Deep Projective Rotation Estimation through Relative Supervision - Appendix

Contents

A Intuitive Example

We present an intuitive example of when optimizing a set of orientations to solve the rotation averaging problem described in Equation (1), in the main text, can fail. In this example, we show the benefits of the Iterative Modified Rodrigues Projective Averaging approach over the baseline approach. We show that, while both $SO(3)$ averaging and Iterative Modified Rodrigues Projective Averaging share a class of non-optimal critical points, in the projective case, these critical points are a subset of the problematic configurations for $SO(3)$ averaging.

6th Conference on Robot Learning (CoRL 2022), Auckland, New Zealand.

A.1 Examples of Critical Points

In this section, we analyze a class of critical points shared by both standard $SO(3)$ averaging and Iterative Modified Rodrigues Projective Averaging. For simplicity, we will examine the $N = 3$ rotation case, where $\mathcal{R} = \{R_1, R_2, R_3\}$ with relative rotations of $R_i^j := R_i R_j^\top$. As this is an iterative algorithm, we need to initialize our predicted rotations to some values $\hat{\mathcal{R}} = \{\hat{R}_1, \hat{R}_2, \hat{R}_3\}$. In this case, we initialize each predictions to $\hat{R}_i := R_i R_0 \exp\left(\left(\theta_0 + i\frac{2\pi}{N}\right)\omega_0\right)$ where R_0 is an arbitrary but constant rotational offset, ω_0 and θ_0 define an arbitrary, but constant axis and constant rotation, about which each initial estimate $\hat{R}_i$ is rotated an additional angle of θ_i. We find that if we use the previously described methods to update this initial configuration, under certain values of $\mathcal{R}$, R_0, θ_0, and ω_0, the expected update at each value $\hat{R}_i$ is $\mathbf{0}$, forming a critical point for each algorithm.

A.1.1 Critical Point for $SO(3)$ Averaging

Given the initial predictions of $\hat{\mathcal{R}}$ defined above, for all values of $\mathcal{R}$, R_0, θ_0, and ω_0, we find that the expectation of the gradient of $SO(3)$ averaging loss, $\mathbb{E}_{i,j}\left[\nabla_{\hat{r}_i}\mathcal{L}_{SO(3)}\right]$, is $\mathbf{0}$. The gradient of any sampled pair i, j is given by

$$\begin{aligned}
\nabla_i \mathcal{L}_{SO(3)}^{i,j} :&= \nabla_{\hat{r}_i}\mathcal{L}_{SO(3)}\left(\hat{R}_i, \hat{R}_j, R_i^j\right) \\
&= \log\left(\hat{R}_i^\top R_i^j \hat{R}_j\right) \\
&= \log\left(\left(R_i R_0 \exp\left(\theta_i\omega_0\right)\right)^\top R_i^j R_j R_0 \exp\left(\theta_j\omega_0\right)\right) \\
&= \log\left(\exp\left(\left(\theta_j - \theta_i\right)\omega_0\right)\right) \\
&= \text{wrap}_{[-\pi,\pi)}\left[\left(\theta_j - \theta_i\right)\right]\omega_0 \\
&= \text{wrap}_{[-\pi,\pi)}\left[\frac{2\pi}{N}(j-i)\right]\omega_0 \\
&= \frac{2\pi}{N}(j-i)\omega_0.
\end{aligned}$$

This lead to an expected gradient of each estimate rotation $\hat{R}_i$ of

$$\mathbb{E}_j\left[\nabla_{\hat{r}_i}\mathcal{L}_{SO(3)}\left(\hat{R}_i, \hat{R}_j, R_i^j\right)\middle| i = 1\right] = \frac{1}{2}\text{wrap}_{[-\pi,\pi)}\left[\sum_{j\neq i}\frac{2\pi}{N}(j-i)\right]\omega_0 = \mathbf{0}.$$

For all estimates $\hat{R}_i$, this sums to an integer multiple of $2\pi\omega_0$, which, due to the definition of the $SO(3)$ exponential map, wraps to $\mathbf{0}$.

A.1.2 Critical Point for Iterative Modified Rodrigues Projective Averaging

When optimizing using our Iterative Modified Rodrigues Projective Averaging method, we find that this configuration is only a critical point when the relative orientations between each pair of rotations are equal and opposite, i.e., $R_i^j = R_i^{k\top} \rightarrow R_i^j = \exp\left(\pm\frac{2\pi}{N}\omega_0\right)$ and the predicted orientations are initialized at identity: $R_0 = \mathbf{I}$. This only happens when the true orientations $\mathcal{R}$ are evenly spaced about an axis of rotations: $R_i := \exp\left(\left(\theta_0 - i\frac{2\pi}{N}\right)\omega_0\right)$, leaving only axis of rotation ω_0 and the constant angular offset θ_0 about that axis as free parameters.

As we are trying to update these rotations using a method compatible with stochastic gradient descent, we are concerned with the expectation of our update with respect to a sampled pair. In this case, the expected loss and update, defined in Equations 4c and 5 in the main paper, respectively, for any projected rotation $\hat{\psi}_i$ and its neighbor $\hat{\psi}_j$ is $\mathcal{L}_{\Psi+}^{i,j} := \left\|\hat{\psi}_i - \phi(q_i^j \otimes \phi^{-1}(\hat{\psi}_j))\right\|^2$ where q_i^j is the

quaternion associated with R_i^j. As all $\hat{\psi}_i$ are initialized to the identity, i.e., $\phi(q_I) = \mathbf{0}$ where q_I is the identity quaternion, we get

$$\mathcal{L}_{\Psi+}^{i,j} := \left\| -\phi^{-1}(q_i^j) \right\|^2 \qquad \nabla_i \mathcal{L}_{\Psi+}^{i,j} := -\phi^{-1}(q_i^j)$$

$$\mathcal{L}_{\Psi-}^{i,j} := \left\| -\phi^{-1}(-q_i^j) \right\|^2 \qquad \nabla_i \mathcal{L}_{\Psi-}^{i,j} := -\phi^{-1}(-q_i^j)$$

