Rethinking Point Cloud Registration as Masking and Reconstruction Supplementary Material

Anonymous ICCV submission

Paper ID 2164

Table 1: Performance on the 3DMatch and 3DLoMatch benchmarks for various depths of the MRA decoder.

Depth	3DMatch			3DLoMatch		
	RR(%)	RRE(°)	RTE(m)	RR(%)	RRE(°)	RTE(m)
1	95.1	1.324	0.043	75.4	2.488	0.072
2	94.6	1.385	0.042	72.8	2.560	0.072
3	94.5	1.390	0.045	71.9	2.471	0.068
4	94.4	1.417	0.045	73.3	2.546	0.073
5	94.5	1.391	0.044	72.9	2.510	0.070
6	94.2	1.432	0.045	72.8	2.473	0.073

Table 2: Performance on the 3DMatch and 3DLoMatch benchmarks for various dimensions of the MRA decoder.

Dim	3DMatch			3DLoMatch		
	RR(%)	RRE(°)	RTE(m)	RR(%)	RRE(°)	RTE(m)
128	94.5	1.353	0.043	73.1	2.447	0.073
256	95.1	1.324	0.043	75.4	2.488	0.072
384	94.6	1.373	0.044	74.1	2.552	0.072
512	94.9	1.435	0.045	74.8	2.533	0.071
768	94.6	1.410	0.045	73.0	2.549	0.074
1024	94.5	1.423	0.045	72.8	2.689	0.081

A. Additional Ablation Studies

The MRA decoder is only used to perform the point cloud reconstruction task during training. Hence, the MRA decoder architecture can be designed in a flexible manner that is independent of the encoder design. In this section, we experiment with various decoder architectures.

Table 1 presents variations in the decoder depth, measured by the number of transformer blocks, while setting the dimensions to 256. In contrast to the findings of MAE [7], we observe that the 1-layer MRA decoder exhibits superior performance over deeper decoders in terms of registration accuracy. We hypothesize that a shallow decoder can more efficiently guide the contextual features in the backbone to capture overall structures, which is crucial for point cloud registration. Additionally, the similarity between point cloud registration and point cloud reconstruction tasks renders it unnecessary to increase the depth of the decoder to bridge the gap between the tasks.

Furthermore, the effects of varying the decoder width in a 1-layer MRA decoder are investigated in Table 2. The decoder width is measured by the number of channels, and the default setting of 256 dimensions achieves promising performance on both 3DMatch and 3DLoMatch benchmarks.

In conclusion, our approach employs a 1-layer MRA decoder with a width of 256 dimensions, which not only achieves outstanding performance in accuracy but also requires modest training resource.

B. Efficiency Evaluation

The inference time of comparison methods is presented on the 3DMatch [17] benchmark. The experiments are conducted on a desktop computer with an Intel I7-10700 CPU and an Nvidia RTX 3090 GPU. The comparison methods include 3DSN [6], FCGF [5], CG-SAC [13], D3Feat [1], DGR [4], PCAM [2], OMNet [14], DHVR [9], Predator [8], CoFiNet [16], RegTR [15], Lepard [10], SC^{2}PCR [3], and GeoTransformer (GeoTR) [12]. As shown in Table 3, our MRT achieves a promising inference time of under 110 ms, which is feasible for many real-time applications. While OMNet may exhibit higher efficiency, it suffers from inaccurate point cloud alignments. Similarly, RegTR underperforms in terms of accuracy and robustness. In comparison to GeoTR, our approach also demonstrates superior performance in both efficiency and accuracy. In general, the MRA network effectively guides the backbone network, resulting in a significant improvement in registration accuracy without incurring additional computational complexity during the inference process. This enables our method to achieve promising performance in both accuracy and efficiency.

C. Reconstruction results

The reconstruction results obtained on the 3DMatch, ModelNet, and KITTI datasets are shown in Fig. 1, Fig. 2, and Fig. 3, respectively, demonstrating the effective recovery of invisible parts by our method. Specifically, our

Table 3: Computational time in seconds on the 3DMatch benchmark.

