

EquiPose: Exploiting Permutation Equivariance for Relative Camera Pose Estimation

Yuzhen Liu^{1,2} Qiulei Dong^{1,2,*}

¹State Key Laboratory of Multimodal Artificial Intelligence Systems, Chinese Academy of Sciences, Institute of Automation

²School of Artificial Intelligence, University of Chinese Academy of Sciences

liuyuzhen2022@ia.ac.cn, qldong@nlpr.ia.ac.cn

Abstract

Relative camera pose estimation between two images is a fundamental task in 3D computer vision. Recently, many relative pose estimation networks have been explored for learning a mapping from two input images to their corresponding relative pose, however, the estimated relative poses by these methods do not have the intrinsic **Pose Permutation Equivariance (PPE)** property: the estimated relative pose from Image A to Image B should be the inverse of that from Image B to Image A. It means that permuting the input order of two images would cause these methods to obtain inconsistent relative poses. To address this problem, we firstly introduce the concept of PPE mapping, which indicates such a mapping that captures the intrinsic PPE property of relative poses. Then by enforcing the aforementioned PPE property, we propose a general framework for relative pose estimation, called EquiPose, which could easily accommodate various relative pose estimation networks in literature as its baseline models. We further theoretically prove that the proposed EquiPose framework could guarantee that its obtained mapping is a PPE mapping. Given a pre-trained baseline model, the proposed EquiPose framework could improve its performance even without fine-tuning, and could further boost its performance with fine-tuning. Experimental results on four public datasets demonstrate that EquiPose could significantly improve the performances of various state-of-the-art models.

1. Introduction

Estimating the relative camera pose between an image pair is one of the fundamental tasks in 3D computer vision [6, 46]. It has a broad range of applications, such as structure-from-motion (SfM) [14, 15, 42], visual localization [10, 40, 45], novel view synthesis [7, 53, 54], etc.

Existing methods for relative camera pose estimation in literature could be roughly divided into two categories: matching-based methods and end-to-end methods. The matching-based methods [13, 33] generally employ a two-stage computational strategy: Firstly, they establish the point correspondences between an input pair of images by utilizing local feature extraction and matching techniques [13, 33]. Then according to the established correspondences, they solve a two-view geometric problem via a typical 5-point algorithm [24] with robust fitting [21] for pursuing the relative pose. However, since the performances of the matching-based methods depend on the established point correspondences, their performance would significantly decrease or even fail in cases of weak/repeated textures and degenerated configurations as indicated in [20].

Unlike the above matching-based methods, many recent methods [6, 12, 19, 28, 36, 46, 57, 61] have employed deep neural networks to directly learn a mapping from the input two images to their relative pose without explicitly establishing point correspondences. Here, it has to be pointed out that ideally, the estimated relative pose by these end-to-end methods should have the following intrinsic *pose permutation equivariance* (PPE) property between an image pair:

Pose Permutation Equivariance

Let $\{I_1, I_2\}$ be a pair of images, then the relative pose P_{12} from I_1 to I_2 should be the inverse of the relative pose P_{21} from I_2 to I_1 :

$$P_{21} = P_{12}^{-1}. \quad (1)$$

By writing P_{12} and P_{21} in the matrix form:

$$P_{12} = \begin{bmatrix} R_{12} & t_{12} \\ 0 & 1 \end{bmatrix}, \quad P_{21} = \begin{bmatrix} R_{21} & t_{21} \\ 0 & 1 \end{bmatrix}, \quad (2)$$

where R_{12} and t_{12} are the relative rotation and translation from I_1 to I_2 , and R_{21} and t_{21} are the relative rotation and

*Corresponding author

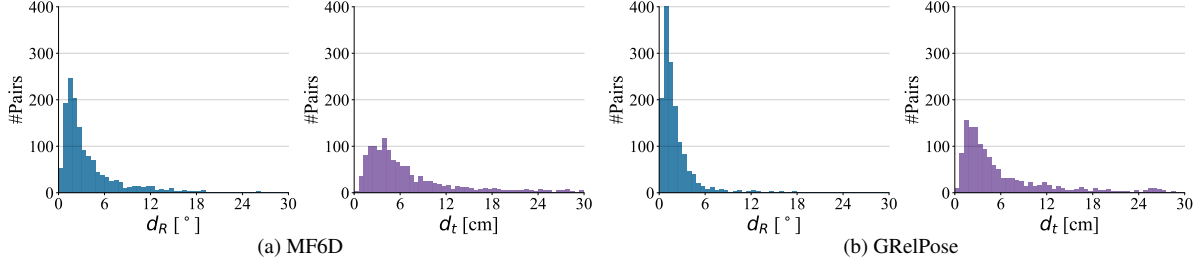


Figure 1. Distribution of the d_R and d_t obtained by existing models MF6D [1] and GRRelPose [28].

translation from I_2 to I_1 , Eq. (1) could be reformulated as:

$$R_{21} = R_{12}^T, \quad t_{21} = -R_{12}^T t_{12}. \quad (3)$$

However, the PPE property is generally overlooked when designing model architectures in existing end-to-end methods [1, 28]. We validate this by conducting the following experiment: we trained two typical end-to-end models (MF6D [1] and GRRelPose [28]) on the training set of ScanNet [11]. Then, we adopt the test set of ScanNet [48, 50] which contains 1500 image pairs, and feed them with two different orders to the above models respectively, leading to totally 1500 pairs of the predicted relative poses by each model, where each pair consists of the estimated pose P_{12} from I_1 to I_2 and the estimated pose P_{21} from I_2 to I_1 . Next, we compute the differences between the estimated relative pose P_{21} and P_{12}^{-1} . Specifically, we compute the angular difference d_R between relative rotations R_{21} and R_{12}^T , and the Euclidean distance d_t between relative translations t_{21} and $-R_{12}^T t_{12}$ respectively via:

$$\begin{aligned} d_R &= \arccos((\text{tr}(R_{21}R_{12}) - 1)/2), \\ d_t &= \|t_{21} + R_{12}^T t_{12}\|_2, \end{aligned} \quad (4)$$

where ‘ $\text{tr}(\cdot)$ ’ denotes matrix trace, ‘ $\|\cdot\|_2$ ’ denotes L_2 norm.

The distributions of the rotation difference d_R and translation difference d_t from the above two models are shown in Fig. 1. As seen from this figure, the pose differences (both rotation and translation) estimated by these models are generally larger than zero (particularly some of them are not trivial), demonstrating that the above models do not satisfy the PPE property. Such an issue may pose risks to downstream applications that require a stable and consistent estimation of relative poses, such as robotics and visual navigation. Moreover, it is also noted that some other visual tasks involve their specific equivariances, different from the PPE property in the relative pose estimation task, and once an existing model among these tasks is equipped with a task-specific equivariance, its performance could be further boosted [4, 27, 35, 39, 43].

