses strikes an effective balance between accuracy and training efficiency.

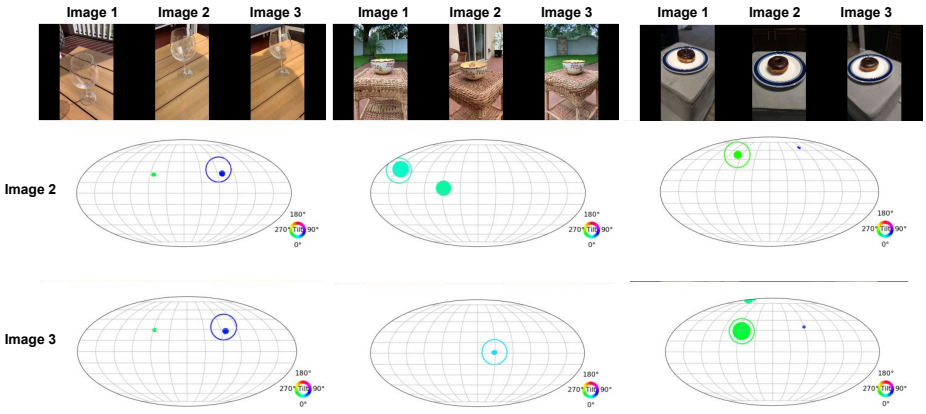
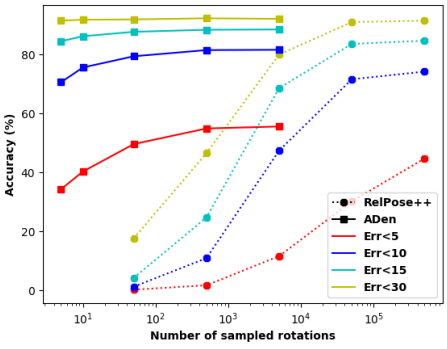


Fig. 4: Relative rotation prediction. We visualize the relative rotations predicted by ADen on ambiguous cases. The circle size of each filled circle represents the probability assigned by the discriminator. The unfilled larger circle is the ground truth.

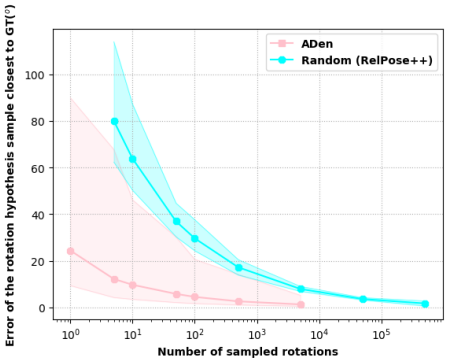
the training dynamics of the discriminator, when utilizing samples generated by the pose generator, resemble those observed in generative adversarial networks (GANs) [14], which may lead to instability in training. To mitigate this, we introduce random noise to the generated queries, enhancing the discriminator's training stability. We evaluate different strategies for training the discriminator to ascertain the impact of this added noise: (1) use randomly sampled camera poses; (2) use the generated samples from the pose generator; (3) use a noisy version of the generated samples from the pose generator. For 1) we explore the effects of using 500, 5,000, and 50,000 randomly sampled camera poses for training. For (3), we add Gaussian noise to the learnt queries embedding to produce noisy camera poses for the discriminator $\mathcal{N}(\mathbf{0}, \mathbf{3}) + \mathbf{e_i}$.

The results are shown in Table 5. First, using randomly generated camera poses shows that the denser the sampling, the more improved the outcomes. This improvement can be attributed to denser sampling increasing the probability of the samples being closer to the true mode, which in turn provides hard negatives that facilitate the learning process of the discriminator. Second, training directly with the camera poses generated by the model poses a challenge; as the generator becomes more adept at producing accurate poses, the task of distinguishing between them becomes increasingly difficult for the discriminator, potentially causing training instability. This could potentially lead to instability in training. An attempt to mitigate this by doubling the training duration resulted in a 5% increase in performance on the Acc@5 metric. However, this still falls short when compared to other training strategies, underscoring the inherent difficulties in training the discriminator. Lastly, introducing a noisy version of the generated samples to the discriminator's training regimen led to an approximate 2-3% improvement across most accuracy thresholds, achieving results comparable to those obtained by training with a large pool of randomly sampled poses (50k in the first setting). This suggests that adding noise to the generated samples

can effectively enhance discriminator performance, paralleling the benefits of extensive random sampling.