The relative rotations in this configuration are

$$R_i^j := \exp\left(\pm \frac{2\pi}{3} \omega_0 \right)$$

with relative quaternions $q_i^j := \begin{bmatrix} \cos(\frac{\pi}{3}) & \pm \sin(\frac{\pi}{3}) \omega_0 \end{bmatrix}$, which leads to

$$\phi(q_i^j) = \frac{\pm \sin(\frac{\pi}{3}) \omega_0}{1 + \cos(\frac{\pi}{3})} = \frac{\pm \omega_0}{\sqrt{3}} \qquad \phi(-q_i^j) = \frac{\mp \sin(\frac{\pi}{3}) \omega_0}{1 - \cos(\frac{\pi}{3})} = \pm \sqrt{3} \omega_0.$$

This results in the potential losses for the positive and negative antipodes of

$$\mathcal{L}_{\Psi+}^{i,j} = \|\phi(q_i^j)\| = \frac{1}{3} \qquad \mathcal{L}_{\Psi-}^{i,j} = \|\phi(-q_i^j)\| = 3$$

for all pairs of i, j. Selecting the minimum loss antipodes, we get gradients of

$$\nabla_i \mathcal{L}_{\Psi}^{i,j} = \frac{\mp 1}{\sqrt{3}} \omega_0 \qquad \nabla_i \mathcal{L}_{\Psi}^{i,j} = \frac{\pm 1}{\sqrt{3}} \omega_0,$$

for $j = i + 1$ and $j = i - 1$, respectively. The final expectation of the gradients with respect neighborhood sampling is

$$\mathbb{E}_j \left[\nabla_{\hat{\psi}_i} \mathcal{L}_{SO(3)}(\hat{\psi}_i, \hat{\psi}_j, R_i^j) | i = 1 \right] = \frac{1}{2} \sum_{j \neq i} \nabla_i \mathcal{L}_{\Psi}^{i,j} = \frac{1}{2} \left(\frac{1}{\sqrt{3}} \omega_0 - \frac{1}{\sqrt{3}} \omega_0 \right) = \mathbf{0}.$$

While this demonstrates that our method is not without critical points, even in this simple example, it shows that this configuration is only problematic when the true rotations are equally spaced around an axis of rotation, ω_0, and the estimates are initialized at identity. This compares very favorably to the $SO(3)$ algorithm, which can be in a critical point for any set of relative rotations, R_i^j, and with initialization that can vary with an additional arbitrary constant rotation R_0.

B Method Details

A full description of the $SO(3)$ Averaging and Iterative Modified Rodrigues Projective Averaging is shown in Algorithm 1 and Algorithm 2, respectively. In practice, we find $\gamma = 0.5$ and $\eta = 0.1$ to produce the best results.

Algorithm 1: $SO(3)$ Averaging

input : Initial estimates $\hat{\mathcal{R}} = \{\hat{R}_1 \dots \hat{R}_N\}$
input : Local neighborhood $\mathcal{N}_i$ for each rotation $\hat{R}_i$
input : Relative rotations $R_i^j,\ \forall j \in \mathcal{N}_i$
input : Learning Rate γ
output: Optimized set $\hat{R}_i \in \hat{\mathcal{R}}$

while *Not Converged* **do**
  Sample a rotation $\hat{R}_i \sim \hat{\mathcal{R}}$ to update
  Sample a neighbor $\hat{R}_j \sim \mathcal{N}_i$
  Compute optimal update r_Δ with Equation 3b in the main paper
  Apply update in $SO(3)$: $\hat{R}_i \leftarrow \hat{R}_i \exp(\gamma r_\Delta)$
end
return $\hat{\mathcal{R}}$

Algorithm 2: Iterative Modified Rodrigues Projective Averaging

input : Initial estimates $\hat{\Psi} = \{\hat{\psi}_1 \dots \hat{\psi}_N\}$
input : Local neighborhood $\mathcal{N}_i$ for each rotations $\hat{\psi}_i$
input : Relative rotations q_i^j for each $j \in \mathcal{N}_i$
input : Learning Rate γ
input : Max gradient η
output: Optimized set $\hat{\psi}_i \in \hat{\Psi}$

while *Not Converged* **do**
  Sample a projected rotation $\hat{\psi}_i$ to update
  Sample a neighbor $\hat{\psi}_j \sim \mathcal{N}_i$
  Update the projected rotation ψ_Δ with Equation 5 in the main paper)
  if *the magnitude of the update is larger than* η **then**
    Resize update to be of size η: $\psi_\Delta \leftarrow \eta \frac{\psi_\Delta}{\|\psi_\Delta\|}$
  end
  Apply update in MRP space $\hat{\psi}_i \leftarrow \hat{\psi}_i + \gamma\psi_\Delta$
end
return $\hat{\Psi}$

C 1DSfM Datasets

We report results on all structure from motions datasets available in the 1DSfM [1]. Each environment is tested with 5 random initializations and the estimated rotations are updated by each algorithm in batches of size 64, for 20K iterations. While Iterative Modified Rodrigues Projective Averaging, **MRP (Ours)** outperform all **PMG [2]** based methods, the direct **Quaternion** optimization regularly converges to relatively accurate local optima more quickly than ours, as shown in Table S3 and Figure S.1. That being said, our method converges to a more accurate final configuration for most datasets, with respect to mean relative error, Table S4, mean absolute error, Table S1, and median absolute error, Table S2. Our method, as well as the baselines, do not appear to perform well on the larger datasets. As a reminder, this algorithm is specifically designed for training deep learned methods, not for direct rotation optimization. When training deep learned methods, all of the weights are shared, allowing the network to use a single example to improve the accuracy of all rotations near that example. Additionally, we see poor performance on datasets with extremely large observation noise, specifically Gendarmenmarkt, whose median observation error is over 12 degrees. All dataset statistics can be found in Table S5. It should be noted that these datasets do not fully cover the orientation space, and tend to largely cover only variations in yaw. For results on datasets that represent full coverage of the orientation space, see the Uniformly Sampled Rotations dataset or the Neural Network Optimization dataset.