3DSN	FCGF	CG-SAC	D3Feat	DGR	PCAM	OMNet	DHVR	Predator	CoFiNet	RegTR	Lepard	SC^{2}PCR	GeoTR	Ours
30.234	1.562	0.263	0.916	1.741	1.786	0.012	3.43	1.572	1.134	0.103	0.522	0.380	1.523	0.106

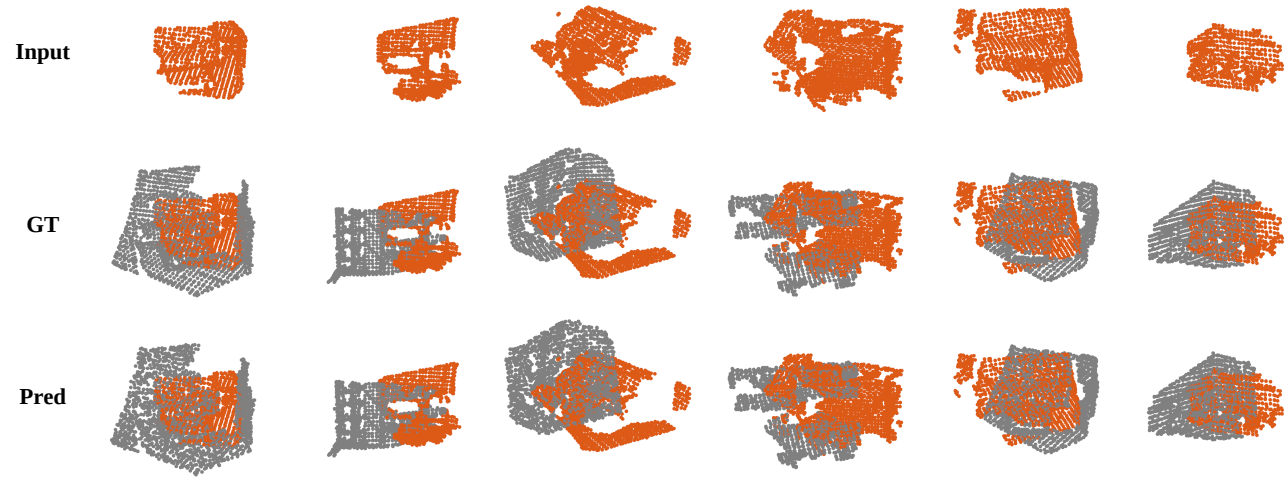

Figure 1: Reconstruction results obtained on the 3DMatch dataset.

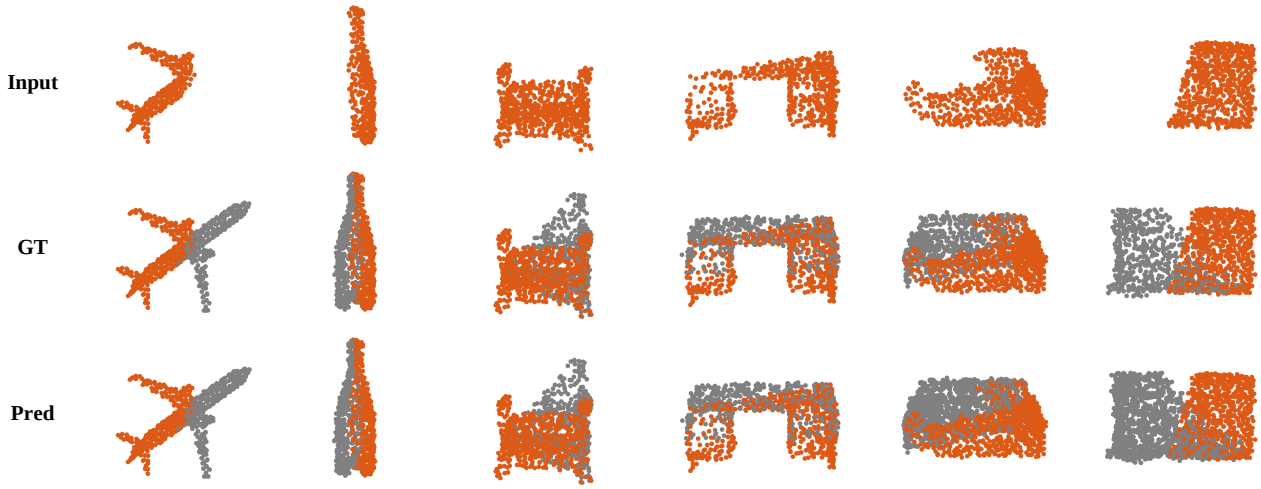

Figure 2: Reconstruction results obtained on the ModelNet dataset.

method effectively recovers the overall structures of the point cloud pair while accurately predicting the fine-grained geometric details of the invisible regions inferred from the corresponding regions in the other point cloud. These results verify that the reconstruction auxiliary task conducted by the MRA guides the contextual features in the backbone to capture the overall structures and the geometric details of point cloud pairs, thus boosting the registration accuracy of our method. However, some point cloud pairs are still inadequately reconstructed in terms of geometric details, as observed in the first example in Fig. 1. We postulate that this is mainly due to the lack of critical local features, such as planar regions, in the inadequately reconstructed areas. As the reconstruction auxiliary task is optimized jointly with the point cloud registration task, the backbone network tends to capture geometric structures with critical local features to improve the accuracy of point cloud registration.