Motivated by the above issues, we investigate whether it is possible to explore such a relative pose estimation framework that could not only bestow the PPE property on various relative pose estimation networks but also improve their

performances further. Specifically, we first introduce the concept of PPE mapping, which refers to a mapping that captures the PPE property in relative poses. Then, we establish the foundational principle for equivariant relative pose estimation. Accordingly, we propose the Equipose framework which holds the PPE property theoretically. The Equipose could improve the performance of a pre-trained model even without fine-tuning, and fine-tuning could enhance their performances further.

To summarize, our main contributions include:

- **Equivariance:** The simple but intrinsic pose permutation equivariance (PPE) property is revealed for relative pose estimation, and the PPE mapping is defined.
- **Generality:** By imposing the PPE property, the framework Equipose for equivariant relative pose estimation is proposed. An arbitrary end-to-end relative pose estimation model could be seamlessly embedded into this framework as a baseline, and accordingly obtains the PPE property as analyzed in theory in Sec. 3. The Equipose framework and the theoretical analyses might give new insights into model design strategies for relative pose estimation.
- **Flexibility:** Thanks to the imposed PPE property, the performances of the derived methods from the Equipose framework by utilizing many off-the-shelf baseline models, would be boosted even without fine-tuning by adding *several lines of code*. Moreover, once they are fine-tuned under Equipose, their performances would be further improved, as demonstrated by the results in Sec. 4.

2. Related Work

In this section, we review some typical matching-based methods and end-to-end methods for relative pose estimation respectively.

Matching-Based Relative Pose Estimation. The matching-based relative pose estimation methods typically involve the following two steps: establishing point correspondence via image matching [3, 33] and solving the two-view geometry with RANSAC [21, 24, 41]. Early approaches relied on handcrafted keypoint detectors and descriptors to establish point correspondences, such as SIFT [33] and SURF [3]. In recent years, numerous methods emerged that replace the handcrafted detectors or descrip-

tors with their learned counterparts [2, 17, 18, 32, 34, 37, 44, 51, 59]. Additionally, a variety of learning-based image-matching methods directly established point correspondences from keypoints or raw images [8, 31, 48, 50, 60].

However, since these methods depend on point correspondences, they may fail in cases of weak textures, repeated textures, large viewpoint changes, and degenerated configurations [20]. Additionally, these methods could be time-consuming due to image matching and RANSAC loops. Moreover, they could only recover the relative pose up to an unknown scale factor [24].

End-to-End Relative Pose Estimation. Recently, unlike the above strategies, many works have adopted deep neural networks to regress relative poses in an end-to-end manner [1, 6, 9, 12, 19, 28, 36]. These methods do not explicitly implement image matching or solve the two-view geometry problem. Instead, the networks directly learn the underlying mapping from the input image pair to the corresponding relative pose. Compared with matching-based methods, they are shown to be more robust in challenging scenes [6, 9, 19] with a significantly faster inference speed. Moreover, these end-to-end methods could be further incorporated with traditional methods for improving robustness [47].

These end-to-end methods generally adopt Siamese architectures [5] to extract feature, which are aggregated and decoded into the relative pose subsequently [6, 12, 26, 28, 46]. To facilitate stable training, early approaches [19, 36] parameterized the rotation as quaternions. Different from them, Zhou *et al.* [62] suggested to represent 3D rotations in a 6D space, which benefits the training of neural networks. Cai *et al.* [6] and Dekel *et al.* [12] showed that the network could learn from implicit cues, allowing the network to estimate the relative pose even when two images have no overlap. Rockwell *et al.* [46] introduced the essential matrix module to emulate the 8-point algorithm [24]. Arnold *et al.* [1] adopted a differentiable warping module for feature aggregation. Khatib *et al.* [28] embedded a semi-dense image matcher [50] for feature extraction to pursue a better generalization ability.

However, the existing end-to-end methods generally do not satisfy the PPE property, leading to inconsistent estimation when permuting the input order of an image pair as shown in Sec. 1. Unlike them, this work does not aim to propose a new model for relative pose estimation, but pursue a general framework that could bestow the PPE property on various relative pose estimation models, and improve their performances further.

3. Methodology

In this section, we first introduce the key definitions and propositions related to the pose permutation equivariance (PPE) property of relative poses. Then, we describe the proposed EquiPose framework in detail.

3.1. PPE Mapping and Preliminary Knowledge

Prior to the introduction of the PPE mapping, ‘equivariant mapping’ in mathematics is defined as [30]:

Definition 1 (Equivariant mapping). Let $\mathcal{G}$ be a group that act on spaces $\mathcal{X}$ and $\mathcal{Y}$, and $\mathbb{T}$ and $\mathbb{T}'$ be the induced actions of $\mathcal{G}$ on $\mathcal{X}$ and $\mathcal{Y}$. A mapping $f : \mathcal{X} \rightarrow \mathcal{Y}$ is *equivariant* with the action in $\mathcal{G}$ if $f(\mathbb{T}_g(x)) = \mathbb{T}'_g(f(x))$, $\forall x \in \mathcal{X}, g \in \mathcal{G}$.

According to Definition 1 and Eq. (1), a PPE mapping with the PPE property is defined as:

Definition 2 (Pose permutation equivariant mapping). A mapping $f : \mathbb{R}^{H \times W} \times \mathbb{R}^{H \times W} \rightarrow \text{SE}(3)$ (the special Euclidean group) is *pose permutation equivariant*, if $f(I_2, I_1) = f(I_1, I_2)^{-1}$, $\forall I_1, I_2 \in \mathbb{R}^{H \times W}$.

Definition 2 could be considered as an instance of ‘equivariant mapping’ from Definition 1 on relative camera poses, where the input variable x is an image pair $\{I_1, I_2\}$, the action $\mathbb{T}_g$ on $\{I_1, I_2\}$ is an order permutation operator, and the action $\mathbb{T}'_g$ is the inverse of a matrix.

It is noted that Definition 2 is applicable to neural networks, since any neural networks essentially aim to learn a mapping from their input to the output.

In addition to the above two definitions, we introduce the following definition on the geodesic mean of rotations [22, 23], as well as two theorems, which would be used to support the fundamental principle for designing a relative pose estimation framework with the PPE property in the next subsection:

Definition 3 (Geodesic mean of rotations). Given a set of rotations $\{R_i\}_{i=1}^n$, their geodesic mean S is defined as:

$$S = \mathfrak{G}(R_1, \dots, R_n) = \arg \min_{S \in \text{SO}(3)} \sum_{i=1}^n \|\log(R_i^T S)\|_2^2, \quad (5)$$

where ‘ $\log(\cdot)$ ’ is the mapping from an element in the Lie group to its corresponding Lie algebra.

From Definition 3, we have the two theorems as:

Theorem 1. Given a set of rotations $\{R_i\}_{i=1}^n$, it always holds that: $\mathfrak{G}(R_{\sigma(1)}, \dots, R_{\sigma(n)}) = \mathfrak{G}(R_1, \dots, R_n)$ for any permutation σ .

Theorem 2. If $S \in \text{SO}(3)$ is the geodesic mean of the set of rotations $\{R_i\}_{i=1}^n$, then S^T is the geodesic mean of the set of rotations $\{R_i^T\}_{i=1}^n$.