D Curriculum for Neural Network Optimization

While the **MRP (Ours)** was able to learn the orientations of a fixed set of rotation images, training results shown in Figure S.2, we find that a curriculum is required for any relatively supervised method to generalized to unseen orientation. This curriculum training involves starting with a initial base rotation. The model is rendered at this base rotation and a random rotation within 30° of this base rotation. This base rotation is initially sampled with $\theta = 30^\circ$ of a constant anchor orientation, until the average training angular error of the previous epoch drops below a given threshold, in this case, 5°. Once the error drops below this threshold, the angular range, θ, from which this base rotation is sampled is increased by 5°. This process is repeated, increasing the value of θ by 5° each time the error threshold is reached. We find that **MRP (Ours)** is able to complete the curriculum in a reasonable number of iterations, about 100K, achieving a median final pairwise accuracy of 5.19° over three training sessions. This test error is sampled from two random rotations across the $SO(3)$, differing from the training error, which are sampled based on the curriculum and are always, at most, 30° apart. The quaternion optimization method, **Quaternion**, stalls out at curriculum angle of 90°, achieving a final pairwise accuracy of 12.41° and the **4D PMG [2]** method never gets past the first

Dataset	*Mean Absolute Error (°)*							
	4D PGM [2]	6D PGM [3]	9D PGM [4]	Quat	MRP (Ours)	IRLS-GM [5]	IRLS-$\ell_{1/2}$ [6]	MLP [7]
Ellis Island	7.5	7.03	6.41	7.44	**5.59**	3.04	2.71	2.61
NYC Library	9.23	8.32	7.38	8.92	**6.03**	2.71	2.66	2.63
Piazza del Popolo	16.37	16.1	15.88	15.24	**10.03**	4.10	3.99	3.73
Madrid Metropolis	13.55	13.23	11.78	13	**11.25**	5.30	4.88	4.65
Yorkminster	9.13	8.34	7.48	8.56	**5.3**	2.60	2.45	2.47
Montreal Notre Dame	8.17	7.65	6.24	7.76	**4.02**	2.63	2.26	2.06
Tower of London	8.02	8.12	8.36	7.44	**5.58**	3.42	3.41	3.16
Notre Dame	8.71	7.96	7.03	8.55	**5.80**	2.63	2.26	2.06
Alamo	9.41	11.98	10.98	8.74	**6.42**	3.64	3.67	3.44
Gendarmenmarkt	66.41	73.7	68.29	**46.63**	48.82	39.24	39.41	44.94
Union Square	32.46	40.86	40.92	13.44	**10.22**	6.77	6.77	6.54
Vienna Cathedral	29.18	31.42	32.94	18.67	**13.60**	8.13	8.07	7.21
Roman Forum	63.23	64.85	60.51	**18.11**	55.65	2.66	2.69	2.62
Piccadilly	53.35	84.37	106.84	**26.29**	29.98	5.12	5.19	3.93
Trafalgar	121.93	124.18	125.15	**69.65**	91.67	-	-	-

Table S1: **Final Mean Absolute Rotation Error Results on 1DSfM [1] dataset.** Results on the left before the double lines are comparisons of local method after 20K iterations. Results on the right after the double lines are obtained from global methods which require optimizing over global set of relative orientations data at each step. Results associated sections with dashed line are not available from global methods [7].

Dataset	*Median Absolute Error (°)*							
	4D PGM [2]	6D PGM [3]	9D PGM [4]	Quat	MRP (Ours)	IRLS-GM [5]	IRLS-$\ell_{1/2}$ [6]	MLP [7]
Ellis Island	3.68	3.25	3.12	4.04	**2.96**	1.06	0.93	0.88
NYC Library	6.11	5.52	4.85	6.11	**4.04**	1.37	1.30	1.24
Piazza del Popolo	9.51	9.32	9.32	9.29	**6.12**	2.17	2.09	1.93
Madrid Metropolis	9.37	9.06	7.86	9.07	**6.99**	1.78	1.88	1.26
Yorkminster	6.44	5.77	4.56	6.11	**3.29**	1.59	1.53	1.45
Montreal Notre Dame	3.86	3.56	2.86	3.90	**2.30**	0.58	0.57	0.51
Tower of London	4.87	5.84	6.36	4.64	**3.59**	2.52	2.50	2.20
Notre Dame	4.39	3.73	3.09	4.48	**2.61**	0.78	0.71	0.67
Alamo	4.73	5.77	5.16	4.90	**3.48**	1.30	1.32	1.16
Gendarmenmarkt	64.08	71.57	62.9	**43.91**	45.92	7.07	7.12	9.87
Union Square	27.75	34.68	34.84	9.75	**6.85**	3.66	3.85	3.48
Vienna Cathedral	13.80	13.77	16.73	11.67	**6.34**	1.92	1.76	2.83
Roman Forum	53.78	62.46	57.71	**16.56**	41.95	1.58	1.57	1.37
Piccadilly	42.34	79.74	107.32	19.67	**15.09**	2.02	2.34	1.81
Trafalgar	126.71	129.57	130.45	**65.54**	89.09	-	-	-

Table S2: **Final Median Absolute Rotation Error Results on 1DSfM [1] dataset.** Results on the left before the double lines are comparisons of local method after 20K iterations. Results on the right after the double lines are obtained from global methods which require optimizing over global set of relative orientations data at each step. Results associated sections with dashed line are not available from global methods [7].

level of the curriculum, with a final error of $125.09°$. The full training progression of each method, over three random initialization each, can be seen in Figure S.3

One way this curriculum could be applied to captured data as follows: given a video, a curriculum could be established based on temporal proximity in the video. Choosing an arbitrary initial frame of the video as a anchoring frame, a curriculum can be generate by increasing temporal distance to neighboring frames until the entire video has been used in training.

