EquiPose: Exploiting Permutation Equivariance for Relative Camera Pose Estimation

Supplementary Material

This supplementary material is organized as follows: Sec. S6 gives the proofs of Theorems 1 and 2 in Sec. 3.1 of the main text. Sec. S7 provides details about datasets, baselines, and implementations. Sec. S8 provides additional experimental results and analysis.

S6. Theorem Proofs

Here, we give the proofs of two theorems regarding the geodesic mean in rotations *i.e.*:

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

$$S = \mathfrak{G}(R_1, R_2, \cdots, R_n) = \mathop{\arg\min}_{S \in \mathrm{SO}(3)} \sum_{i=1}^{n} \| \log(R_i^{\mathrm{T}} S) \|_2^2, \tag{16}$$

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

From the above definition, the following two theorems hold:

Theorem 1. Given a set of rotations $\{R_i\}_{i=1}^n$, it always holds that: $\mathfrak{G}(R_{\sigma(1)}, R_{\sigma(2)}, \cdots, R_{\sigma(n)}) = \mathfrak{G}(R_1, R_2, \cdots, R_n)$ for any permutation σ.

Proof. The theorem could be straightforwardly proved using the commutative law of addition:

$$\begin{aligned}
&\mathfrak{G}(R_1, R_2, \cdots, R_n) \\
&= \mathop{\arg\min}_{S \in \mathrm{SO}(3)} \sum_{i=1}^{n} \| \log(R_i^{\mathrm{T}} S) \|_2^2 \\
&= \mathop{\arg\min}_{S \in \mathrm{SO}(3)} \sum_{i=1}^{n} \| \log(R_{\sigma(i)}^{\mathrm{T}} S) \|_2^2 \\
&= \mathfrak{G}(R_{\sigma(1)}, R_{\sigma(2)}, \cdots, R_{\sigma(n)}).
\end{aligned} \tag{17}$$

□

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

Proof. Let S be the geodesic mean of the set of rotations $\{R_i^{\mathrm{T}}\}$, *i.e.*, $S = \mathop{\arg\min}_{S \in \mathrm{SO}(3)} \sum_{i=1}^{n} \| \log(R_i^{\mathrm{T}} S) \|_2^2$. Transposing S, we have:

$$S^{\mathrm{T}} = \mathop{\arg\min}_{S \in \mathrm{SO}(3)} \sum_{i=1}^{n} \| \log(R_i^{\mathrm{T}} S^{\mathrm{T}}) \|_2^2 \tag{18}$$

Since $\| \log(X^{\mathrm{T}} Y) \|_2 = \| \log(X Y^{\mathrm{T}}) \|_2, \forall X, Y \in \mathrm{SO}(3)$ [23], we have:

$$\mathop{\arg\min}_{S \in \mathrm{SO}(3)} \sum_{i=1}^{n} \| \log(R_i^{\mathrm{T}} S^{\mathrm{T}}) \|_2^2 = \mathop{\arg\min}_{S \in \mathrm{SO}(3)} \sum_{i=1}^{n} \| \log(R_i S) \|_2^2 \tag{19}$$

Then, according to Eqs. (18) and (19), we have

$$\begin{aligned}
S^{\mathrm{T}} &= \sum_{i=1}^{n} \| \log(R_i S) \|_2^2 \\
&= \mathop{\arg\min}_{S \in \mathrm{SO}(3)} \mathfrak{G}(R_1^{\mathrm{T}}, R_2^{\mathrm{T}}, \cdots, R_n^{\mathrm{T}})
\end{aligned} \tag{20}$$

which completes the proof. □

S7. Details

S7.1. Dataset Details

In this work, we have used three datasets for evaluation, including the ScanNet [11], the 7-Scene [49], the 12-Scene [52] and the Mapfree relocalization datasets [1]. These three datasets all provide ground truth camera poses and depth images.

For training on ScanNet, we sample 2M image pairs that have the overlap scores in [0.4,0.8] following [48, 50]. For evaluation, we use the same 1500 test image pairs as in [48, 50]. The 7-Scene and 12-Scene datasets are only used for evaluation, and we sample 720 pairs of images from each of them with overlap scores in [0.4,0.8]. For the Mapfree dataset [1], we follow the official setting in [1], *i.e.*, we train the models on 460 training scenes and evaluate them on the 130 testing scenes.