Due to the space limitation, the proofs of the above two theorems are given in the supplementary material.

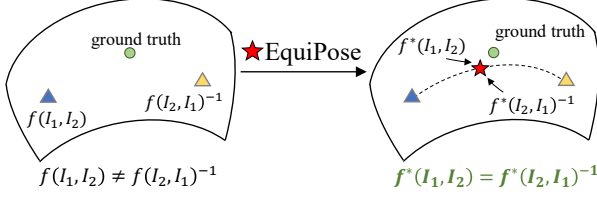


Figure 2. Illustration of the existing methods f and our method f^* on the $SE(3)$ manifold. **Left:** The existing method f does not capture the PPE property in the relative pose. **Right:** Within EquiPose f^* , the desired PPE property is satisfied (Theorem 3), and the results could be more reliable (Sec. 4).

3.2. Fundamental Principle of Pose Estimation Framework with PPE

According to the aforementioned PPE mapping and related preliminary knowledge, we elaborate the fundamental principle for designing a pose estimation framework with PPE, which could use an arbitrary end-to-end relative pose estimation model as its baseline and derive the corresponding upgraded model with the PPE property as follows:

Given an arbitrary relative pose estimation network corresponding to a mapping f , there exists the following mapping f^* , which is a PPE mapping:

Theorem 3. Given an arbitrary mapping $f : \mathbb{R}^{H \times W} \times \mathbb{R}^{H \times W} \rightarrow SE(3)$, where $f(I_1, I_2) := (R_{12}, t_{12})$ and $f(I_2, I_1) := (R_{21}, t_{21})$, the mapping f^* defined as:

$$f^*(I_1, I_2) := (R_{12}^*, t_{12}^*), \quad (6)$$

where

$$R_{12}^* = \mathfrak{G}(R_{12}, R_{21}^T), \quad (7)$$

$$t_{12}^* = (t_{12} - R_{12}^* t_{21})/2, \quad (8)$$

is a PPE mapping.

Proof. According to Theorem 1, we have

$$R_{12}^* = \mathfrak{G}(R_{12}, R_{21}^T) = \mathfrak{G}(R_{21}^T, R_{12}) \quad (9)$$

Then, according to Theorem 2, we have:

$$\mathfrak{G}(R_{21}^T, R_{12}) = \mathfrak{G}(R_{21}, R_{12}^T)^T = R_{21}^{*T} \quad (10)$$

According to Eqs. (9) and (10), it holds:

$$R_{12}^* = R_{21}^{*T}, \quad (11)$$

i.e., f^* is permutation equivariant in terms of rotation.

Now, according to (6), (7), (8) and (11), it holds:

$$\begin{aligned} f^*(I_1, I_2)^{-1} &= (R_{12}^{*T}, -R_{12}^{*T} t_{12}^*) \\ &= (R_{12}^{*T}, -R_{12}^{*T} (t_{12} - R_{12}^* t_{21})/2) \\ &= (R_{21}^*, (t_{21} - R_{21}^* t_{12})/2) \\ &= f^*(I_2, I_1), \end{aligned} \quad (12)$$

which completes the proof. $\square$

Algorithm 1 The EquiPose Framework

Input: A relative pose estimation network f ;
A pair of images $\{I_1, I_2\}$

Output: Equivariant pose estimation R_{12}^*, t_{12}^*

- 1: $R_{12}, t_{12} \leftarrow f(I_1, I_2)$
- 2: $R_{21}, t_{21} \leftarrow f(I_2, I_1)$
- 3: $R_{12}^* \leftarrow R_{12}(\exp(\frac{1}{2} \log(R_{12}^T R_{21}^T)))$
- 4: $t_{12}^* \leftarrow (t_{12} - R_{12}^* t_{21})/2$

For ease of understanding, we visualize the difference between f and f^* in Fig. 2. As seen from this figure, given two input images $\{I_1, I_2\}$, the existing relative pose estimation model (considered as a mapping) f does not satisfy the PPE property. Once it is embedded into EquiPose, the upgrade version f^* captures the PPE property strictly. Moreover, its estimation could be more reliable than the baseline model f , as demonstrated in Sec. 4 (also in Sec. S8.3 of the supplementary material).

It is noted in Eq. (7) that the construction of f^* involves computing the geodesic mean of two rotations, which could be computed explicitly via [38]:

$$\mathfrak{G}(R_{12}, R_{21}^T) = R_{12}(R_{12}^T R_{21}^T)^{1/2}. \quad (13)$$

Considering that we work on the 3D rotation group, the square root of a rotation $R \in SO(3)$ could be computed by:

$$R^{1/2} = \exp(\frac{1}{2} \log(R)), \quad (14)$$

where ‘ $\exp(\cdot)$ ’ is the mapping from the Lie algebra to the Lie group.

The above analysis reveals that given an arbitrary relative pose estimation network, once its upgrade version f^* is constructed by (6), the mapping is a pose permutation equivariant mapping, this is to say, the upgrade version has the PPE property.

3.3. EquiPose

According to the revealed point above as well as Eqs. (7) and (8), a simple framework for relative pose estimation, i.e., the EquiPose, could be straightforwardly built.

Specifically, the proposed EquiPose uses an existing model as its baseline, and outputs the final estimation on relative pose by calculating Eqs. (7) and (8), whose variable values are assigned by the baseline with or even without fine-tuning. Algorithm 1 shows the complete pipeline of EquiPose to obtain the relative pose.

It has to be pointed out that the computational cost of f^* in this implementation is twice that of f , since it involves two times of forward propagation of f : one for $f(I_1, I_2)$, and one for $f(I_2, I_1)$ after permuting the input images $\{I_1, I_2\}$. However, in practice, the computational

