HeMoRa: Unsupervised Heuristic Consensus Sampling for Robust Point Cloud Registration

Supplementary Material

A. Derivation of Eq. (5) and Eq. (6)

A.1. Derivation of Eq. (5)

The optimization problem, as defined in Eq. (4) (main paper), is given by:

$$\Theta^\star = \underset{\Theta}{\operatorname{argmax}}\ \mathbb{E}_{q\sim\hat{\mathbf{P}}^q}[R^q]. \tag{1}$$

To optimize $\mathbb{E}_{q\sim\hat{\mathbf{P}}^q}[R^q]$ using policy gradients, it is necessary to compute its gradient with respect to Θ. Applying the property $\frac{\partial}{\partial\Theta}p(x;\Theta) = p(x;\Theta)\frac{\partial}{\partial\Theta}\ln p(x;\Theta)$, we derive:

$$\frac{\partial}{\partial\Theta}\mathbb{E}_{q\sim\hat{\mathbf{P}}^q}[R^q] = \mathbb{E}_{q\sim\hat{\mathbf{P}}^q}\left[R^q\frac{\partial}{\partial\Theta}\ln\hat{\mathbf{P}}^q\right]. \tag{2}$$

The expectation is approximated by drawing Q samples:

$$\frac{\partial}{\partial\Theta}\mathbb{E}_{q\sim\hat{\mathbf{P}}^q}[R^q] \approx \frac{1}{Q}\sum_{q=1}^{Q}R^q\frac{\partial}{\partial\Theta}\ln(\hat{\mathbf{P}}^q). \tag{3}$$

A.2. Derivation of Eq. (6)

Starting from Eq. (5) (main paper) and substituting Eq. (3) (main paper), the derivation proceeds as follows:

$$\begin{aligned}
\frac{\partial}{\partial\Theta}\mathbb{E}_{q\sim\hat{\mathbf{P}}^q}[R^q] &\approx \frac{1}{Q}\sum_{q=1}^{Q}R^q\frac{\partial}{\partial\Theta}\ln(\hat{\mathbf{P}}^q) \\
&\approx \frac{1}{Q}\sum_{q=1}^{Q}R^q\frac{\partial}{\partial\Theta}\ln\left(\prod_{k=1}^{K}\mathbf{P}_k^q\right) \\
&= \frac{1}{Q}\sum_{q=1}^{Q}R^q\frac{\partial}{\partial\Theta}\sum_{k=1}^{K}\ln\left(\mathbf{P}_k^q\right) \\
&= \frac{1}{Q}\sum_{q=1}^{Q}R^q\frac{\partial}{\partial\Theta}\sum_{i=1}^{N}\mathbb{I}_i^q\ln(\mathbf{P}_i) \\
&= \frac{\partial}{\partial\Theta}\sum_{i=1}^{N}\ln(\mathbf{P}_i)\cdot\frac{1}{Q}\sum_{q=1}^{Q}\mathbb{I}_i^qR^q.
\end{aligned} \tag{4}$$

B. MoRa vs. Supervised Loss in PointDSC

We analyze the similarities and advantages of the MoRa loss compared to the supervised losses used in PointDSC [1]. This comparison demonstrates that MoRa loss provides a unified formulation that does not rely on explicit transformation labels.

B.1. First-order MoRa vs. Node-wise Supervision

Node-wise Supervision ($\mathcal{L}_{\text{class}}$). The node-wise supervision loss $\mathcal{L}_{\text{class}}$ in PointDSC [1] is formulated using binary cross-entropy (BCE):

$$\begin{aligned}
\mathcal{L}_{class} &= \text{BCE}(\mathbf{P}, \boldsymbol{w}^*) \\
&= -\frac{1}{N}\sum_{i=1}^{N}\left[\boldsymbol{w}_i^*\log(\mathbf{P}_i) + (1-\boldsymbol{w}_i^*)\log(1-\mathbf{P}_i)\right],
\end{aligned} \tag{5}$$

