Efficient and Accurate Point Cloud Registration with Sparsepoint Transfomer for Landslide Detection

Guiyu Zhao[1,2], Xin Wang[1,2], Zhentao Guo[1,2], and Hongbin Ma[1,2]

[1] School of Automation, Beijing Institute of Technology, Beijing, China
[2] National Key Lab of Autonomous Intelligent Unmanned Systems, Beijing Institute of Technology, Beijing, China

Abstract. We present an efficient and accurate point cloud registration method that has been successfully applied to real-world landslide detection tasks. Existing point cloud registration methods often suffer from either low accuracy or high computational complexity, rendering them impractical for real-world applications. In this paper, we propose an efficient and accurate **SparsePoint Transformer** framework for point cloud registration, which directly applies an attention mechanism to sparse points, significantly enhancing computational efficiency. Additionally, we introduce a **cascaded feature aggregation encoder** to enrich the contextual details in point clouds, thereby improving registration accuracy. To further adapt our framework for landslide detection, we propose the point cloud registration for landslide detection **(PCR4LD)** framework, built on the SparsePoint Transformer pipeline. This framework first addresses vegetation interference with two proposed solutions. Subsequently, it employs **ICP-based pose refinement** for further pose refinement, ensuring accurate landslide detection. Finally, **region merging and filtering** are applied to identify the landslide-affected regions. Our method not only achieves state-of-the-art registration results on the public KITTI dataset with a significant speedup but also demonstrates outstanding performance in real-world landslide detection tasks, significantly outperforming other methods.

Keywords: Point cloud registration, landslide detection, Transfomer, feature extraction

1 Introduction

Point cloud registration [11] has long been a prominent task in computer vision, aiming to align point clouds captured from different viewpoints into a common reference coordinate system. This technique has a wide range of applications, including high-precision map generation in autonomous driving [30, 22], landslide detection in remote sensing [27], and pose estimation for robotic grasping [15, 8]. There are numerous approaches to point cloud registration, which can broadly be categorized into three types: optimization-based methods [4, 19, 18, 7], learning-based methods [26, 1, 10, 14], and handcrafted feature-based methods [16, 15, 17].

Before deep learning became widely used in vision tasks, most methods relied on the Iterative Closest Point (ICP) algorithm [4, 19, 18] and handcrafted descriptor matching techniques [15, 17]. These traditional methods can efficiently perform registration without significant computational resources. However, they suffer from poor generalization performance [1] and struggle in outdoor real-world scenarios. As a result, their performance in highly noisy and cluttered mountain scenes is notably poor, leading to suboptimal point cloud alignment and, consequently, inaccurate landslide detection.

With the advent and widespread adoption of deep learning [21, 12], learning-based methods [1, 10, 14] have gained significant attention in the community, leading to the emergence of learning-based methods. They can be divided into two categories: correspondence-based methods [1, 10, 14] and correspondence-free methods [23, 31, 3]. Correspondence-based methods first utilize deep networks to learn point cloud features, then establish correspondences through feature matching, and finally estimate the pose using techniques like RANSAC [7] or Singular Value Decomposition (SVD) [4]. In contrast, correspondence-free methods directly learn the SE(3) pose parameters through a network, bypassing the need to estimate correspondences to achieve registration. While both of these learning-based methods demonstrate significantly improved performance over traditional methods, they also come with drawbacks. Due to the deep networks and their direct handling of dense point clouds, these approaches are often inefficient and heavily reliant on computational resources. This creates challenges for practical deployment, making them less suitable for real-world applications.

To achieve robust and accurate registration, we have opted for a learning-based approach as our solution. To address the issues of high computation time and resource overhead in current learning-based methods, we propose the **SparsePoint Transformer** feature extraction framework. By processing sparse rather than dense point data directly, our approach significantly improves efficiency. Additionally, we observed that using a transformer directly on sparse points results in the loss of local detail information, ultimately leading to a decrease in accuracy. To mitigate this issue, we introduce a **cascaded feature aggregation encoder** that performs feature downsampling while aggregating and cascading the features, thereby reducing the loss of local information and enhancing feature robustness.