(a) Rotation accuracy with different number of sampled camera poses (N=5). The proposed method, which directly generates samples, achieves high accuracy with just a few hundred samples. In contrast, RelPose++ relies on random rotations sampled uniformly from SO(3), necessitating a larger sample size to adequately cover the parameter space. This difference becomes more pronounced when measuring accuracy at lower error thresholds.



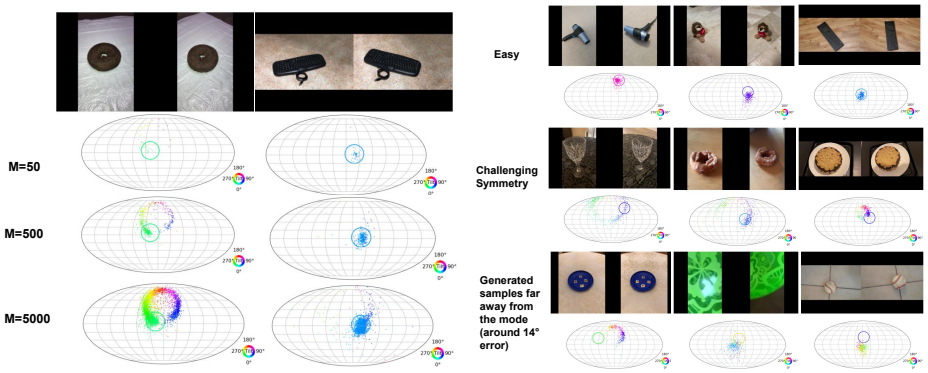
(b) Generator performance measured as the error of the rotation hypothesis that is closest to the GT. We compare our generated samples with those randomly sampled from SO(3). We assess our generator's effectiveness using the error from the closest rotation hypothesis to the ground truth, as the true pose distribution is unknown. While dense sampling from SO(3) might require over 500,000 samples to cover all rotations, our generator achieves similar accuracy with just 500 samples.

Fig. 5: Performance of the generator and discriminator as number of sampled poses.

We demonstrate the performance variations with different numbers of samples, primarily focusing on a comparison with RelPose++ because it also draws samples during inference. The key distinction lies in our method's utilization of the generator to produce samples, as opposed to RelPose++ which randomly selects samples from the SO(3) sphere. Our method is designed to circumvent generating samples in regions unlikely to contain the true mode, thereby significantly improving data efficiency. As illustrated in Figure 5a, ADen requires merely a few hundred samples to surpass the accuracy that RelPose++ achieves with 500,000 rotations. This ablation demonstrates ADen is sample efficient and its accuracy is not constraint by the resolution of the sampled grid.

Performance of the generator.

We present the following analysis to examine how the performance of the generator varies both qualitatively and quantitatively, given its critical role in overall performance. Specifically, Figure 5b illustrates the change in error for the optimal rotation hypothesis as the sample size is adjusted. Furthermore, Figure 6a provides a visual representation of the rotation hypothesis distribution for varying numbers of samples (M). Figure 6 shows several examples of the generator's outputs for images that pose greater challenges, such as those with symmetry. Notably, there is an observed increase in the variance of hypotheses for these more difficult examples. However, the generator accurately models symmetry as a distribution, such as the donut and wine glass.



(a) Visualization of generator's output with different number of samples.

(b) Generator's output in challenging (symmetric) cases.

Fig. 6: Visualization of the output from generator. Unfilled circles represent ground truth rotation. Dots represent generated rotation samples.