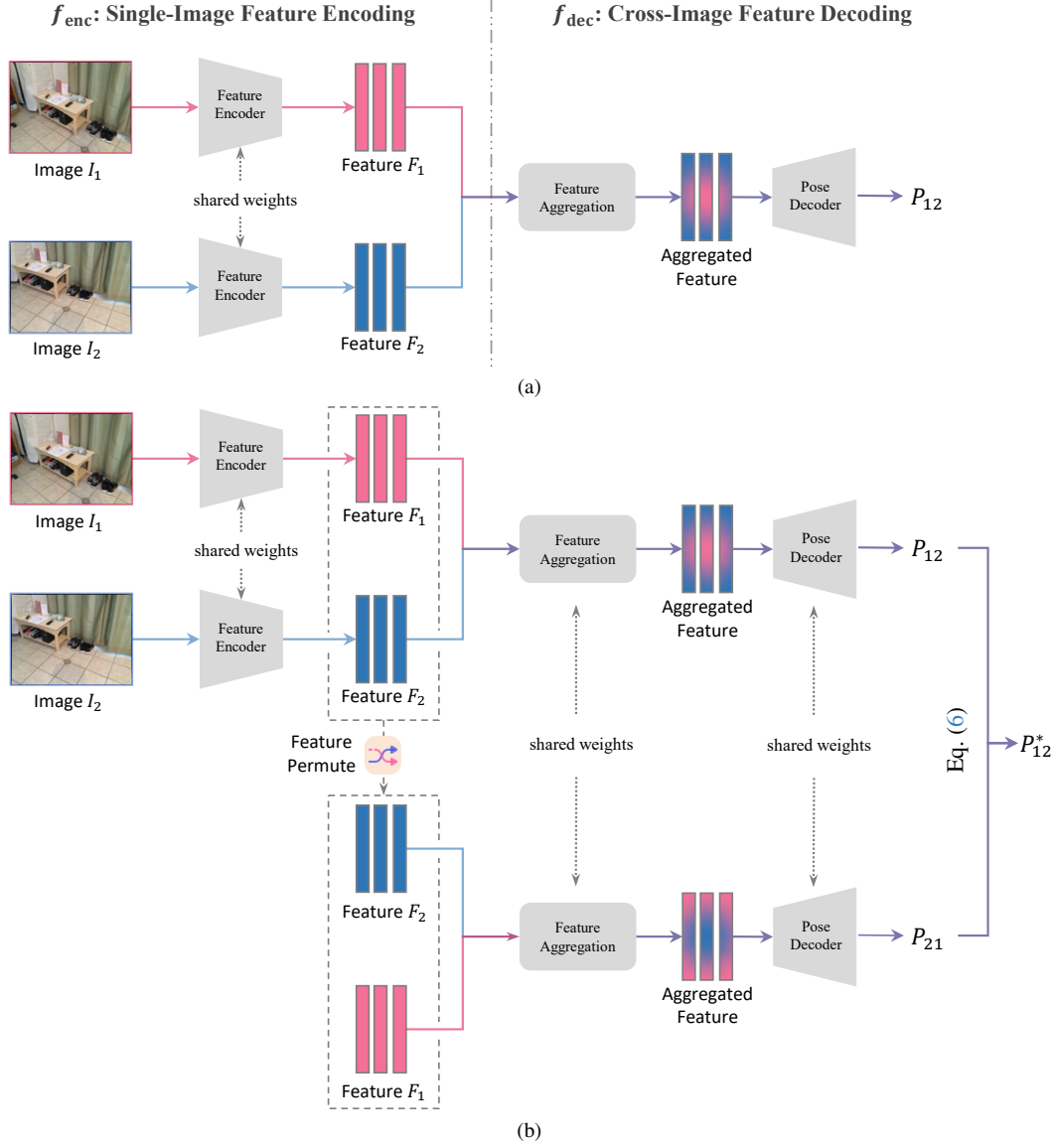


Figure 3. **The pipelines of existing methods and our method.** (a) The pipeline of existing relative pose estimation networks, which consists of two stages: Stage (i) Single-Image Feature Encoding (f_{enc}) extracts the feature maps $\{F_1, F_2\}$ from $\{I_1, I_2\}$. Stage (ii) Cross-Image Feature Decoding (f_{dec}) takes the extracted two feature maps $\{F_1, F_2\}$ as input, and it outputs the estimated relative pose. (b) The pipeline of EquiPose. The computational overhead from EquiPose is an additional forward propagation of f_{dec} , which is slight.

overhead could be reduced further with the *feature permutation* strategy, which is elaborated in the following.

To begin with, it is noted that the general pipeline of existing relative pose estimation models [1, 6, 9, 12, 19, 36, 46] could be summarized in Fig. 3a. As is seen, the pipeline consists of the two stages, which we name as (i) Single-Image Feature Encoding (denoted as f_{enc}), and (ii) Cross-Image Feature Decoding (denoted as f_{dec}) respectively.

(i) Single-Image Feature Encoding f_{enc} : This stage takes two images as inputs, and it outputs the corresponding feature maps $\{F_1, F_2\}$, *i.e.*, $f_{\text{enc}}(I_1, I_2) := \{F_1, F_2\}$.

It contains a feature encoder, which could be any feature-extraction backbone in computer vision. It should be noted that the weights of the feature encoder for extracting the features of image I_1 and I_2 are *shared* (generally referred to as the Siamese architecture [5, 29]).

(ii) Cross-Image Feature Decoding f_{dec} : This stage takes the feature maps $\{F_1, F_2\}$ obtained from Stage (i) as input, and it outputs the estimated relative pose P_{12} , *i.e.*, $f_{\text{dec}}(F_1, F_2) := P_{12}$. It contains a feature aggregation module and a pose decoder. The feature aggregation module maps $\{F_1, F_2\}$ into a single aggregated feature map (or vec-

Method	Rot. AUC $\uparrow$			Trans. AUC $\uparrow$			RE $\downarrow$	TE $\downarrow$
	@3 $^\circ$	@5 $^\circ$	@10 $^\circ$	@.1m	@.2m	@.5m		
ER[6]	13.15	28.23	50.43	-	-	-	14.66	-
+EquiPose	15.22	30.40	51.51	-	-	-	14.11	-
+EquiPose †	16.00	31.01	52.98	-	-	-	12.53	-
Reg6D[62]	10.16	22.97	45.76	-	-	-	12.26	-
+EquiPose	11.76	25.28	47.94	-	-	-	11.76	-
+EquiPose †	16.14	31.58	54.20	-	-	-	10.81	-
EightVit[46]	23.69	40.25	61.92	14.90	32.95	58.47	8.90	0.41
+EquiPose	24.58	41.32	62.82	15.49	34.11	60.03	8.40	0.35
+EquiPose †	25.30	42.54	64.05	16.37	36.14	63.07	7.97	0.31
MF6D[1]	22.36	37.19	57.10	14.39	32.57	57.34	9.81	0.44
+EquiPose	25.49	40.53	59.94	17.51	35.78	60.49	8.42	0.35
+EquiPose †	27.14	42.99	62.66	17.79	37.70	63.48	7.47	0.29
GRelPose[28]	28.90	46.36	66.53	16.46	35.36	60.06	8.16	0.38
+EquiPose	30.70	48.02	67.90	17.85	37.17	61.80	7.97	0.32
+EquiPose †	32.32	49.74	69.37	19.67	40.40	65.95	7.61	0.28

Table 1. **Comparison on the ScanNet dataset [11, 50].** **RE**: rotation errors in degrees. **TE**: translation errors in meters. **+EquiPose**: the models are implemented under EquiPose only during the inference stage. **+EquiPose †** : the models are further fine-tuned under EquiPose. -: the models only predict the relative camera rotation and thus could not be evaluated in translation. The best methods for each baseline are marked in **bold**.

tor), which is subsequently decoded by the pose decoder to obtain the estimated relative pose.

Since the weights of the feature encoder in f_{enc} are shared, we do not have to implement two complete forward propagation of f . Instead, we could reuse the extracted feature maps $\{F_1, F_2\}$ at the first forward propagation, then permute the feature maps to $\{F_2, F_1\}$, and implement one additional forward propagation of f_{dec} .