Finally, to better apply our registration algorithm to landslide detection, we integrated ICP-based fine registration and a region merging and elimination algorithm into the registration pipeline. **ICP-based pose refinement** allows for pose refinement and more precise alignment. Following this, we applied an algorithm based on Euclidean distance and region-growing segmentation to identify the **landslide dectection**.

In this paper, we have two primary objectives: first, to design an efficient and accurate point cloud registration algorithm that does not rely on extensive computational resources; second, to apply this algorithm to a remote sensing task—landslide detection through point cloud registration. The design of the SparsePoint Transformer enables both lightweight and efficient performance. Ad-

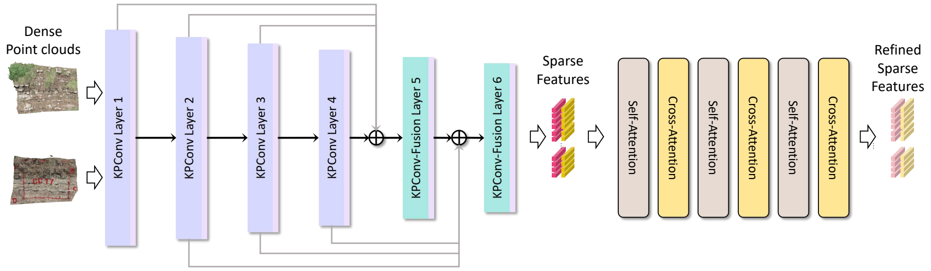


Fig. 1: Feature extraction pipeline of our method.

ditionally, the cascaded feature aggregation encoder and ICP-based fine registration further ensure accuracy and robustness. With these designs, our method can effectively perform landslide detection in real-world scenarios.

We validated our approach on both indoor and outdoor datasets, demonstrating its performance. Our method achieves efficient point cloud registration while maintaining accuracy, with a runtime that is only 0.04 times that of SpinNet and 0.2 times that of GeoTransformer [14]. Notably, on the KITTI dataset, our method achieves comparable registration recall (RR) and rotation error (RE) to GeoTransformer, realizing accurate registration. Finally, we conduct landslide detection on our own collected mountain data, where our method achieves a 85.2% mean Average Precision (mAP), demonstrating accurate and robust landslide detection. Our main contributions are as follows:

- We propose the SparsePoint Transformer, a lightweight registration framework that significantly improves computational efficiency and achieves efficient, accurate registration.
- We introduce a feature aggregation encoder that minimizes the loss of local information, ensuring registration accuracy.
- We present a new landslide detection solution based on point cloud registration, which has shown excellent performance in real-world scenarios.

2 Related Work

Point cloud registration [11] is a classic and significant task in computer vision. Historically, most point cloud registration methods [4, 15, 17, 7] are based on optimization techniques or handcrafted feature descriptors. The introduction of the Fast Point Feature Histogram (FPFH) descriptor [15] marks a milestone in point cloud registration, as it enables global registration through matching local descriptors. In recent years, with the rise of deep learning [12], learning-based methods [26, 10, 1, 14] emerge. 3DMatch [26] pioneers learning-based approaches by utilizing convolutional networks to capture geometric features of local patches. Predator [10] addresses the issue of low overlap in point clouds by predicting overlap scores using graph convolutional networks. With the advent of transformers [21] and their application in vision tasks [9], numerous frameworks [24, 14, 25] based on transformer feature extraction and matching are proposed, achieving state-of-the-art results in point cloud registration.

3 Point Cloud Registration

3.1 Problem definition

Point cloud registration (PCR) involves estimating a transformation to align two point clouds. This task can be framed as a least-squares optimization problem

$$\underset{\mathbf{R}\in SO(3),\mathbf{t}\in\mathbb{R}^3}{\arg\min} \sum_{\left(\mathbf{p}_{x_i},\mathbf{q}_{y_i}\right)\in\mathcal{C}} \left\|\mathbf{R}\cdot\mathbf{p}_{x_i}+\mathbf{t}-\mathbf{q}_{y_i}\right\|_2^2 \tag{1}$$

where $\mathcal{C}$ is the set of correspondences. In this paper, we first align the source and target point clouds, and then assess whether changes in the terrain have occurred by analyzing the distance discrepancies between corresponding points.