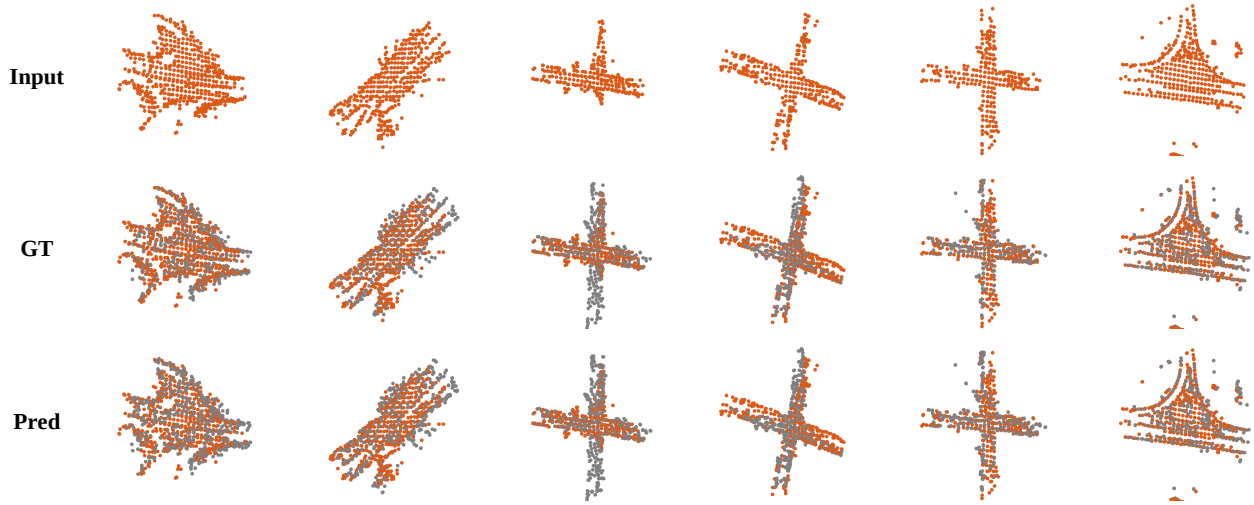

Figure 3: Reconstruction results obtained on the KITTI dataset.

D. Loss Functions

Our method is additionally trained with three other loss functions: an overlap loss $\mathcal{L}_o$, a correspondence loss $\mathcal{L}_c$, and a feature loss $\mathcal{L}_f$. By introducing coefficients λ_c and λ_f, the final loss function is constructed and formulated as

$$\mathcal{L} = \mathcal{L}_o + \lambda_c \mathcal{L}_c + \lambda_f \mathcal{L}_f. \tag{1}$$

Overlap loss. $\mathcal{L}_o$ measures the consistency between the ground-truth overlap labels $\boldsymbol{o}^{\tilde{X}}$, $\boldsymbol{o}^{\tilde{Y}}$ and the predicted overlap scores $\hat{\boldsymbol{o}}^{\tilde{X}}$, $\hat{\boldsymbol{o}}^{\tilde{Y}}$. $\mathcal{L}_o = \mathcal{L}_o^X + \mathcal{L}_o^Y$, and $\mathcal{L}_o^X$ is defined as

$$\mathcal{L}_o^X = \frac{-1}{M'} \sum_{i=1}^{M'} [\boldsymbol{o}^{\tilde{X}_i} \times \log \hat{\boldsymbol{o}}^{\tilde{X}_i} + (1 - \boldsymbol{o}^{\tilde{X}_i}) \times \log(1 - \hat{\boldsymbol{o}}^{\tilde{X}_i})]. \tag{2}$$

The overlap labels $\boldsymbol{o}^{\tilde{X}}$, $\boldsymbol{o}^{\tilde{Y}}$ are obtained by downsampling the overlap labels $\boldsymbol{o}^X$, $\boldsymbol{o}^Y$ of point clouds $\boldsymbol{X}$, $\boldsymbol{Y}$, where $\boldsymbol{o}^X$, $\boldsymbol{o}^Y$ are computed by setting a threshold r_o for the closest point distances between the aligned point clouds.