S7.2. Baseline Details

Here, we introduce the overall architecture of the baseline models:

- **ExtremeRotation** [6] adopts a Siamese ResUNet [58] as the feature encoder. The extracted features are aggregated via a correlation volume, which is then decoded into the relative rotation parameterized using Euler angles.
- **Reg6D** [6, 62] adopts a Siamese ResNet architecture [25] as the feature encoder. The extracted features are simply concatenated for aggregation, and then decoded into the rotation parameterized in a 6D space [62]. Same with ExtremeRotation, it only outputs the rotation.
- **EightVit** [46] adopts the Siamese ResNet [25] followed by several blocks from ViT [16] with self-attention as the

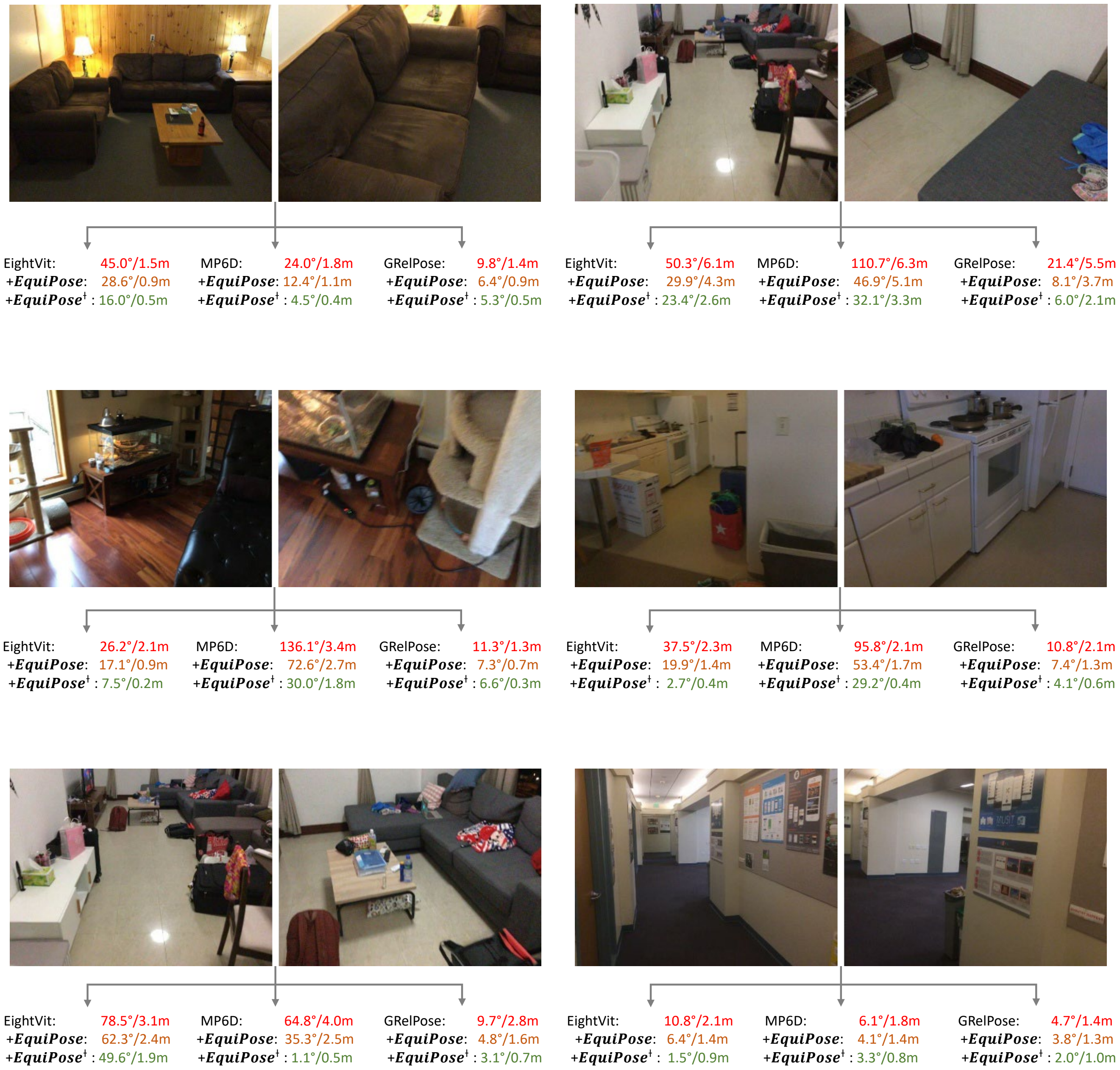


Figure S5. **Additional qualitative results on ScanNet [11].** We compare the three baseline models EightVit [46], MP6D [1], GRelPose [28], and their implementation under the EquiPose framework. For each case, we show the image pairs on the top and rotation/translation errors on the bottom. ***+EquiPose***: the models are implemented under EquiPose only during the inference stage. ***+EquiPose***†: the models are further fine-tuned under EquiPose.

feature encoder. The features are further aggregated with an essential matrix module and decoded into the 6D pose.

- **MF6D** [1] adopts the Siamese ResUNet [58] to encode images into feature maps, and then the feature map of the second image is warped and concatenated with the first one for aggregation. The aggregated features are finally flattened and decoded into the 6D pose via an MLP.
- **GRelPose** [28] adopts LoFTR [50] as the feature encoder to extract features from the two images respectively, and the features are warped together for aggregation. The aggregated features are flattened and decoded into the 6D pose via several MLPs.

Accordingly, the architectures of all the above baseline models could be described with the general pipeline as plotted in Fig. 3a of the main text. Thus, the introduced feature permutation strategy can be applied to all of them.

S7.3. Implementation Details

The evaluated baseline models in this work include ExtremeRotation [6], Reg6D [62], EightVit [46], MF6D [1], GRelPose [28].