where $\mathbf{P}$ is the predicted confidence of correspondences. $\boldsymbol{w}^*$ represents ground-truth labels, indicating whether a correspondence $\boldsymbol{c}_i$ is an inlier (1) or outlier (0). The loss for an individual correspondence is given by:

$$[\mathcal{L}_{\text{class}}]_i = \begin{cases} -\log(\mathbf{P}_i), & \text{if } \boldsymbol{w}_i^* = 1 \\ -\log(1-\mathbf{P}_i), & \text{otherwise} \end{cases}. \tag{6}$$

Optimizing $\mathcal{L}_{\text{class}}$ increases the confidence of inliers and reduces that of outliers.

Our MoRa$^{(1)}$. The first-order MoRa loss is defined as:

$$\mathcal{L}^{(1)} = -\sum_{i=1}^{N}\log(\mathbf{P}_i)\cdot\mathbf{E}_i, \tag{7}$$

where $\mathbf{E}_i$ represents the potential energy of correspondence $\boldsymbol{c}_i$. Typically, inliers have higher energy values than outliers. The loss for an individual correspondence is:

$$[\mathcal{L}^{(1)}]_i = -\log(\mathbf{P}_i)\cdot\mathbf{E}_i. \tag{8}$$

By optimizing $\mathcal{L}^{(1)}$, correspondences with higher potential energy (primarily inliers) are prioritized, naturally improving their sampling probabilities and confidence values.

B.2. Second-order MoRa vs. Edge-wise Supervision

Edge-wise Supervision ($\mathcal{L}_{\text{sm}}$). The edge supervision loss $\mathcal{L}_{\text{sm}}$ in PointDSC [1] is defined as:

$$\mathcal{L}_{\text{sm}} = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\left(\gamma_{ij} - \gamma_{ij}^*\right)^2, \tag{9}$$

where $\gamma_{ij}^* = \boldsymbol{w}_i^*\cdot\boldsymbol{w}_j^*$ indicates whether both correspondences $\boldsymbol{c}_i$ and $\boldsymbol{c}_j$ are inliers. γ_{ij} is the predicted feature similarity between $\boldsymbol{c}_i$ and $\boldsymbol{c}_j$, computed as:

$$\gamma_{ij} = \max\left(1 - \frac{1}{\sigma_f^2}\|\mathbf{F}_i - \mathbf{F}_j\|^2, 0\right), \tag{10}$$

where $\mathbf{F}_i$ and $\mathbf{F}_j$ are feature descriptors, and σ_f is a parameter to control sensitivity to feature difference [1]. This loss encourages higher feature similarity between inliers while reducing similarity between outliers.

Our MoRa$^{(2)}$. The second-order MoRa loss is defined as:

$$\mathcal{L}^{(2)} = -\sum_{i=1}^{N}\sum_{j=1}^{N} \log(\mathbf{P}_{ij}^{(2)}) \cdot \mathbf{E}_{ij}^{(2)}, \tag{11}$$

where $\mathbf{E}_{ij}^{(2)}$ is the potential energy of a correspondence pair $\{\boldsymbol{c}_i, \boldsymbol{c}_j\}$, which is typically higher for inlier pairs. By optimizing $\mathcal{L}^{(2)}$, inlier pairs with higher potential energy are prioritized, enhancing feature similarity in line with the objectives of $\mathcal{L}_{\text{sm}}$.