E 3D Object Rotation Estimation via Relative Supervision from Pascal3D+ Images

E.1 Experimental Setup

Pascal3D+ [9] is a standard benchmark for categorical 6D object pose estimation from real images. We follow similar experimental settings as in [2, 4] for 3D object pose estimation from single images. Following [2, 4], we discard occluded or truncated objects and augment with rendered images from [10]. We report 3D object pose estimation via relative orientation supervision results on two object categories of Pascal3D+ image dataset: *sofa* and *bicycle*. We compare our method **MRP** with five baselines: **Quaternion**, **4D PMG** [2], **6D PMG** [3], **9D PMG** [4] and **10D PMG** [11].

Dataset	*Mean nAUC*				
	4D PGM [2]	6D PGM [3]	9D PGM [4]	Quat	MRP (Ours)
Ellis Island	22.56	24.07	25.02	15.05	**14.58**
NYC Library	28.53	31.12	32.07	18.20	**16.84**
Piazza del Popolo	37.36	44.18	43.98	25.13	**22.21**
Madrid Metropolis	35.91	38.49	39.15	**24.34**	24.48
Yorkminster	36.82	42.37	44.91	18.71	**18.43**
Montreal Notre Dame	33.97	37.54	40.37	17.69	**16.19**
Tower of London	39.98	45.99	49.54	**18.14**	18.85
Notre Dame	38.77	43.04	46.05	**20.78**	21.10
Alamo	39.87	49.08	50.22	**20.47**	22.05
Gendarmenmarkt	97.45	101.77	100.11	74.76	**71.39**
Union Square	77.22	87.01	89.76	**34.60**	46.20
Vienna Cathedral	72.25	81.07	83.48	**38.74**	42.94
Roman Forum	103.59	105.73	108.88	**52.05**	82.30
Piccadilly	115.83	123.41	126.16	**62.87**	78.31
Trafalgar	126.43	126.49	126.5	**108.19**	115.90

Table S3: Final Mean Normalized AUC on all 1DSfM [1] datasets after 20K iterations

Dataset	*Mean Relative Error (°)*				
	4D PGM [2]	6D PGM [3]	9D PGM [4]	Quat	MRP (Ours)
Ellis Island	12.21	11.49	10.37	11.87	**9.03**
NYC Library	14.29	12.94	11.51	13.67	**9.30**
Piazza del Popolo	21.91	21.24	20.64	20.74	**13.49**
Madrid Metropolis	20.43	19.84	17.85	19.62	**17.09**
Yorkminster	13.73	12.64	11.58	12.97	**8.35**
Montreal Notre Dame	12.5	11.59	9.58	11.93	**6.22**
Tower of London	12.41	12.24	12.44	11.56	**8.71**
Notre Dame	14.15	13.1	11.65	13.86	**9.66**
Alamo	14.23	17.47	15.75	13.17	**9.78**
Gendarmenmarkt	84.21	89.61	84.77	**60.25**	62.98
Union Square	44.44	55.4	55.94	19.98	**15.52**
Vienna Cathedral	41.8	45.62	44.18	26.64	**20.32**
Roman Forum	79.24	77.18	78.03	**25.04**	64.25
Piccadilly	74.25	105.15	122.06	**38.61**	46.21
Trafalgar	126.18	126.42	126.49	**81.28**	97.53

Table S4: Final Mean Relative Error (°) on all 1DSfM [1] datasets after 20K iterations

We use ResNet18 [12] as the model backbone to predict object rotation from single images. The model is supervised by the geodesic error between the induced relative orientation between the predicted absolute orientations for a pair of images, and the relative orientation between the ground truth absolute orientations for the image pair.

Specifically, **MRP** is supervised by the geodesic distance on the MRP manifold as described in Equations 4 and 5 in the main paper. **Quaternion** is supervised by quaternion geodesic distance as described in Section 4 in the main paper. While **4D/6D/9D/10D PMG** are supervised by the geodesic error derived from projective manifold gradients as in [2]. We use the same batch size of 20 as in [2, 4], and use Adam [13] with learning rate of 1e-4.

E.2 Result Analysis

Results for *sofa* showed in Figure S.4 and Table S6. Results for *bicycle* showed in Figure S.5 and Table S7. **Pascal3D+ Sofa.** For *sofa* category, as seen in Table S6, we find that after 50K training iterations, **MRP (Ours)** achieves a mean angular pairwise error of 14.09° on the test set, outperforms all other baselines. **Quaternion** achieves the worst error out of all methods, with final angular pairwise error of 26.35°. Besides achieving the lowest test angular error, we also find that **MRP (Ours)** has the fastest convergence speed, as seen in Figure S.4.