Training Loss. For the baselines EightVit, MF6D, GRelPose, we use an $\mathcal{L}_1$ loss for training as done in [1]:

$$\mathcal{L}_1 = \|R - \hat{R}\|_1 + \lambda \|t - \hat{t}\|_1, \tag{21}$$

where R and t are the predicted relative rotation and translation from the model. $\hat{R}$ and $\hat{t}$ are the ground truth relative rotation and translation. λ is a weighting factor set to 1 as done in [1] for all experiments.

For Reg6D which only predicts the relative rotation, we only use the rotation part of the $\mathcal{L}_1$ loss in Eq. (21).

For ExtremeRotation [6] which outputs the discrete probability distribution of three Euler angles, we use the cross-entropy loss as done in [6].

Optimization Details. All the models are trained on an RTX 3090 Ti GPU (24GB). Following [1], we use the Adam optimizer with a learning rate of 1×10^{-4} and a batch size of 10 for optimization. The models are optimized for around 150M steps. During fine-tuning under EquiPose, we use a learning rate of 1×10^{-5}. For ExtremeRotation which outputs the discrete probability distributions of three Euler angles, we use soft argmax instead of hard assignment [6] to compute the Euler angles for gradients to backpropagate under EquiPose.

Inference Time. The inference time is measured on an RTX 3090Ti GPU (24GB). For all the models, we use a batch size of 20. To obtain the inference time, we first perform 20 reruns of warm-up. Then, the time costs are averaged over another 100 reruns.

S8. Additional Experiments and Analysis

S8.1. Additional Visualization Results

In Fig. S5, we present additional visualization results on the ScanNet dataset [11]. As seen from this figure, the proposed EquiPose framework could improve the performances of different models. Notably, the improvement is particularly pronounced for challenging samples, such as image pairs exhibiting large viewpoint changes and texture-less regions.

S8.2. Comparison of d_R and d_t

In Sec. 3 of the main text, we have theoretically proved that under the EquiPose framework, an arbitrary relative pose estimation model could achieve the pose permutation equivariant (PPE) property, *i.e.*, the estimated pose from image I_1 to image I_2 (denoted as P_{12}) should be the inverse of the relative pose from I_2 to I_1 (denoted as P_{21}). Here, we further confirm it by empirical experiments.