B.3. Summary of Key Findings

1. Alignment with PointDSC Objectives:
   - MoRa$^{(1)}$ aligns with $\mathcal{L}_{class}$ when optimizing correspondence sampling probabilities.
   - MoRa$^{(2)}$ aligns with $\mathcal{L}_{sm}$ when optimizing feature similarity.
2. Generalized Formulation:
   - MoRa provides a unified optimization framework, integrating both node and edge objectives.
   - **MoRa eliminates the need for explicit transformation labels, enabling unsupervised learning.**

C. Additional Implementation Details

C.1. Second-order MoRA Loss Computation

The MoRA$^{(2)}$ loss (Eq. 12 in the main paper) requires defining $\mathbf{P}_{i,j}^{(2)}$, which quantifies the probability of sampling the correspondence pair $\{\boldsymbol{c}_i, \boldsymbol{c}_j\}$ simultaneously. This probability is obtained by applying dual-softmax to feature similarity [2, 5], reflecting the intuition that correspondences with higher similarity are more likely to co-occur in sampling. Given feature embeddings $\mathbf{F} \in \mathbb{R}^{N\times D}$ where D is the feature dimension, the computation is as follows:

1. Compute the similarity matrix:

$$\mathbf{S} = \mathbf{F}\mathbf{F}^T. \tag{12}$$

2. Apply dual-softmax on $\mathbf{S}$ to achieve probabilistic normalization. This produces two matrices, $\mathbf{S}^{\text{row}} \in \mathbb{R}^{N\times N}$ and $\mathbf{S}^{\text{col}} \in \mathbb{R}^{N\times N}$, defined as:

$$\mathbf{S}_{i,:}^{\text{row}} = \text{Softmax}(\mathbf{S}_{i,:}), \quad \mathbf{S}_{:,j}^{\text{col}} = \text{Softmax}(\mathbf{S}_{:,j}). \tag{13}$$

3. Combine the row-wise and column-wise softmax results, we get:

$$\mathbf{P}^{(2)} = \mathbf{S}^{\text{row}} \cdot \mathbf{S}^{\text{col}}. \tag{14}$$

C.2. Reward Score Normalization

To evaluate the quality of sampled correspondences, we use the inlier ratio as the reward signal, as discussed in L226–L227 of the main paper. A higher inlier ratio indicates better sampling quality. However, its absolute value varies significantly across point cloud pairs, introducing bias: pairs with inherently higher-quality correspondences (i.e., higher inlier ratios under the ground-truth transformation) naturally achieve better scores, thereby overshadowing those with lower-quality correspondences. To mitigate this imbalance, we normalize the reward as:

$$\hat{R}^q = \begin{cases} \left(\frac{R^q - \tau^{IR}}{\max(R^q) - \tau^{IR}}\right)^{\alpha}, & \text{if } R^q \geq \tau^{IR} \\ 0, & \text{otherwise} \end{cases}, \tag{15}$$

where τ^{IR} is the inlier ratio threshold and α controls the sharpness of the normalization.

C.3. Details of Our Subset Sampler

We provide additional implementation details regarding the subset sampling strategies employed in our method.

Ours+PSAC. Following the RANSAC [4] paradigm, we execute 4 million iterations [6]. In each iteration, three correspondences are randomly sampled from the full set of correspondences $\mathcal{C}$, guided by our sampling probability distribution $\mathbf{P}$. This strategy ensures that correspondences with higher confidence scores are more likely to be selected.

Ours+SM. SM refers to the Seed Mechanism in PointDSC. This approach begins by identifying reliable and spatially well-distributed correspondences, designated as seeds, guided by $\mathbf{P}$. Subsequently, correspondences that are consistent with these seeds are selected based on feature similarity.

Ours+SC^2. SC^2-PCR [3] performs subset sampling by first selecting correspondence seeds based on the leading eigenvector of the similarity matrix, followed by an SC^2-based consensus set search. In contrast, our method deviates from this approach by not using the leading eigenvector for sampling, instead leveraging our $\mathbf{P}$ for correspondence seeds selection.