Dataset	# Nodes	# Edges	Mean Error	Median Error
Ellis Island	227	20K	12.52	2.89
NYC Library	332	21K	14.15	4.22
Piazza del Popolo	338	25K	8.4	1.81
Madrid Metropolis	341	24K	29.31	9.34
Yorkminster	437	28K	11.17	2.68
Montreal Notre Dame	450	52K	7.54	1.67
Tower of London	472	24K	11.6	2.59
Notre Dame	553	104K	14.16	2.7
Alamo	577	97K	9.1	2.78
Gendarmenmarkt	677	48K	33.33	12.3
Union Square	789	25K	9.03	3.61
Vienna Cathedral	836	103K	11.28	2.59
Roman Forum	1084	70K	13.84	2.97
Piccadilly	2152	309K	19.1	4.93
Trafalgar	5058	679K	8.64	3.01

Table S5: Dataset sizes and observation accuracies (°) for all 1DSfM [1] datasets

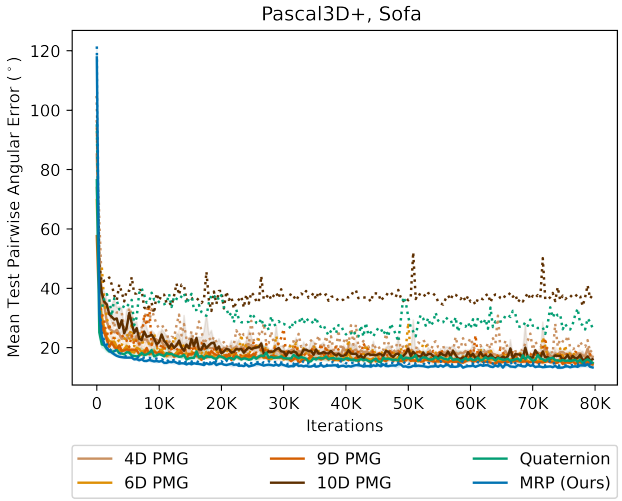

Figure S.4: **3D Object Pose Estimation via Relative Supervision on Pascal3D+ *Sofa* Images.** Mean test pairwise angular error in degrees of *sofa* at different iterations of training. Trained over 80K training steps for 8 random seeds per method. Solid lines stand for mean errors, dashed line stand for max errors, and shaded area represents error standard deviation.

Algorithm	Mean (°)	Min (°)	Max (°)
4D PMG [2]	17.39 ± 1.14	19.42	16.07
6D PMG [3]	15.20 ± 0.77	16.43	14.44
9D PMG [4]	14.61 ± 0.50	15.66	14.18
10D PMG [11]	19.28 ± 7.58	37.76	15.03
Quaternion	16.52 ± 4.12	26.57	14.38
MRP (Ours)	$\mathbf{13.63 \pm 0.78}$	**15.08**	**12.62**

Table S6: **Final Mean Test Angular Pairwise Error on Pascal3D+ *sofa* Images after 80K training iterations, over 8 random seeds.**

Pascal3D+ Bicycle. For *bicycle* category, as seen in Table S7, we find that after 50K training iterations, **MRP (Ours)** achieves a mean angular pairwise error of $29.21°$ on the test set, outperforms all other baselines. Besides achieving the lowest test angular error, we also find that **MRP (Ours)** has the fastest convergence speed, as seen in Figure S.5.

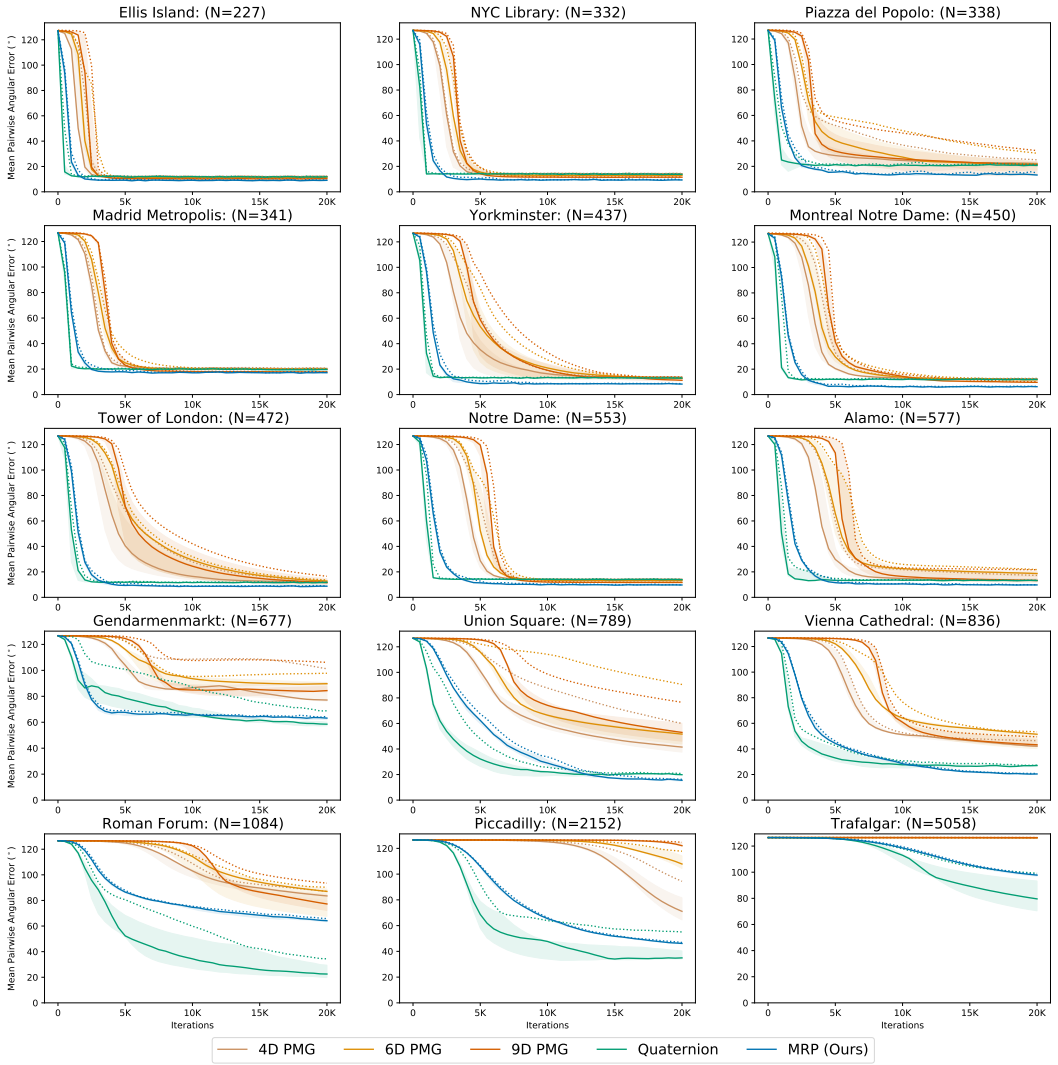

Figure S.1: Optimization results for all 1DSfM [1] datasets, ordered by number of cameras (N). Median average-pairwise angular error (°) is shown with shaded areas representing the first and third quartile over all training sessions. The max average-pairwise angular error for each algorithm at each iteration is shown as a dashed line.