Correspondence loss. $\mathcal{L}_c$ measures the correctness of the predicted corresponding point clouds in the overlapping regions based on the ℓ^1 loss. $\mathcal{L}_c = \mathcal{L}_c^X + \mathcal{L}_c^Y$, with the correspondence loss $\mathcal{L}_c^X$ defined as

$$\mathcal{L}_c^X = \frac{1}{\sum_{i=1}^{M'} \boldsymbol{o}^{\tilde{X}_i}} \sum_{i=1}^{M'} \boldsymbol{o}^{\tilde{X}_i} \left| \boldsymbol{T}_X^Y \left(\tilde{X}_i \right) - \hat{Y}_i \right|, \tag{3}$$

where $\boldsymbol{T}_X^Y$ is the ground-truth transformation from $\boldsymbol{X}$ to $\boldsymbol{Y}$.

Feature loss. $\mathcal{L}_f$ measures the discriminative power of the extracted features based on the InfoNCE loss [11]. $\mathcal{L}_f = \mathcal{L}_f^X + \mathcal{L}_f^Y$, with $\mathcal{L}_f^X$ defined as

$$\mathcal{L}_f^X = -\mathbb{E}_{x \in \mathcal{X}} \left[\log \frac{f(x, p_x)}{f(x, p_x) + \sum_{n_x} f(x, n_x)} \right], \tag{4}$$
$$f(x, c) = \exp(\boldsymbol{\mathcal{F}}^{x\,T} W_f \boldsymbol{\mathcal{F}}^c),$$

where $\mathcal{X}$ denotes the set of points $\mathcal{X} \subseteq \tilde{\boldsymbol{X}}$ with a correspondence in $\tilde{\boldsymbol{Y}}$; $\boldsymbol{\mathcal{F}}^x$ indicates the extracted features for point x. p_x and n_x denote the positive and negative points in $\tilde{\boldsymbol{Y}}$, which are selected based on the positive and negative margins (r_p, r_n); and W_f is a learnable linear transformation.

E. Additional Implementation Details

Our MRT employs a 6-layer transformer encoder and a 1-layer MRA decoder, with 8 heads in the transformers. The dimensions of both the transformer encoder and MRA decoder are set to 256. Regarding the 3DMatch dataset, each point cloud is first downsampled into 32 center points and then divided into point patches, where each point patch contains 32 points. The values of (r_p, r_n) are set to $(0.2, 0.4)$. During training, MRT uses a batch size of 1 over 70 epochs, and the multi-step LR policy reduces the learning rate by 0.5 at epochs [20, 40, 60]. For the ModelNet40 dataset, each point cloud is downsampled into 32 center points, and each point patch contains 16 points. The values of (r_p, r_n) are set to $(0.12, 0.24)$. During training, MRT uses a batch size of 4 over 400 epochs, and the learning rate is halved every 100 epochs. Regarding the KITTI dataset, each point cloud is downsampled into 32 center points and divided into point patches, where each point patch contains 16 points. The values of (r_p, r_n) are set to $(4.8, 9.6)$. During training, MRT uses a batch size of 1 over 200 epochs, and the learning rate is halved every 50 epochs.

F. Notations

To further improve the reading experience, the notations utilized in the article are shown in Table 4.

Table 4: Notations utilized throughout the article.