C.4. Details of Evaluation Metrics

To thoroughly assess the performance of our proposed method, we employ the following evaluation metrics:

Translation Error (TE). Translation Error quantifies the deviation of the estimated translation vector $\hat{\mathbf{t}}$ from the ground truth translation vector $\mathbf{t}^*$. It is computed as:

$$\text{TE}(\hat{\mathbf{t}}) = \|\hat{\mathbf{t}} - \mathbf{t}^*\|_2, \tag{16}$$

where $\|\cdot\|_2$ represents the Euclidean norm.

Rotation Error (RE). Rotation Error measures the angular discrepancy between the estimated rotation matrix $\hat{\mathbf{R}}$ and

	Dataset	RANSAC	**Ours+PSAC**	PointDSC	**Ours+SM**	SC^2-PCR	**Ours+SC^2**
FPFH	3DMatch	66.10(3123ms)	72.21(265ms)	77.57(98ms)	81.39(98ms)	83.72(64ms)	83.9(132ms)
	KITTI	74.41(4321ms)	83.39(271ms)	98.31(84ms)	99.13(84ms)	98.93(71ms)	99.13(141ms)
FCGF	3DMatch	91.44(3081ms)	92.79(267ms)	92.85(98ms)	93.41(98ms)	93.28(68ms)	93.65(129ms)
	KITTI	80.36(4471ms)	95.79(271ms)	97.66(84ms)	98.1(84ms)	97.76(71ms)	98.1(142ms)

Table 1. RR (%) and running time (ms) comparison.

the ground truth rotation matrix $\mathbf{R}^*$. It is defined as:

$$\mathrm{RE}(\hat{\mathbf{R}}) = \arccos\left(\frac{\operatorname{Tr}\left(\hat{\mathbf{R}}^T\mathbf{R}^*\right) - 1}{2}\right), \tag{17}$$

where $\operatorname{Tr}(\cdot)$ denotes the trace of a matrix.

Registration Recall (RR). Registration Recall evaluates the proportion of successful registrations, where success is defined as achieving both RE and TE below predefined thresholds. To ensure robustness against randomness, each registration process is repeated 20 times, with the average performance reported as the final metric.

D. Inference efficiency

Tab. 1 reports running time and RR results on four datasets. Ours+PSAC surpasses RANSAC in both speed and RR, and Ours+SM outperforms PointDSC in RR and matches its speed. Though Ours+SC^2 is slightly slower than SC2PCR, it achieves better RR.

E. Qualitative Results

In this section, we present cases where comparative methods fail to achieve successful registrations, while our method consistently produce highly accurate results, as illustrated in Figs. 1-3.

References

[1] Xuyang Bai, Zixin Luo, Lei Zhou, Hongkai Chen, Lei Li, Zeyu Hu, Hongbo Fu, and Chiew-Lan Tai. Pointdsc: Robust point cloud registration using deep spatial consistency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15859–15869, 2021. 1, 2

[2] Axel Barroso-Laguna, Sowmya Munukutla, Victor Adrian Prisacariu, and Eric Brachmann. Matching 2d images in 3d: Metric relative pose from metric correspondences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4852–4863, 2024. 2

[3] Zhi Chen, Kun Sun, Fan Yang, and Wenbing Tao. Sc2-pcr: A second order spatial compatibility for efficient and robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13221–13231, 2022. 2

[4] 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. 2

[5] 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

[6] Xiyu Zhang, Jiaqi Yang, Shikun Zhang, and Yanning Zhang. 3d registration with maximal cliques. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17745–17754, 2023. 2