3.2 Cascaded feature aggregation encoder

We process sparse point features directly using a Transformer [14, 21] to improve efficiency, which inevitably leads to the loss of local detail information in the original point cloud $\mathbf{P}$ and $\mathbf{Q}$. To address this, we designed a cascaded feature aggregation encoder to capture and aggregate local details from dense points. This allows the resulting sparse features to incorporate rich contextual geometric information, minimizing feature loss and ensuring accurate registration.

The feature aggregation encoder is implemented using KPConv [20] as the basic module, using original point clouds $\mathbf{P}$ and $\mathbf{Q}$ as input, performing multi-level downsampling and feature aggregation. At each layer Θ^i, both the points $\mathbf{P}^i$ and their features $\mathbf{F}^i$ are downsampled where $i = 1, 2, ..., 6$. And KPConv [20] is used to aggregate features from the original points to the downsampled points. After 6 layers Θ of feature extraction, we obtain the final sparse features $\widetilde{\mathbf{F}}^{\mathcal{P}}$ and $\widetilde{\mathbf{F}}^{\mathcal{Q}}$.

Notably, in the last two downsampling steps, to prevent excessive sparsity and the subsequent loss of detail, we introduced multi-stage cascaded feature embedding. The specific approach is shown in Figure 1 on the left. The outputs of the first three layers are concatenated with the inputs of the second-to-last layer, while the outputs of the middle three layers are concatenated with the inputs of the last layer.

$$\mathbf{F}^{i+1} = \Theta^i(\text{concat}[\mathbf{F}^{i-1}, \mathbf{F}^{i-2}, \mathbf{F}^{i-3}, \mathbf{F}^{i-4}]) \tag{2}$$

where $i = 5, 6$, $\widetilde{\mathbf{F}} = \mathbf{F}^6$. This approach establishes connections between dense surface-level features and sparse deep-level features, fully leveraging contextual information. As a result, the final sparse features retain the ability to represent local details effectively.

3.3 SparsePoint Transformer

Unlike PointTransformer [29] and GeoTransformer [14], which directly use points or superpoints as inputs for the Transformer, we utilize sparsely sampled points as input. This approach significantly reduces computational cost and inference time, and due to the smaller data volume, it also greatly decreases memory usage, making it less dependent on extensive computational resources.

In the cascaded feature aggregation encoder, we ensure comprehensive contextual information representation while performing downsampled feature aggregation. The output $\widetilde{\mathbf{F}}$ from the encoder is then fed into the SparsePoint Transformer for further refinement of the features. This stage captures long-range dependencies between sparse points $\widetilde{\mathbf{P}}$ and enhances the global representation of local features. Through iterative alternation of self-attention and cross-attention mechanisms, we obtain refined sparse features $\widehat{\mathbf{F}}$. These features possess global awareness and enable cross-point cloud interaction, which benefits subsequent feature matching tasks.

Self-attention mechanism. First, the sparse point features $\widetilde{\mathbf{F}} \in \mathbb{R}^{|\widehat{\mathcal{P}}|\times\widetilde{d}}$ of the point cloud $\widetilde{\mathbf{P}} \in \mathbb{R}^{|\widehat{\mathcal{P}}|\times 3}$ are used as the input to the model, and the weighted projection feature $\mathbf{Z} \in \mathbb{R}^{|\widehat{\mathcal{P}}|\times\widetilde{d}}$ is computed using the value matrix $\mathbf{W}^V$

$$\mathbf{z}_i = \sum_{j=1}^{|\widehat{\mathcal{P}}|} a_{i,j} \left(\mathbf{x}_j \mathbf{W}^V\right) \tag{3}$$

Here, $a_{i,j}$ represents the attention weight coefficient, which is obtained by applying a row-wise softmax to the attention scores E. The attention score $e_{i,j}$ is calculated by the product of the query matrix $\mathbf{W}^Q$ and the key matrix $\mathbf{W}^K$:

$$e_{i,j} = \frac{\left(\mathbf{x}_i \mathbf{W}^Q\right)\left(\mathbf{x}_j \mathbf{W}^K\right)^T}{\sqrt{\widetilde{d}}} \tag{4}$$

Subsequently, the self-attention module computation is completed by applying a linear layer, a normalization layer, and a feed-forward layer. Similarly, the same steps are applied to the point cloud $\mathcal{P}$ to complete the self-attention module computation.