Symbol	Description	Symbol	Description
R	The ground truth rotation matrix $R \in SO(3)$	t	The ground truth translation vector $t \in \mathbb{R}^3$
$\hat{R}$	The predicted rotation matrix $\hat{R} \in SO(3)$	$\hat{t}$	The predicted translation vector $\hat{t} \in \mathbb{R}^3$
X	Source point cloud $X = \{x_1, x_2, ..., x_M\} \subseteq \mathbb{R}^3$	Y	Target point cloud $Y = \{y_1, y_2, ..., y_N\} \subseteq \mathbb{R}^3$
$\tilde{X}$	The downsampled source point cloud $\tilde{X} \in \mathbb{R}^{M^{'} \times 3}$	$\tilde{Y}$	The downsampled target point cloud $\tilde{Y} \in \mathbb{R}^{N^{'} \times 3}$
$F^{\tilde{X}}$	The encoded features $F^{\tilde{X}} \in \mathbb{R}^{M^{'} \times D}$ of $\tilde{X}$ by KPConv	$F^{\tilde{Y}}$	The encoded features $F^{\tilde{Y}} \in \mathbb{R}^{N^{'} \times D}$ of $\tilde{Y}$ by KPConv
$\tilde{X}_c$	The center points of $\tilde{X}$	$\tilde{Y}_c$	The center points of $\tilde{Y}$
$\tilde{X}_p$	The point patches in the $\tilde{X}$	$\tilde{Y}_p$	The point patches in the $\tilde{Y}$
$T_m^{\tilde{X}}$	The mask tokens of $\tilde{X}$	$T_m^{\tilde{Y}}$	The mask tokens of $\tilde{Y}$
$T_f^{\tilde{X}}$	The full set of tokens of $\tilde{X}$	$T_f^{\tilde{Y}}$	The full set of tokens of $\tilde{Y}$
ψ_Y^X	The ground truth transformations from Y to X	ψ_X^Y	The ground truth transformations from X to Y
$P_m^{\tilde{X}}$	The positional encoding of mask tokens used for $\tilde{X}$	$P_m^{\tilde{Y}}$	The positional encoding of mask tokens used for $\tilde{Y}$
$T^{\tilde{X}}$	The full set of tokens of $\tilde{X}$ with positional encoding	$T^{\tilde{Y}}$	The full set of tokens of $\tilde{Y}$ with positional encoding
$\hat{Y}_p$	The point patches in the aligned $\tilde{Y}$ predicted by $\tilde{X}$	$\hat{X}_p$	The point patches in the aligned $\tilde{X}$ predicted by $\tilde{Y}$
$\hat{\mathcal{Y}}$	The corresponding point cloud predicted by $\tilde{X}$	$\hat{\mathcal{X}}$	The corresponding point cloud predicted by $\tilde{Y}$
$\hat{o}$	The predicted overlap scores	o	The ground truth overlap scores

References

[1] Xuyang Bai, Zixin Luo, Lei Zhou, Hongbo Fu, Long Quan, and Chiew-Lan Tai. D3feat: Joint learning of dense detection and description of 3d local features. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6359–6367, 2020. 1

[2] Anh-Quan Cao, Gilles Puy, Alexandre Boulch, and Renaud Marlet. Pcam: Product of cross-attention matrices for rigid registration of point clouds. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 13229–13238, 2021. 1

[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. 1

[4] Christopher Choy, Wei Dong, and Vladlen Koltun. Deep global registration. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2514–2523, 2020. 1

[5] Christopher Choy, Jaesik Park, and Vladlen Koltun. Fully convolutional geometric features. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8958–8966, 2019. 1

[6] Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5545–5554, 2019. 1

[7] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16000–16009, 2022. 1

[8] Shengyu Huang, Zan Gojcic, Mikhail Usvyatsov, Andreas Wieser, and Konrad Schindler. Predator: Registration of 3d point clouds with low overlap. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4267–4276, 2021. 1

[9] Junha Lee, Seungwook Kim, Minsu Cho, and Jaesik Park. Deep hough voting for robust global registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15994–16003, 2021. 1

[10] Yang Li and Tatsuya Harada. Lepard: Learning partial point cloud matching in rigid and deformable scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5554–5564, 2022. 1

[11] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018. 3

[12] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, and Kai Xu. Geometric transformer for fast and robust point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11143–11152, 2022. 1

[13] Siwen Quan and Jiaqi Yang. Compatibility-guided sampling consensus for 3-d point cloud registration. *IEEE Transactions on Geoscience and Remote Sensing*, 58(10):7380–7392, 2020. 1

[14] Hao Xu, Shuaicheng Liu, Guangfu Wang, Guanghui Liu, and Bing Zeng. Omnet: Learning overlapping mask for partial-to-partial point cloud registration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3132–3141, 2021. 1

[15] Zi Jian Yew and Gim Hee Lee. Regtr: End-to-end point cloud correspondences with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6677–6686, 2022. 1

[16] Hao Yu, Fu Li, Mahdi Saleh, Benjamin Busam, and Slobodan Ilic. Cofinet: Reliable coarse-to-fine correspondences

for robust pointcloud registration. *Advances in Neural Information Processing Systems*, 34:23872–23884, 2021. 1

[17] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1802–1811, 2017. 1