We adopt the test set of ScanNet which contains 1500 image pairs in total, and use EightVit [46], MF6D [1], GRelPose [28] to estimate P_{12} and P_{21} for these image pairs. As done in Sec. 1 of the main text, we compute the d_R and d_t as follows:

$$\begin{aligned} d_R &= \arccos((\mathrm{tr}(R_{21}R_{12}) - 1)/2) \\ d_t &= \|t_{21} + R_{12}^{\mathrm{T}} t_{12}\|_2 \end{aligned} \tag{22}$$

The histograms of d_R and d_t among the 1500 pairs for these baseline models as well as their implementations under EquiPose are shown in Fig. S6. Moreover, we also compute the mean and standard deviation values of d_R and d_t respectively, which are reported in the tables under each subfigure in Fig. S6.

As seen from the subfigures and tables, the baseline models generally do not satisfy the PPE property (d_R and d_t are generally larger than zero). However, when implemented under the proposed EquiPose framework, these models could effectively capture this property (d_R and d_t are equal to zero). These results further validate the PPE property of EquiPose empirically.

S8.3. Fine-Grained Analysis

In the main text, we have conducted comparative evaluation on the overall accuracy between the baseline models and EquiPose, where the metrics are computed and averaged over all image pairs in the datasets. Here, we provide a fine-grained analysis by evaluating the performance of EquiPose on each individual image pair.

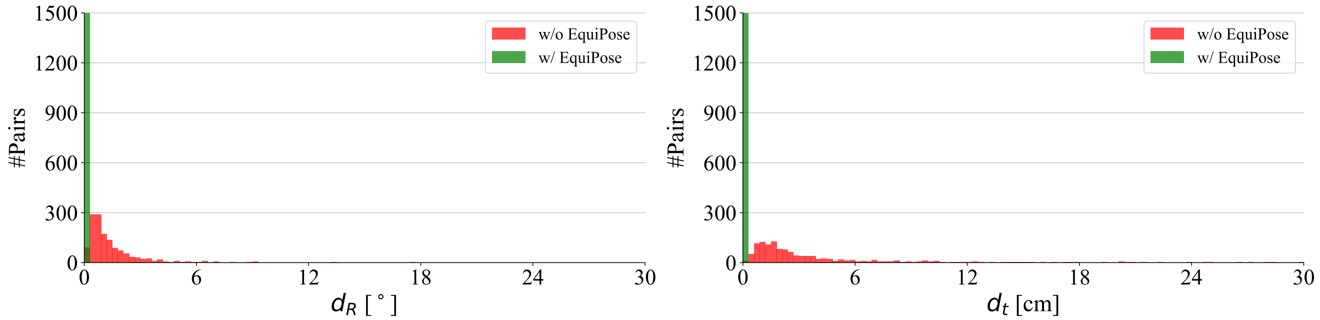
We first discuss the limitation of the existing evaluation criterion which prevents a more fine-grained analysis, and then introduce a novel evaluation criterion to address the issue. Next, we compare our method with the baseline models under this criterion.

S8.3.1 Expectation Error

According to the PPE property of relative camera poses, *i.e.*, $P_{12} = P_{21}^{-1}$, the model could obtain the relative camera pose P_{12} of an image pair $\{I_1, I_2\}$ via the following two input orders: $\{I_1, I_2\}$ and $\{I_2, I_1\}$.

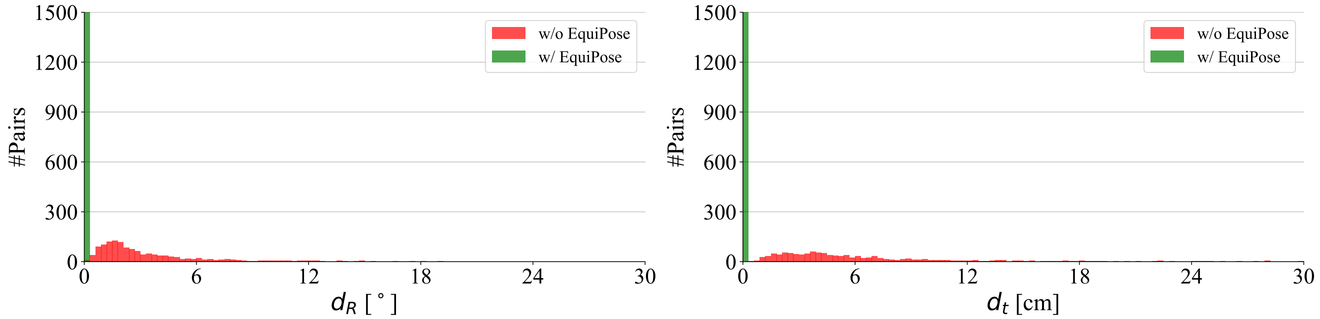
However, since existing models do not satisfy the PPE property as demonstrated in Sec. 1, the estimated pose from the above two orders by existing methods are generally inconsistent, and would result in a different evaluation result. Specifically, given an arbitrary input order of an image pair, the estimation would fall into one of the two cases:

Case #1: The adopted input order results in a better estimation than the reverse order.



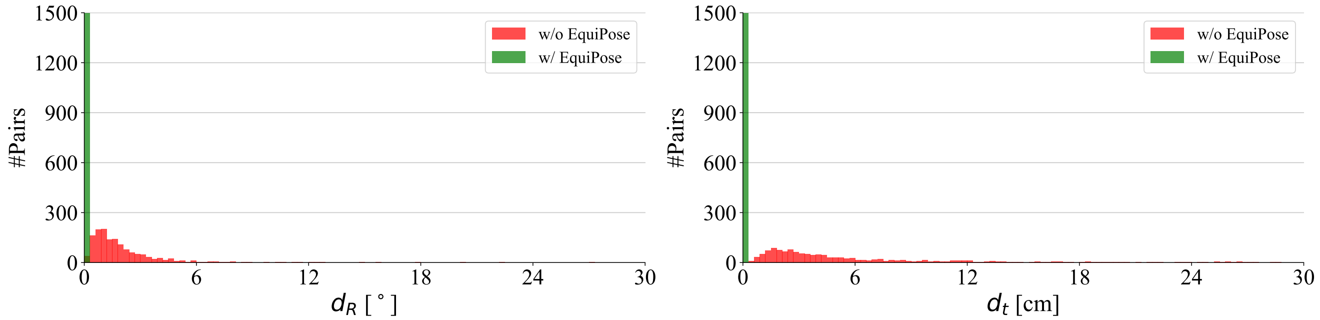
Method	**Mn d_R**	**STD d_R**	**Mn d_t**	**STD d_t**
w/o EquiPose	2.7	6.1	11.6	23.6
w/ EquiPose	0.0	0.0	0.0	0.0

(a) EightVit



Method	**Mn d_R**	**STD d_R**	**Mn d_t**	**STD d_t**
w/o EquiPose	7.2	16.4	17.8	31.4
w/ EquiPose	0.0	0.0	0.0	0.0