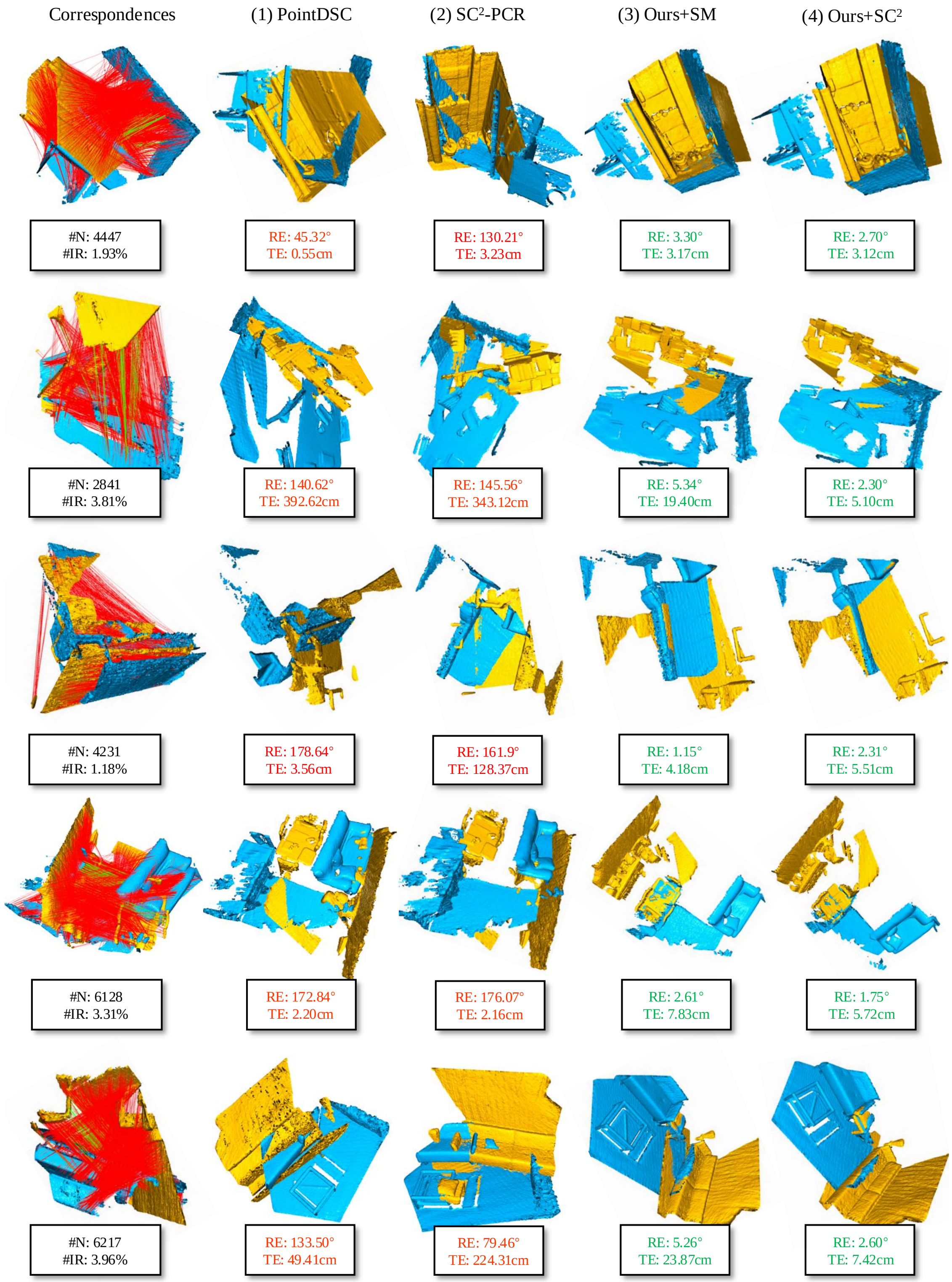

Figure 1. Qualitative comparison on 3DMatch. Red and green represent failed and successful registrations, respectively.