Fig. 3b illustrates the EquiPose framework with the feature permutation strategy. As is seen, it first performs one forward propagation of f_{enc} to extract feature maps $\{F_1, F_2\}$. Then, the feature maps $\{F_1, F_2\}$ and its permuted counterpart $\{F_2, F_1\}$ are inputted to f_{dec} to obtain P_{12} and P_{21} respectively. The final relative pose is obtained via P_{12} and P_{21} through Eq. (6). Compared to the original implementations of the baseline models, EquiPose only involves an additional forward propagation of f_{dec} .

Under the EquiPose framework, a pre-trained model could be directly adopted as the baseline without fine-tuning to boost its performance. Moreover, the model could also be further fine-tuned under EquiPose to achieve further improvement. We simply use an $\mathcal{L}_1$ loss for fine-tuning:

$$\mathcal{L}_1 = \|\hat{R}^* - \hat{R}\|_1 + \lambda \|\hat{t}^* - \hat{t}\|_1, \quad (15)$$

where $\hat{R}$ and $\hat{t}$ are the ground truth relative rotation and translation. λ is a preset constant factor.

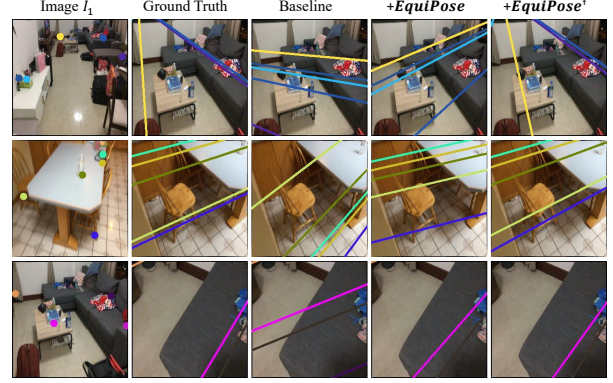


Figure 4. **Visualization of epipolar lines on ScanNet [11].** Five SIFT keypoints are sampled from image I_1 (column 1) and their epipolar lines are projected onto image I_2 , using the pose from: ground truth (column 2), GRelPose baseline [28] (column 3), EquiPose without (column 4) and with finetuning (column 5).

4. Experiment

4.1. Datasets and Evaluation Metrics

Datasets. We evaluate our method on both indoor and outdoor scenes. For indoor scenes, we use the ScanNet [11], the 7-Scene [49] and the 12-Scene [52] datasets, which are widely used for indoor pose estimation. For outdoor scenes, we use the recently proposed Mapfree dataset [1].

For ScanNet, we use 2M image pairs from the training set for training, and use the test set which contains 1500 pairs for evaluation following [8, 48, 50, 55, 56]. For both 7-Scene and 12-Scene, we use 720 image pairs for evaluation using the model trained on ScanNet. For Mapfree, we follow the official setup in [1] for training and evaluation. More details are provided in the supplementary material.

Evaluation Metrics. For ScanNet [11], 7-Scene [49] and 12-Scene [52], we calculate the area under the cumulative curve (AUC) as done in [31, 48, 50]. We report the AUC for rotation (3 $^\circ$ /5 $^\circ$ /10 $^\circ$) and translation (0.1m/0.2m/0.5m) respectively. Besides, we also report the average rotation and translation errors respectively. For the Mapfree dataset [1], we report the pose error and virtual correspondence re-projection error (VCRE) respectively following [1].

4.2. Baselines

It should be pointed out that the goal of this study is to introduce a general framework for existing end-to-end relative pose estimation models to achieve the PPE property with improved performances. Thus, we do not further evaluate and analyze other traditional matching-based methods, such as the classic SIFT [33] followed by the 5-point solver [41].

Specifically, we adopt the following five end-to-end models as baselines and evaluate the effectiveness of EquiPose in boosting their performances: (i) **ExtremeRotation** [6], (ii) **Reg6D** [6, 62], (iii) **EightVit** [46], (iv)

Method	7-Scene								12-Scene							
	Rot. AUC↑			Trans. AUC↑			RE↓	TE↓	Rot. AUC↑			Trans. AUC↑			RE↓	TE↓
	@3°	@5°	@10°	@.1m	@.2m	@.5m			@3°	@5°	@10°	@.1m	@.2m	@.5m		
ER[6]	11.17	24.28	46.70	-	-	-	9.82	-	14.61	31.60	57.15	-	-	-	6.94	-
+EquiPose	13.72	26.60	47.83	-	-	-	9.09	-	17.57	34.39	58.96	-	-	-	6.66	-
+EquiPose [†]	13.82	27.14	49.35	-	-	-	7.84	-	17.83	35.38	60.10	-	-	-	5.39	-
Reg6D[62]	10.08	22.44	45.92	-	-	-	7.78	-	13.16	28.42	55.27	-	-	-	5.67	-
+EquiPose	11.20	23.77	46.93	-	-	-	7.67	-	14.05	30.08	56.62	-	-	-	5.41	-
+EquiPose [†]	13.26	27.32	51.60	-	-	-	6.60	-	16.99	33.55	59.59	-	-	-	5.07	-
EightVit[46]	19.77	35.00	59.25	13.99	32.05	61.11	4.92	0.22	30.09	50.39	72.95	38.00	62.80	84.03	2.89	0.08
+EquiPose	19.88	35.51	59.87	14.21	32.91	62.90	4.78	0.20	31.17	51.53	73.58	38.67	63.72	84.59	2.78	0.08
+EquiPose [†]	20.87	37.01	61.44	14.47	33.73	65.22	4.56	0.19	32.74	52.72	74.41	40.63	65.35	85.45	2.71	0.07
MF6D[1]	18.73	34.46	57.62	11.90	30.35	60.13	4.98	0.23	30.44	51.07	73.11	36.37	61.52	83.24	2.79	0.09
+EquiPose	21.78	37.40	60.03	15.46	34.28	63.71	4.71	0.19	34.17	54.29	74.98	40.87	64.67	85.02	2.56	0.08
+EquiPose [†]	24.44	40.83	63.32	17.17	36.90	66.44	4.29	0.18	35.80	55.36	75.82	43.02	66.36	85.78	2.45	0.07
GRelPose[28]	26.78	45.47	67.95	15.70	34.27	62.48	3.96	0.21	35.62	55.20	75.63	39.87	63.81	84.34	2.58	0.08
+EquiPose	28.44	46.85	68.87	15.81	34.41	63.48	3.88	0.20	37.13	56.67	76.71	42.24	65.65	85.21	2.41	0.07
+EquiPose [†]	29.76	48.24	70.08	15.83	35.43	64.91	3.66	0.19	38.63	58.11	77.86	44.19	67.43	86.04	2.24	0.07

Table 2. Cross-dataset evaluation on the 7-Scene [49] and 12-Scene [52] datasets. All the models are trained on ScanNet.

MF6D [1], and (v) GRelPose [28]. Note that (i) ExtremeRotation and (ii) Reg6D only predict relative rotation, while the remaining ones could predict the 6 DoF relative camera pose (including both rotation and translation).