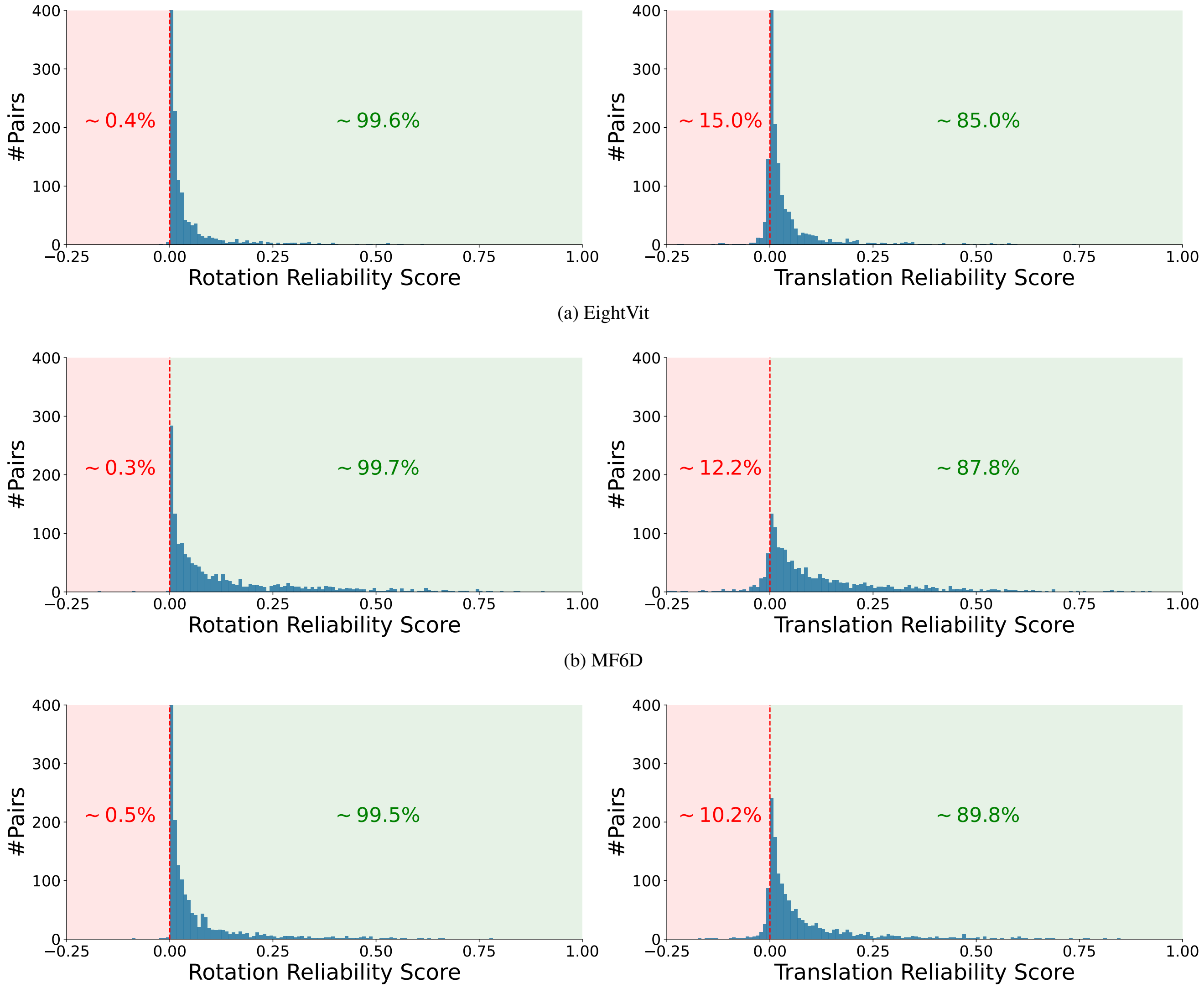
(b) MF6D



Method	**Mn d_R**	**STD d_R**	**Mn d_t**	**STD d_t**
w/o EquiPose	3.5	9.3	12.9	23.5
w/ EquiPose	0.0	0.0	0.0	0.0

(c) GRelPose

Figure S6. Distribution of the d_R and d_t estimated by existing models EightVit [46], MF6D [1], GRelPose [28], and their implementation under EquiPose. We also report the mean and standard deviation of d_R and d_t respectively of each method.



(a) EightVit

(b) MF6D

(c) GRelPose

Figure S7. Distribution of the **reliability scores** (see Sec. S8.3.2 for details) of EquiPose, evaluated on the ScanNet dataset across three baseline models: EightVit [46], MF6D [1], and GRelPose [28]. A **positive** reliability score indicates that EquiPose **improves** the reliability of the baseline model, while a negative reliability score indicates that EquiPose lowers the reliability of the baseline model. The percentages of samples with negative and positive reliability scores are also reported. In most cases, EquiPose achieves a positive reliability score, suggesting that EquiPose improves model reliability in most cases.

Case #2: The adopted input order results in a worse estimation than (or the same with) the reverse order.

This is to say, when comparing the performances of two methods on an image pair, one method might fall into Case #1, while the other method might fall into Case #2.

Generally, when the evaluation is conducted on a large dataset with hundreds or thousands of image pairs, this should not be considered as an issue: According to the law of large numbers, the input orders of these image pairs would not favor a specific method overall if the input order of image pairs is constructed randomly (actually, it is done in the main text of this work as well as other existing works [1, 19, 26, 28, 46]).