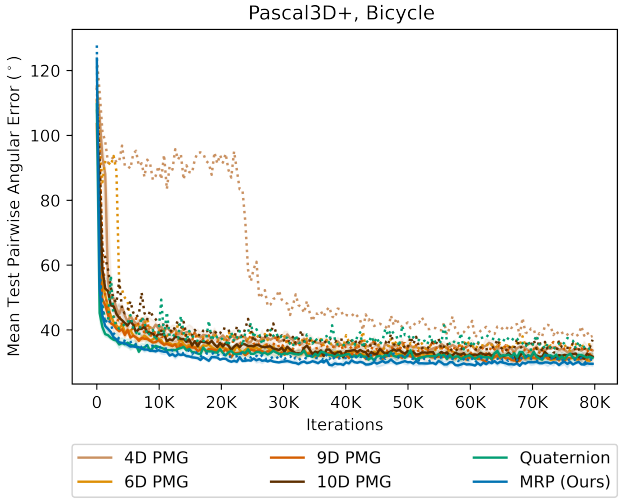

Figure S.5: **3D Object Pose Estimation via Relative Supervision on Pascal3D+ *Bicycle* Images.** Mean test pairwise angular error (°) of *bicycle* at different iterations of training. Trained over 80K training steps for 8 random seeds per method. Solid lines stand for mean errors, dashed line stand for max errors, and shaded area represents error standard deviation.

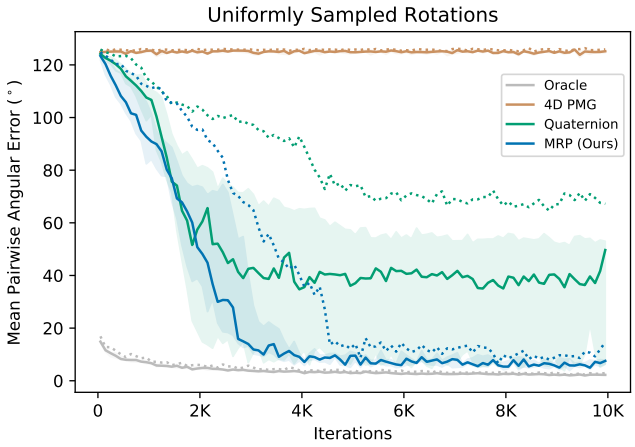

Figure S.2: Training results for rotations estimated by neural networks given images of the YCB drill [8] rendered at each of 100 random rotations with various supervisions.Median average-pairwise angular error (°) is shown with shaded areas representing the first and third quartile over all training sessions. The max average-pairwise angular error for each algorithm at each iteration is shown as a dashed line.

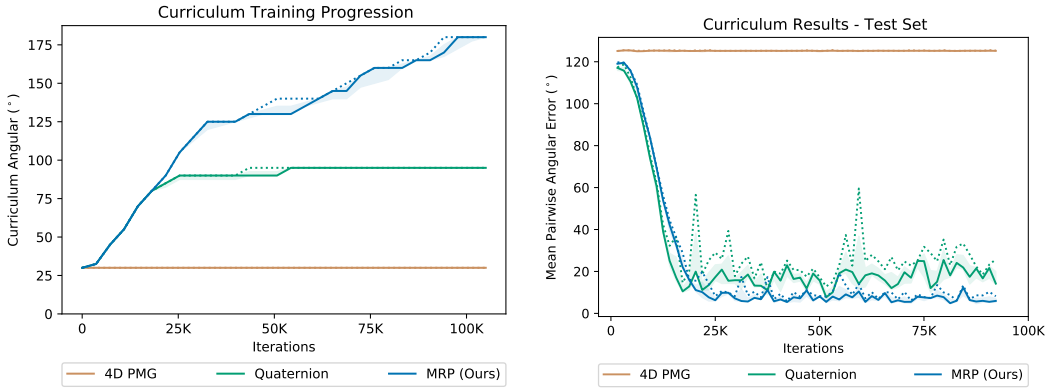

Figure S.3: Curriculum Angle (left) and Average Pairwise Error (right), sampled over the full orientation space for three training sessions with each method. Median average-pairwise angular error (°) is shown with shaded areas representing the first and third quartile over all training sessions. The max average-pairwise angular error for each algorithm at each iteration is shown as a dashed line.

Algorithm	Mean (°)	Min (°)	Max (°)
4D PMG [2]	34.57 ± 2.21	38.13	31.90
6D PMG [3]	31.58 ± 2.24	35.66	**28.42**
9D PMG [4]	31.80 ± 1.52	34.87	29.96
10D PMG [11]	32.23 ± 2.10	36.98	29.87
Quaternion	31.92 ± 1.00	33.61	30.61
MRP (Ours)	**29.46 ± 0.66**	**30.74**	28.62

Table S7: **Final Mean Test Angular Pairwise Error on Pascal3D+ *bicycle* Images after 80K training iterations, over 8 random seeds.**

F 3D Object Rotation Estimation via Relative Supervision from ModelNet40 Point Clouds

F.1 Experimental Setup

ModelNet40 [14] is a standard benchmark for categorical 6D object pose estimation from 3D point clouds. We follow similar experimental settings as in [2]. We follow the same train/test data split as in [2] and report 3D object pose estimation via relative orientation supervision results on the *airplane* category of ModelNet40 dataset. We compare our method **MRP** with four baselines: **Quaternion**, **4D PMG** [2], **6D PMG** [3], **9D PMG** [4] and **10D PMG** [11] . We use PointNet++ [15] as the model backbone to predict 3D absolute object rotation from single point cloud generated from the ModelNet40 3D CAD models, as in [2]. The model is supervised by the geodesic error between the induced relative orientation between the predicted absolute orientations for a pair of point clouds, and the relative orientation between the ground truth absolute orientations for the point cloud pair.