4.3. Comparative Evaluation

Evaluation Setup. For each of the five baseline models, including ER (ExtremeRotation) [6], Reg6D [62], EightVit [46], MF6D [1] and GRelPose [28], we compare the following variants: 1) the model in its original implementation, 2) the model implemented under EquiPose only during the inference stage (denoted with ‘+EquiPose’), 3) the model fine-tuned under EquiPose (denoted with ‘+EquiPose[†]’).

Indoor Pose Evaluation. The evaluation results on ScanNet are reported in Table 1. As seen from this table, the five models under the EquiPose framework achieve consistently better performance than their corresponding baselines under the eight metrics. Moreover, these models after fine-tuning under EquiPose could achieve further enhancements in most cases by a significant margin. For example, an inference-only implementation of GRelPose under EquiPose leads to a relative decrease of 2.3%(1-7.97/8.16) and 15.8%(1-0.32/0.38) in rotation and translation errors respectively. Its fine-tuned implementation has a relative reduction of **6.7%** and **26.3%** in rotation and translation errors respectively. The fine-tuned MF6D could even achieve a relative reduction of **23.9%** and **34.1%** in rotation and translation errors respectively. These results validate the effectiveness of the proposed EquiPose framework.

The cross-dataset evaluation results on the 7-Scene [49] and 12-Scene [52] datasets are reported in Tab. 2, where the models trained on ScanNet are used for evaluation. As seen from the table, the five models under EquiPose could also achieve consistent improvements than their corresponding baselines, and fine-tuning further boosts the performances.

Method	Pose Error		VCRE		
	Pr.↑	Avg.Med.↓	Med.↓	Pr.@45↑	Pr.@90↑
MFBin[1]	9.1	210cm/33.7°	199.7	15.4	30.1
+EquiPose	12.0	185cm/32.5°	196.9	18.2	31.6
MFBinTS[1]	10.5	183cm/23.3°	166.4	18.1	35.5
+EquiPose	12.9	152cm/23.4°	166.0	21.7	39.3
MFQuat[1]	5.1	203cm/27.3°	160.7	14.3	33.6
+EquiPose	6.3	184cm/23.3°	166.4	16.6	35.0
MFQuatTS[1]	4.4	185cm/26.3°	154.8	14.2	34.7
+EquiPose	6.6	170cm/23.5°	157.6	17.3	36.4
MF6D[1]	6.1	168cm/22.5°	147.1	18.7	40.2
+EquiPose	9.6	153cm/18.6°	148.2	22.1	40.9
+EquiPose [†]	10.1	148cm/18.4°	145.0	23.0	42.1
EightVit[46]	6.0	204cm/26.6°	155.2	17.8	35.8
+EquiPose	6.4	195cm/26.3°	157.2	18.3	36.4
+EquiPose [†]	6.8	188cm/25.1°	153.1	19.1	37.0
GRelPose[28]	12.7	168cm/22.4°	137.7	25.8	43.3
+EquiPose	13.0	166cm/21.9°	136.4	26.1	43.3
+EquiPose [†]	13.2	163cm/21.7°	132.9	26.2	44.1

Table 3. Comparison on the Mapfree dataset [1]. As done in [1], we report Pose Error in terms of rotation error and translation error and VCRE in terms of median error and precision.

These results demonstrate that EquiPose is also effective under the cross-dataset setup.

Besides the above quantitative results, we also provide visualization results from ScanNet. Considering GRelPose [28] performs relatively better than other baseline models in most cases, we adopt it as the baseline and compare it with EquiPose. The corresponding results are shown in Fig. 4. As seen from this figure, the proposed EquiPose is effective in improving the performance of relative pose estimation across different types of challenges, such as large viewpoint changes, and texture-less patterns. An inference-only implementation demonstrates obvious performance improvements, while subsequent fine-tuning could achieve

Method	d_R	d_t	Rot. AUC			Trans. AUC			RE	TE
			@3°	@5°	@10°	@.1m	@.2m	@.5m		
EightVit[46]	2.7	0.12	23.69	40.25	61.92	14.90	32.95	58.47	8.90	0.41
+RP	1.2	0.05	22.72	39.93	61.83	14.60	33.82	61.23	8.23	0.33
+RP+EquiPose	0	0	22.96	40.11	62.03	15.02	34.30	61.73	8.20	0.31
+EquiPose [†]	0	0	25.30	42.54	64.05	16.37	36.14	63.07	7.97	0.31
MF6D[1]	7.3	0.18	22.36	37.19	57.10	14.39	32.57	57.34	9.81	0.44
+RP	5.4	0.11	20.42	36.62	58.11	14.56	33.01	60.36	8.17	0.32
+RP+EquiPose	0	0	24.61	40.26	60.85	16.43	35.63	62.75	7.74	0.30
+EquiPose [†]	0	0	27.14	42.99	62.66	17.79	37.70	63.48	7.47	0.29
GRelPose[28]	3.2	0.12	28.90	46.36	66.53	16.46	35.36	60.06	8.16	0.38
+RP	2.1	0.06	28.11	45.66	66.36	16.95	37.65	63.64	7.74	0.30
+RP+EquiPose	0	0	29.02	46.35	66.80	18.20	38.88	64.47	7.65	0.29
+EquiPose [†]	0	0	32.32	49.74	69.37	19.67	40.40	65.95	7.61	0.28

Table 4. **EquiPose VS. Random Permutation.** ‘+RP’ denotes the model is trained with random permutation. d_R and d_t denote the rotation and translation difference computed via Eq. (4).

additional performance gains.

Outdoor Pose Evaluation. The Mapfree dataset [1] was introduced for evaluating 6DoF relocalization, thus we only evaluate EightVit, MF6D, and GRelPose which could predict 6DoF poses. As a supplement, we further evaluate EquiPose in an inference-only manner on four additional baselines, including MFBin, MFBinTS, MFQuatTS which were released¹ along with MF6D [1]. The results are reported in Tab. 3. As seen from this table, EquiPose could boost the performances of the above seven models in most cases. And the fine-tuned MF6D+EquiPose[†] achieves the lowest pose error among them. These results validate the effectiveness of our method under the outdoor setup.

EquiPose VS. Random Permutation. Intuitively, another way to leverage the PPE property for relative pose estimation is to permute the input order of image pairs randomly during training. To evaluate the effect of random permutation, we have further trained and tested EightVit [46], MF6D [1] and GRelPose [28] on the ScanNet dataset by using random permutation (RP). The corresponding results are reported in Tab. 4. In addition, we also report the average values of d_R and d_t computed via Eq. (4).

As seen from this table, random permutation could improve model performances in average rotation and translation errors. However, the models trained under EquiPose perform significantly better than using random permutation. Besides, random permutation could not strictly enforce the PPE property as evidenced by both d_R and d_t being larger than zero, while EquiPose could achieve PPE strictly. To this end, EquiPose is not opposed to using random permutation, *i.e.*, by enforcing the PPE property, it could also boost the performance of models trained using random permutation as observed in this table. The above results further validate the effectiveness of our method.