However, when the evaluation and analysis are conducted under a more fine-grained level, *i.e.*, on each individual image pair, it may hamper us to make a meaningful comparison. Consider the following case:

Given an image pair with input order of $\{I_1, I_2\}$, model A and model B have a rotation error of 5° and 4° respectively. When the input order is reversed to $\{I_2, I_1\}$, model

Baseline	**Rotation↑**				**Translation↑**			
	>0	>0.05	>0.1	>0.2	>0	>0.05	>0.1	>0.2
EightVit [46]	99.6	17.8	9.3	4.4	85.0	20.1	10.6	4.6
MF6D [1]	99.7	52.5	37.5	22.7	87.8	53.3	37.9	22.4
GRelPose [28]	99.5	32.7	19.1	9.9	89.8	38.9	24.5	12.0

Table S6. The percentage of reliability scores of EquiPose that are larger than a certain threshold over three baseline models, including EightVit [46], MF6D [1] and GRelPose [28].

A and model B have a rotation error of 6° and 80° respectively. This highlights a limitation of the current evaluation criterion, which only considers one input order: there is a probability of 50% to observe model A performing better than model B, and a probability of 50% to observe model B performing better than model A.

In real-world applications, where the optimal input order is unknown, model A would likely be preferred, since it on average (or in *expectation*, from a probabilistic perspective) produces an error of $\frac{5^\circ+6^\circ}{2} = 5.5^\circ$, while model B produces an error of $\frac{4^\circ+80^\circ}{2} = 42^\circ$, indicating that model A would be more reliable overall.

The above case and analysis prompts us to propose the *expectation error* as an alternative evaluation criterion on each individual image pair $\{I_1, I_2\}$:

$$E = (E_{12} + E_{21})/2 \tag{23}$$

where E_{12} and E_{21} are the rotatation (or translation) errors from input order $\{I_1, I_2\}$ and $\{I_2, I_1\}$ respectively.

Straightforwardly, a lower expectation error indicates that the model is more reliable, and a higher expectation error indicates that the model is less reliable.

S8.3.2 Fine-Grained Comparison under Expectation Error

To analyze the performance of EquiPose on each image pair, we consider the difference between the expectation errors of the baseline model and its EquiPose implementation as follows (named as the ***Reliability Score***):

$$S_\mathrm{R} = (E_\mathrm{b} - E_\mathrm{eq})/E_\mathrm{b} \in (-\infty, 1], \tag{24}$$

where E_b denotes the expectation rotation (or translation) error from the baseline model, E_eq denotes the expectation rotation (or translation) error when the baseline models are implemented under the proposed EquiPose framework. The denominator is used to compute the relative improvement. Note that since the proposed EquiPose intrinsically holds the PPE property, its expectation error is actually its true error.

According to the above definition, the reliability score has the following four properties:

- When $S_\mathrm{R} > 0$, *i.e.*, $E_\mathrm{b} > E_\mathrm{eq}$, EquiPose could improve the reliability of rotation (or translation) estimation of the baseline model.
- When $S_\mathrm{R} = 0$, *i.e.*, $E_\mathrm{b} = E_\mathrm{eq}$, EquiPose does not have an impact on the reliability of rotation (or translation) estimation.
- When $S_\mathrm{R} < 0$, *i.e.*, $E_\mathrm{b} < E_\mathrm{eq}$, EquiPose decreases the reliability of rotation (or translation) estimation of the baseline model.
- The magnitude of the reliability score S_R measures the reliability of EquiPose, *i.e.*, a larger positive S_R indicates EquiPose improves the estimation better, and vice versa.

We conduct the evaluation on the test set of ScanNet [11], which contains 1500 image pairs in total. The reliability scores of EquiPose over three baseline models (including EightVit [46], MF6D [1] and GRelPose [28]) are computed. Then, the histograms of the reliability scores over the three baseline models are plotted in Fig. S7. Besides, we also compute the percentage of image pairs whose reliability scores are larger than the thresholds {0, 0.05, 0.1, 0.2}, which are reported in Tab. S6.

As seen from Fig. S7 and Tab. S6, EquiPose could achieve a positive reliability score in most cases in terms of both rotation and translation, indicating EquiPose could improve the model performance generally. Moreover, in many cases the improvements are significant. For example, in the case of MF6D, EquiPose could achieve a relative improvement of more than 10% in 37.5% situations, and achieve a relative improvement of more than 20% in 22.7% situations.

The above results demonstrate that EquiPose has a high chance to improve the reliability of existing models, with notable improvements in many cases.