We sample 1024 points per point cloud as in [2, 4], use a batch size of 14. As for training, we use Adam [13] with learning rate of 1e-3, and run over 1 trial for each method.

We find that for any of the compared methods to generalize to unseen test point cloud instances, a curriculum is needed. We train with a curriculum over the rotation space, the curriculum details can be found in Section D. Specifically we start with base rotation range with $\theta = 30^\circ$ of a constant anchor orientation, and θ is increased by 5° whenever the previous mean epoch train angular error drops below the curriculum threshold, 5°. To speed up the training procedure, we increase this curriculum threshold to 8° once θ gets to 125°.

F.2 Result Analysis

Results on the *airplane* object class from ModelNet40 dataset is shown in Figure S.6 and Table S8.

As seen in Figure S.6 and Table S8, **MRP (Ours)** is able to go through the curriculum in 250K iterations, reaching final test pairwise angular error of 5.49°. **Quaternion** goes through the curriculum much slower, reaching curriculum angle $\theta = 90^\circ$ at the end of 250K steps. **4D PMG**, **6D PMG**, **9D PMG** and **10D PMG**, on the other hand, is not able to progress beyond the original curriculum angle of $\theta = 30^\circ$, reaching final test pairwise angular error around 35° after 200K iterations. In summary, **MRP (Ours)** achieves faster convergence rate than all baselines, and is able to achieve final test angular error on the order of 5° after progressing through the curriculum.

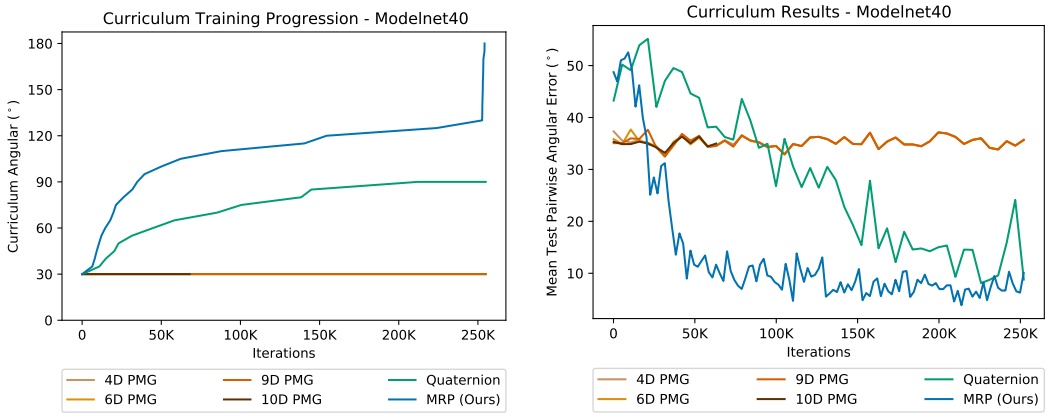

Figure S.6: **3D Object Rotation Estimation via Relative Supervision from ModelNet40 Point Clouds - *airplane*. Left:** Curriculum angle progression through training iterations. **Right:** Average test pairwise angular error (°), sampled over the full orientation space for 1 training session with each method.

Algorithm	Mean Test Angular Pairwise Error (°)
4D PMG [2]	35.35
6D PMG [3]	34.12
9D PMG [4]	35.80
10D PMG [11]	35.26
Quaternion	12.86
MRP (Ours)	**5.49**

Table S8: **Final Mean Test Angular Pairwise Error on ModelNet40 *airplane* Point Clouds after at most 250K training iterations.**

G Absolute Orientation Supervision

G.1 Experimental Setup

In this paper, we are assuming that only relative orientation supervision is available; however, in this section we explore how different orientation representations perform if absolute orientation supervision is available, and specifically how Modified Rodriguez Parameters (MRP) [16] used in this paper compare. To explore this, we perform an experiment on rotation estimation from 2D images of rendered YCB drill supervised with absolute orientation instead of relative supervision. We follow the same experimental setup as in Section 6.2 in the main paper, utilizing ResNet18 [12] as the model backbone to predict absolute 3D object orientations from sets of 2D rendered object images, rendered at 100 random rotations each. The neural network model is supervised by the geodesic error between the predicted absolute orientation and the ground truth absolute orientation. We compare the performance of different rotation parameterizations on this task. Specifically, we compare the Modified Rodriguez Parameters (MRP) [16] (**Oracle-MRP**) with Quaternions (**Oracle-Quaternion**). Each method is trained for 10K steps, over 8 different rendered image sets. We report the mean global pairwise angular error over the whole set of 100 images over the training process in Table S9.

G.2 Result Analysis

We report results on three metrics: 1) mean global train absolute angular error; 2) median global train absolute angular error; 3) percentage of runs that converge with final pairwise angular error $< 2^\circ$ after 10K steps, which is referred to as 2° *Acc*. Specifically, global relative angular error is calculated as the all-pair relative angular error for all pairs within the image set of 100. As see in Table S9, **Oracle-MRP** achieves comparable but larger mean and median pairwise angular error compared to **Oracle-Quaternion**, while both methods achieves the same 2° *Acc* of 87.5%. In summary, through this simple experiment, we find that MRP is able to achieve comparable but slightly worse train error for absolute orientation supervision compared to quaternions. Thus in the case of direct pose supervision, MRP may not be the best choice of rotation representation; using an open manifold such as in MRP is beneficial only in the case of relative pose supervision.