4.4 Inference speed

We also compared inference speeds across various methods, each evaluated using nine images as input, tested 50 times on an A6000 GPU. Since SparsePose is not open sourced, we directly quote their runtime from their paper. As can be seen from Table 6, ADen reaches an average speed of 0.05s per 9 images and that is above 20 FPS, significantly outperforming all previous methods. In summary, ADen not only surpasses state-of-the-art methods in accuracy but also demonstrates remarkable real-time speed, averaging 20 FPS with 9 input images, underscoring its efficiency and effectiveness in application.

	Time (secs)	FPS
COLMAP (SP + SG)	18	0.056
SparsePose	3.6	0.278
RelPose/RelPose++	48	0.020
PoseDiff	63.8	0.015
Ours	**0.05**	**20**

Table 6: Inference speed comparison. ADen achieves significantly faster inference speed than other methods, averaging 20 FPS with nine images as input.

5 Conclusion

In this paper, we present a new learning-based method for recovering camera poses from sparse-view RGB only images. The design of a pose generator and a pose discriminator in ADen empowers the network to navigate the ambiguity inherent in wide baseline images and generate multiple modes. Experiments demonstrates ADen achieves the SoTA performance on CO3D dataset, surpassing previous methods, particularly in accuracy at lower rotation error thresholds. Additionally, owing to its efficient generator, ADen can infer poses for nine images in real-time (20 FPS), demonstrating a significant speed improvement over all prior methods.

References

1. Ahmadyan, A., Zhang, L., Ablavatski, A., Wei, J., Grundmann, M.: Objectron: A large scale dataset of object-centric videos in the wild with pose annotations. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7822–7831 (2021)
2. Arnold, E., Wynn, J., Vicente, S., Garcia-Hernando, G., Monszpart, A., Prisacariu, V., Turmukhambetov, D., Brachmann, E.: Map-free visual relocalization: Metric pose relative to a single image. In: European Conference on Computer Vision. pp. 690–708. Springer (2022)
3. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: Computer Vision–ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part I 9. pp. 404–417. Springer (2006)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European Conference on Computer Vision. pp. 213–229. Springer (2020)
5. Chen, H., Wang, P., Wang, F., Tian, W., Xiong, L., Li, H.: Epro-pnp: Generalized end-to-end probabilistic perspective-n-points for monocular object pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2781–2790 (2022)
6. Chen, T., Kornblith, S., Norouzi, M., Hinton, G.: A simple framework for contrastive learning of visual representations. In: International conference on machine learning. pp. 1597–1607. PMLR (2020)
7. Choi, S., Zhou, Q.Y., Koltun, V.: Robust reconstruction of indoor scenes. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5556–5565 (2015)
8. DeTone, D., Malisiewicz, T., Rabinovich, A.: Superpoint: Self-supervised interest point detection and description. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 224–236 (2018)
9. Downs, L., Francis, A., Koenig, N., Kinman, B., Hickman, R., Reymann, K., McHugh, T.B., Vanhoucke, V.: Google scanned objects: A high-quality dataset of 3d scanned household items. In: 2022 International Conference on Robotics and Automation (ICRA). pp. 2553–2560. IEEE (2022)
10. Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., Sattler, T.: D2-net: A trainable cnn for joint detection and description of local features. arXiv preprint arXiv:1905.03561 (2019)
11. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM **24**(6), 381–395 (1981)
12. Furukawa, Y., Curless, B., Seitz, S.M., Szeliski, R.: Towards internet-scale multi-view stereo. In: 2010 IEEE computer society conference on computer vision and pattern recognition. pp. 1434–1441. IEEE (2010)
13. Gleize, P., Wang, W., Feiszli, M.: Silk–simple learned keypoints. arXiv preprint arXiv:2304.06194 (2023)
14. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial networks. Communications of the ACM **63**(11), 139–144 (2020)
15. Hartley, R., Zisserman, A.: Multiple view geometry in computer vision. Cambridge university press (2003)