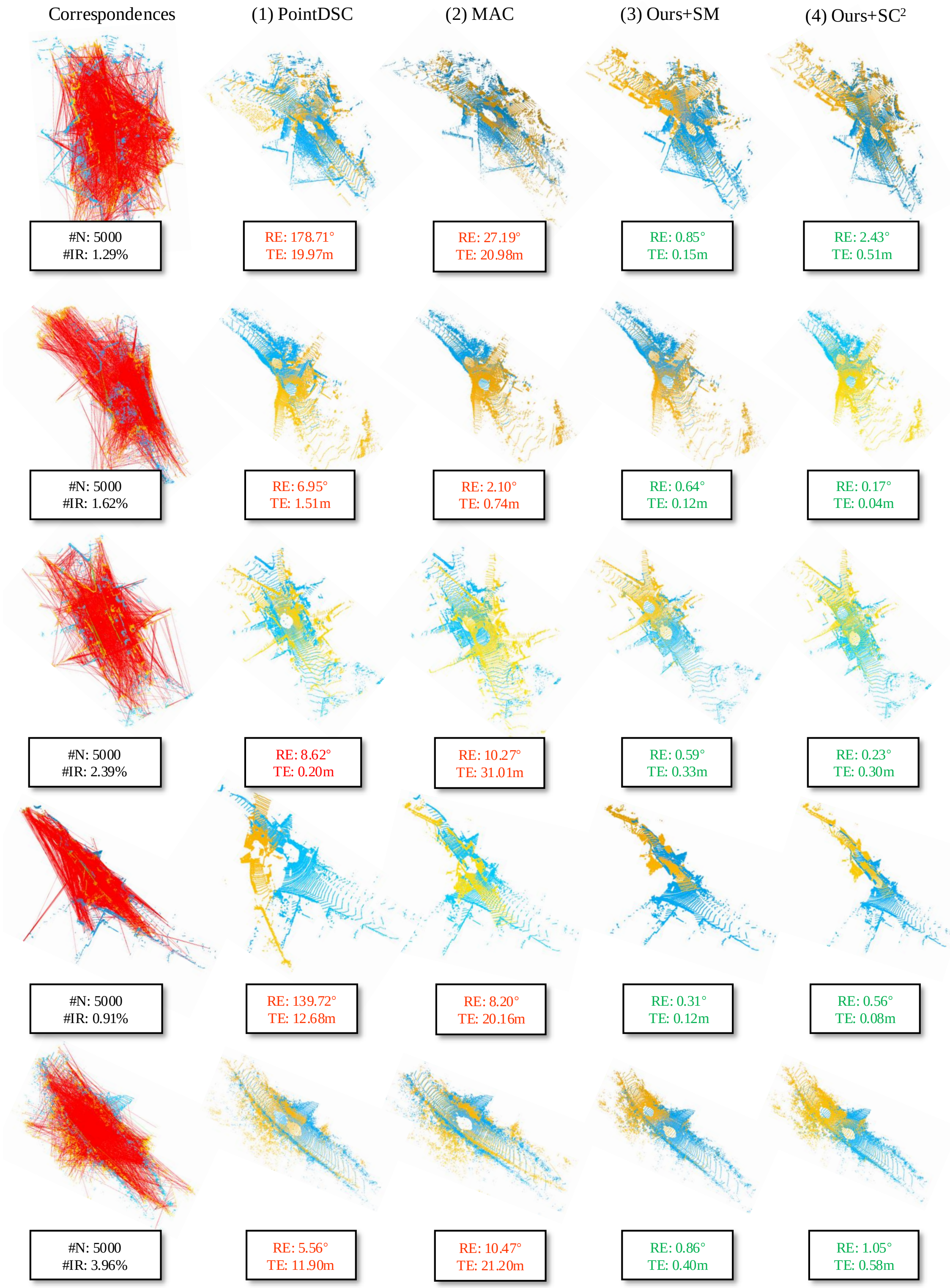

Figure 2. Qualitative comparison on KITTI-20m. Red and green represent failed and successful registrations, respectively.