Algorithm	Mean Error (°)	Median Error (°)	2° Acc (%)
Oracle-Quaternion	**1.58**	**1.56**	**87.5**
Oracle-MRP	1.81	1.86	**87.5**

Table S9: **Absolute Orientation Supervision for Image Based Rotation Estimation from Rendered YCB Drill Images using MRP vs Quaternions Parametrization.** Final mean, median angular train error ($\circ$) and convergence ($< 2^\circ$) percentage for image based rotation estimation from rendered YCB drill images with absolute orientation supervision, after 10K training steps over 8 sets of 100 rendered images.

H Object Orientation Prediction Qualitative Visual Results

We further show some qualitative visual illustrations of the object orientation prediction of trained model at convergence, trained using our iterative MRP averaging method via relative orientation supervision below. Examples from orientation estimation on the rendered YCB drill data as described

in Section 6.2 in the main paper is shown in Figure S.7. Examples from orientation estimation on unseen Pascal3D+ *sofa* and *bicycle* categories, as described in Section E.1, are shown in Figure S.8.

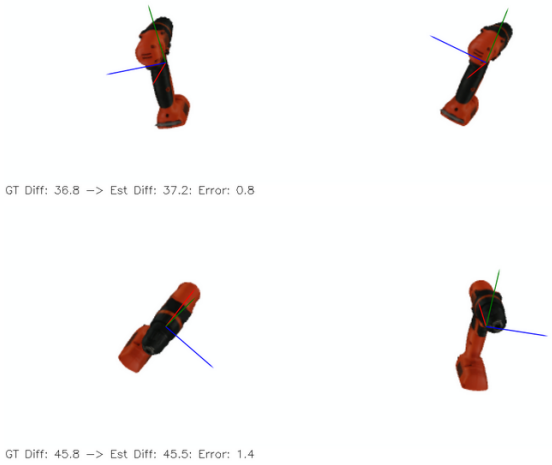

Figure S.7: **Qualitative Visual Examples for Object Orientation Estimation of MRP (Ours) on Rendered YCB Drill Images.** We show qualitative visual examples of predicted object 3D orientation from orientation prediction model trained via iterative MRP averaging with relative orientation supervision, the model is evaluated after training for 10K steps from neural net optimization experiment described Section 6.2 of the main paper. The predicted orientation is shown as coordinate frame (x, y, z). On the bottom of each example, we show in text of the ground truth relative orientation angular difference (°) between the pair of images, and their predicted relative orientation angular difference (°) induced from the absolute object orientation predicted for each image. And finally we show the difference between the predicted relative angular difference and the ground truth relative angular difference as angular error (°).

Figure S.8: **Qualitative Visual Examples for Object Orientation Estimation of MRP (Ours) on Unseen Pascal3D+ Images.** We show qualitative visual examples of predicted object 3D orientation from orientation prediction model trained via iterative MRP averaging with relative orientation supervision on *Sofa* (left) and *Bicycle* (right) images. The model is evaluated after training for 50K steps from 3D object rotation estimation on Pascal3D+ experiment as described Section E. The predicted orientation is shown as coordinate frame (x, y, z). On the bottom of each example, we show in text of the ground truth relative orientation angular difference (°) between the pair of images, and their predicted relative orientation angular difference (°) induced from the absolute object orientation predicted for each image. And finally we show the difference between the predicted relative angular difference and the ground truth relative angular difference as angular error (°).

References

[1] K. Wilson and N. Snavely. Robust global translations with 1dsfm. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014.

[2] J. Chen, Y. Yin, T. Birdal, B. Chen, L. Guibas, and H. Wang. Projective manifold gradient layer for deep rotation regression. *arXiv preprint arXiv:2110.11657*, 2021.

[3] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li. On the continuity of rotation representations in neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5745–5753, 2019.

[4] J. Levinson, C. Esteves, K. Chen, N. Snavely, A. Kanazawa, A. Rostamizadeh, and A. Makadia. An analysis of svd for deep rotation estimation. *Advances in Neural Information Processing Systems*, 33:22554–22565, 2020.

[5] A. Chatterjee and V. M. Govindu. Efficient and robust large-scale rotation averaging. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 521–528, 2013.

[6] A. Chatterjee and V. M. Govindu. Robust relative rotation averaging. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):958–972, 2017.

[7] Y. Shi and G. Lerman. Message passing least squares framework and its application to rotation synchronization. In *International Conference on Machine Learning*, pages 8796–8806. PMLR, 2020.

[8] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. 2018.

[9] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *IEEE winter conference on applications of computer vision*, pages 75–82. IEEE, 2014.

[10] H. Su, C. R. Qi, Y. Li, and L. J. Guibas. Render for cnn: Viewpoint estimation in images using cnns trained with rendered 3d model views. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2686–2694, 2015.

[11] V. Peretroukhin, M. Giamou, D. M. Rosen, W. N. Greene, N. Roy, and J. Kelly. A smooth representation of belief over so(3) for deep rotation learning with uncertainty. *In Proceedings of Robotics: Science and Systems (RSS'20)*, 2020.

[12] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[13] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In *ICLR (Poster)*, 2015.

[14] Z. Wu, S. Song, A. Khosla, F. Yu, L. Zhang, X. Tang, and J. Xiao. 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920, 2015.

[15] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[16] J. Crassidis and F. Markley. Attitude estimation using modified rodrigues parameters. In *Flight Mechanics/Estimation Theory Symposium*, pages 71–86. NASA, 1996.