16. Hartley, R.I.: In defense of the eight-point algorithm. IEEE Transactions on pattern analysis and machine intelligence **19**(6), 580–593 (1997)
17. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16000–16009 (2022)
18. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
19. Kendall, A., Grimes, M., Cipolla, R.: Posenet: A convolutional network for real-time 6-dof camera relocalization. In: Proceedings of the IEEE international conference on computer vision. pp. 2938–2946 (2015)
20. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics **42**(4) (July 2023), `https://repo-sam.inria.fr/fungraph/3d-gaussian-splatting/`
21. Li, H., Hartley, R.: Five-point motion estimation made easy. In: 18th International Conference on Pattern Recognition (ICPR'06). vol. 1, pp. 630–633. IEEE (2006)
22. Lin, A., Zhang, J.Y., Ramanan, D., Tulsiani, S.: Relpose++: Recovering 6d poses from sparse-view observations. arXiv preprint arXiv:2305.04926 (2023)
23. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the seventh IEEE international conference on computer vision. vol. 2, pp. 1150–1157. Ieee (1999)
24. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: IJCAI'81: 7th international joint conference on Artificial intelligence. vol. 2, pp. 674–679 (1981)
25. Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., Zhang, F., Chang, C.L., Yong, M.G., Lee, J., et al.: Mediapipe: A framework for building perception pipelines. arXiv preprint arXiv:1906.08172 (2019)
26. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. Communications of the ACM **65**(1), 99–106 (2021)
27. Murphy, K., Esteves, C., Jampani, V., Ramalingam, S., Makadia, A.: Implicit-pdf: Non-parametric representation of probability distributions on the rotation manifold. arXiv preprint arXiv:2106.05965 (2021)
28. Niemeyer, M., Barron, J.T., Mildenhall, B., Sajjadi, M.S., Geiger, A., Radwan, N.: Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5480–5490 (2022)
29. Özyeşil, O., Voroninski, V., Basri, R., Singer, A.: A survey of structure from motion*. Acta Numerica **26**, 305–364 (2017)
30. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordone, L., Labatut, P., Novotny, D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10901–10911 (2021)
31. Revaud, J., De Souza, C., Humenberger, M., Weinzaepfel, P.: R2d2: Reliable and repeatable detector and descriptor. Advances in neural information processing systems **32** (2019)
32. Sarlin, P.E., Cadena, C., Siegwart, R., Dymczyk, M.: From coarse to fine: Robust hierarchical localization at large scale. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12716–12725 (2019)

33. Sarlin, P.E., DeTone, D., Malisiewicz, T., Rabinovich, A.: Superglue: Learning feature matching with graph neural networks. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 4938–4947 (2020)
34. Schonberger, J.L., Frahm, J.M.: Structure-from-motion revisited. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4104–4113 (2016)
35. Sinha, S., Zhang, J.Y., Tagliasacchi, A., Gilitschenski, I., Lindell, D.B.: Sparsepose: Sparse-view camera pose regression and refinement. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21349–21359 (2023)
36. Sun, J., Shen, Z., Wang, Y., Bao, H., Zhou, X.: Loftr: Detector-free local feature matching with transformers. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 8922–8931 (2021)
37. Triggs, B., McLauchlan, P.F., Hartley, R.I., Fitzgibbon, A.W.: Bundle adjustment—a modern synthesis. In: Vision Algorithms: Theory and Practice: International Workshop on Vision Algorithms Corfu, Greece, September 21–22, 1999 Proceedings. pp. 298–372. Springer (2000)
38. Truong, P., Danelljan, M., Timofte, R.: Glu-net: Global-local universal network for dense flow and correspondences. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 6258–6268 (2020)
39. Umeyama, S.: Least-squares estimation of transformation parameters between two point patterns. IEEE Transactions on Pattern Analysis & Machine Intelligence **13**(04), 376–380 (1991)
40. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. In: Advances in neural information processing systems. pp. 5998–6008 (2017)
41. Wang, J., Rupprecht, C., Novotny, D.: Posediffusion: Solving pose estimation via diffusion-aided bundle adjustment. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9773–9783 (2023)
42. Yu, A., Ye, V., Tancik, M., Kanazawa, A.: pixelnerf: Neural radiance fields from one or few images. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4578–4587 (2021)
43. Zhang, J.Y., Ramanan, D., Tulsiani, S.: Relpose: Predicting probabilistic relative rotation for single objects in the wild. In: European Conference on Computer Vision. pp. 592–611. Springer (2022)