¹Pretrained weights are downloaded from the website of Mapfree: <https://github.com/nianticlabs/map-free-reloc>.

Method	Params.	FLOPs	Time (ms)
ER	1.92×10^7	9.86×10^9	2.0
ER+EquiPose	1.92×10^7	1.09×10^{10}	2.3
Reg6D	2.33×10^7	9.91×10^9	2.6
Reg6D+EquiPose	2.33×10^7	9.97×10^9	2.6
EightVit	1.91×10^7	6.88×10^9	3.5
EightVit+EquiPose	1.91×10^7	9.64×10^9	5.1
MF6D	2.19×10^7	4.57×10^{10}	5.6
MF6D+EquiPose	2.19×10^7	4.60×10^{10}	6.7
GRelPose	1.37×10^7	9.82×10^9	11.8
GRelPose+EquiPose	1.37×10^7	1.13×10^{10}	12.0

Table 5. Comparison of parameter numbers, FLOPs, and inference times (ms) per image on an RTX 3090Ti GPU.

4.4. Computation Comparison and Limitation

Here, we compare the parameter numbers, Floating Point Operations (FLOPs) and the inference times of the baselines and their EquiPose implementations. All the models are evaluated on an RTX 3090Ti GPU on ScanNet. The results are reported in Tab. 5. As is seen, compared with the baseline model, the EquiPose implementation does not introduce any additional parameters, since it uses the same backbone as the baseline, and thus does not incur extra storage costs. Moreover, compared with each baseline model, EquiPose results in a modest increment in computational costs generally, due to the feature permutation strategy.

However, the extra computational cost depends on the architecture of the baseline model: For models with lightweight feature aggregation modules and pose decoders, the increase in computation is slight. For models with heavy feature aggregation modules and pose decoders, the extra computational cost would be high, which is a limitation of EquiPose. Despite this, all the models under EquiPose maintain an inference time of less than 15 ms per image, which is sufficient for practical applications.

5. Conclusions

This work focuses on end-to-end relative pose estimation. Firstly, we introduce the overlooked pose permutation equivariance (PPE) property for relative pose estimation. Leveraging this property, we establish the foundational principle for designing a relative pose estimation framework that incorporates the PPE property. Building on this theoretical basis, we propose the general and flexible *EquiPose* framework for relative pose estimation. Experiments on four public datasets further demonstrate that the performances of pre-trained models could be improved even without re-training, and moreover, their performances could be further boosted after being fine-tuned under EquiPose.

Acknowledgements. This work was supported by the National Natural Science Foundation of China (Grant No. 62376269).

References

- [1] Eduardo Arnold, Jamie Wynn, Sara Vicente, Guillermo Garcia-Hernando, Aron Monszpart, Victor Prisacariu, Daniyar Turmukhambetov, and Eric Brachmann. Map-free visual relocalization: Metric pose relative to a single image. In *European Conference on Computer Vision*, pages 690–708. Springer, 2022. 2, 3, 5, 6, 7, 8, 1, 4
- [2] Axel Barroso-Laguna, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Key. net: Keypoint detection by hand-crafted and learned cnn filters. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5836–5844, 2019. 3
- [3] Herbert Bay, Andreas Ess, Tinne Tuytelaars, and Luc Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110(3):346–359, 2008. 2
- [4] Georg Bökman, Johan Edstedt, Michael Felsberg, and Fredrik Kahl. Steerers: A framework for rotation equivariant keypoint descriptors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4885–4895, 2024. 2
- [5] Jane Bromley, Isabelle Guyon, Yann LeCun, and et al. Signature verification using a “siamese” time delay neural network. *Advances in Neural Information Processing Systems*, 6, 1993. 3, 5
- [6] Ruojin Cai, Bharath Hariharan, Noah Snavely, and Hadar Averbuch-Elor. Extreme rotation estimation using dense correlation volumes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 14566–14575, 2021. 1, 3, 5, 6, 7
- [7] David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19457–19467, 2024. 1
- [8] Hongkai Chen, Zixin Luo, Lei Zhou, Yurun Tian, Mingmin Zhen, Tian Fang, David Mckinnon, Yanghai Tsin, and Long Quan. Aspanformer: Detector-free image matching with adaptive span transformer. In *European Conference on Computer Vision*, pages 20–36. Springer, 2022. 3, 6
- [9] Kefan Chen, Noah Snavely, and Ameesh Makadia. Wide-baseline relative camera pose estimation with directional learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3258–3268, 2021. 3, 5
- [10] Shuai Chen, Tommaso Cavallari, Victor Adrian Prisacariu, and Eric Brachmann. Map-relative pose regression for visual re-localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20665–20674, 2024. 1
- [11] Angela Dai, Angel X. Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2017. 2, 6, 1, 3
- [12] Shay Dekel and Yosi Keller. Estimating extreme 3d image rotation with transformer cross-attention. *arXiv preprint arXiv:2303.02615*, 2023. 1, 3, 5
- [13] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition workshops*, pages 224–236, 2018. 1
- [14] Qiulei Dong, Xiang Gao, Hainan Cui, and Zhanyi Hu. Robust camera translation estimation via rank enforcement. *IEEE Transactions on Cybernetics*, 52(2):862–872, 2020. 1
- [15] Qiulei Dong, Shuang Deng, and Yuzhen Liu. Cayley rotation averaging: Multiple camera averaging under the cayley framework. *IEEE Transactions on Image Processing*, 2024. 1
- [16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1
- [17] Mihai Dusmanu, Ignacio Rocco, Tomas Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint detection and description of local features. In *Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 3
- [18] Johan Edstedt, Georg Bökman, Mårten Wadenbäck, and Michael Felsberg. Dedode: Detect, don’t describe—describe, don’t detect for local feature matching. In *2024 International Conference on 3D Vision*, pages 148–157. IEEE, 2024. 3
- [19] Sovann En, Alexis Lechervy, and Frédéric Jurie. Rpnnet: An end-to-end network for relative camera pose estimation. In *Proceedings of the European Conference on Computer Vision Workshops*, 2018. 1, 3, 5
- [20] Hongyi Fan, Joe Kileel, and Benjamin Kimia. On the instability of relative pose estimation and ransac’s role. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8935–8943, 2022. 1, 3
- [21] Martin A Fischler and Robert C Bolles. 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. 1, 2
- [22] Richard Hartley, Khurram Aftab, and Jochen Trumpf. L1 rotation averaging using the weiszfeld algorithm. In *CVPR 2011*, pages 3041–3048. IEEE, 2011. 3
- [23] Richard Hartley, Jochen Trumpf, Yuchao Dai, and Hongdong Li. Rotation averaging. *International Journal of Computer Vision*, 103:267–305, 2013. 3, 1
- [24] Richard I Hartley. In defense of the eight-point algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(6):580–593, 1997. 1, 2, 3
- [25] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 1
- [26] Will Hutchcroft, Yuguang Li, Ivaylo Boyadzhiev, Zhiqiang Wan, Haiyan Wang, and Sing Bing Kang. Covispose: Co-visibility pose transformer for wide-baseline relative pose estimation in 360 indoor panoramas. In *European Conference on Computer Vision*, pages 615–633. Springer, 2022. 3, 5