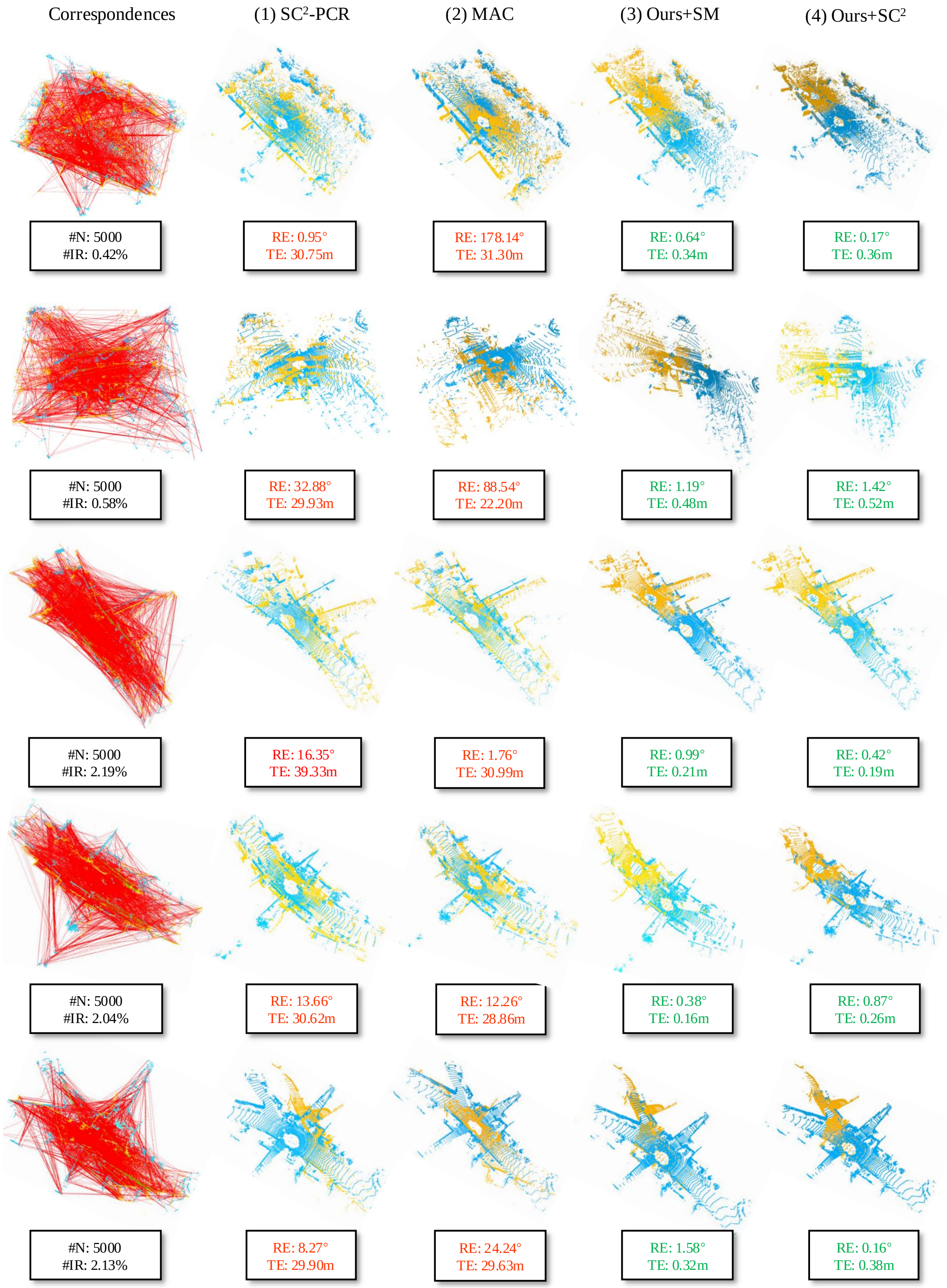

Figure 3. Qualitative comparison on KITTI-30m. Red and green represent failed and successful registrations, respectively.