- [27] Nicolas Keriven and Gabriel Peyré. Universal invariant and equivariant graph neural networks. *Advances in Neural Information Processing Systems*, 32, 2019. 2
- [28] Fadi Khatib, Yuval Margalit, Meirav Galun, and Ronen Basri. Leveraging image matching toward end-to-end relative camera pose regression, 2024. 1, 2, 3, 6, 7, 8, 4, 5
- [29] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML Deep Learning Workshop*, 2015. 5
- [30] Risi Kondor and Shubhendu Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In *International Conference on Machine Learning*, pages 2747–2755. PMLR, 2018. 3
- [31] Philipp Lindenberger, Paul-Edouard Sarlin, and Marc Pollefeys. Lightglue: Local feature matching at light speed. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17627–17638, 2023. 3, 6
- [32] Yuzhen Liu and Qiulei Dong. Descriptor distillation: A teacher-student-regularized framework for learning local descriptors. *International Journal of Computer Vision*, pages 1–19, 2024. 3
- [33] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 1, 2, 6
- [34] Zixin Luo, Lei Zhou, Xuyang Bai, Hongkai Chen, Jiahui Zhang, Yao Yao, Shiwei Li, Tian Fang, and Long Quan. Aslfeat: Learning local features of accurate shape and localization. In *Proceedings of the 2019 IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 3
- [35] Clare Lyle, Mark van der Wilk, Marta Kwiatkowska, Yarin Gal, and Benjamin Bloem-Reddy. On the benefits of invariance in neural networks. *arXiv preprint arXiv:2005.00178*, 2020. 2
- [36] Iaroslav Melekhov, Juha Ylioinas, Juho Kannala, and Esa Rahtu. Relative camera pose estimation using convolutional neural networks. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 675–687. Springer, 2017. 1, 3, 5
- [37] Anastasiya Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. Working hard to know your neighbor’s margins: Local descriptor learning loss. In *Advances in Neural Information Processing Systems*, 2017. 3
- [38] Maher Moakher. Means and averaging in the group of rotations. *SIAM journal on matrix analysis and applications*, 24(1):1–16, 2002. 4
- [39] Dror Moran, Hodaya Koslowsky, Yoni Kasten, Haggai Maron, Meirav Galun, and Ronen Basri. Deep permutation equivariant structure from motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5976–5986, 2021. 2
- [40] Tony Ng, Adrian Lopez-Rodriguez, Vassileios Balntas, and Krystian Mikolajczyk. Ood-pose: camera pose regression from out-of-distribution synthetic views. In *International Conference on 3D Vision*, pages 722–732. IEEE, 2022. 1
- [41] David Nistér. An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, 2004. 2, 6
- [42] Linfei Pan, Dániel Baráth, Marc Pollefeys, and Johannes L Schönberger. Global structure-from-motion revisited. In *European Conference on Computer Vision*, 2024. 1
- [43] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 652–660, 2017. 2
- [44] Jerome Revaud, Cesar De Souza, Martin Humenberger, and Philippe Weinzaepfel. R2d2: Reliable and repeatable detector and descriptor. *Advances in Neural Information Processing Systems*, 32, 2019. 3
- [45] Jerome Revaud, Yohann Cabon, Romain Brégier, JongMin Lee, and Philippe Weinzaepfel. Sacreg: Scene-agnostic coordinate regression for visual localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 688–698, 2024. 1
- [46] Chris Rockwell, Justin Johnson, and David F. Fouhey. The 8-point algorithm as an inductive bias for relative pose prediction by vits. In *2022 International Conference on 3D Vision*, 2022. 1, 3, 5, 6, 7, 8, 2, 4
- [47] Chris Rockwell, Nilesh Kulkarni, Linyi Jin, Jeong Joon Park, Justin Johnson, and David F Fouhey. Far: Flexible accurate and robust 6dof relative camera pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19854–19864, 2024. 3
- [48] Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4938–4947, 2020. 2, 3, 6, 1
- [49] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2930–2937, 2013. 6, 7, 1
- [50] Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8922–8931, 2021. 2, 3, 6, 1
- [51] Yurun Tian, Axel Barroso-Laguna, Tony Ng, Vassileios Balntas, and Krystian Mikolajczyk. Hynet: Learning local descriptor with hybrid similarity measure and triplet loss. In *Advances in Neural Information Processing Systems*, pages 7401–7412, 2020. 3
- [52] Julien Valentin, Angela Dai, Matthias Nießner, Pushmeet Kohli, Philip Torr, Shahram Izadi, and Cem Keskin. Learning to navigate the energy landscape. In *International Conference on 3D Vision*, pages 323–332. IEEE, 2016. 6, 7, 1
- [53] Guangcong Wang, Zhaoxi Chen, Chen Change Loy, and Ziwei Liu. Sparsenerf: Distilling depth ranking for few-shot novel view synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9065–9076, 2023. 1
- [54] Guangyu Wang, Jinzhi Zhang, Fan Wang, Ruqi Huang, and Lu Fang. Xscale-nvs: Cross-scale novel view synthesis with

- hash featurized manifold. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21029–21039, 2024. [1](#)
- [55] Qing Wang, Jiaming Zhang, Kailun Yang, Kunyu Peng, and Rainer Stiefelhagen. Matchformer: Interleaving attention in transformers for feature matching. In *Asian Conference on Computer Vision*, 2022. [6](#)
- [56] Yifan Wang, Xingyi He, Sida Peng, Dongli Tan, and Xiaowei Zhou. Efficient lofr: Semi-dense local feature matching with sparse-like speed. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21666–21675, 2024. [6](#)
- [57] Jamie Watson, Oisín Mac Aodha, Victor Prisacariu, Gabriel Brostow, and Michael Firman. The temporal opportunist: Self-supervised multi-frame monocular depth. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1164–1174, 2021. [1](#)
- [58] Zhengxin Zhang, Qingjie Liu, and Yunhong Wang. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, 15(5):749–753, 2018. [1](#), [2](#)
- [59] Xiaoming Zhao, Xingming Wu, Jinyu Miao, Weihai Chen, Peter CY Chen, and Zhengguo Li. Alike: Accurate and lightweight keypoint detection and descriptor extraction. *IEEE Transactions on Multimedia*, 2022. [3](#)
- [60] Qunjie Zhou, Torsten Sattler, and Laura Leal-Taixe. Patch2pix: Epipolar-guided pixel-level correspondences. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4669–4678, 2021. [3](#)
- [61] Tinghui Zhou, Matthew Brown, Noah Snavely, and David G Lowe. Unsupervised learning of depth and ego-motion from video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1851–1858, 2017. [1](#)
- [62] Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019. [3](#), [6](#), [7](#), [1](#)