Cross-attention mechanism. To enhance the cross-point-cloud interaction capability of the features, we also compute the cross-attention between point cloud $\mathcal{P}$ and point cloud $\mathcal{Q}$. First, similar to the self-attention mechanism, the weighted projection feature is computed for point cloud $\mathcal{P}$ using the matrix $\mathbf{W}^V$:

$$\mathbf{z}_i^{\mathcal{P}} = \sum_{j=1}^{|\widehat{\mathcal{Q}}|} a_{i,j} \left(\mathbf{x}_j^{\mathcal{Q}} \mathbf{W}^V\right) \tag{5}$$

Here, $a_{i,j}$ represents the attention weight coefficient, which is obtained by applying a row-wise softmax to the attention scores E. The difference in cross-attention is that the attention score $e_{i,j}$ e is computed by the product of the

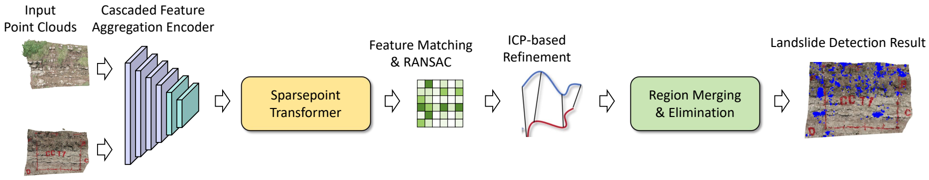


Fig. 2: The whole framework of point cloud registration for landslide detection.

query $\mathbf{W}^Q$ from point cloud $\mathcal{P}$ and the key $\mathbf{W}^K$ from point cloud $\mathcal{Q}$:

$$e_{i,j} = \frac{\left(\mathbf{x}_i^{\mathcal{P}} \mathbf{W}^Q\right)\left(\mathbf{x}_j^{\mathcal{Q}} \mathbf{W}^K\right)^T}{\sqrt{\widetilde{d}}} \tag{6}$$

Similarly, we apply the same steps to point cloud $\mathcal{Q}$ to complete the cross-attention module computation. By iteratively alternating between the self-attention and cross-attention mechanisms, we obtain refined sparse features $\widehat{\mathbf{F}}$.

4 PCR for Landslide detection

4.1 ICP-based pose refinement

Our sparse point feature matching method is efficient and robust, but it does have some limitations in accuracy compared to high-precision point cloud registration methods. Given that landslide detection based on point cloud registration requires precise alignment, we introduce an ICP-based pose refinement to enhance the accuracy of the pose estimation.

The approach is straightforward: we use ICP to achieve fine registration. Specifically, considering that outdoor environments often contain large flat regions, we employ a point-to-plane ICP that incorporates discovered constraints to accelerate convergence.

$$\mathbf{T} \leftarrow \arg\min_{\mathbf{T}} \sum_i w_i \left\| \eta_i \left(\mathbf{T} \cdot \mathbf{p}_i - \mathbf{q}_i \right) \right\|_2^2 \tag{7}$$

where w_i is the weight and η_i is the normal at $\mathbf{p}_i$. Through iterative optimization, we ultimately obtain a transformation matrix $\mathbf{T} = \{\mathbf{R}, \mathbf{t}\}$ with reduced error.

4.2 Region detection

By performing point cloud registration, we achieve high-precision alignment of point clouds captured from different viewpoints at different times. Given the static nature of the actual mountain structure, we assume that areas with significant changes in the mountain point cloud indicate the presence of landslides. However, considering that seasonal changes in vegetation, such as trees, could interfere with landslide detection, we propose two solutions:

- Semantic segmentation: Using a mature semantic segmentation network to identify and remove trees from the point cloud.
- Color mask: A more efficient method that removes trees by applying a simple color threshold.

In practice, the choice between these two methods depends on the trade-off between accuracy and efficiency. After applying the selected method, we obtain the filtered point clouds $\mathbf{P}^f$ and $\mathbf{Q}^f$.

Landslide point classification. Subsequently, we define a landslide determination factor and use it to distinguish between landslide $\mathbf{P}_l$ and non-landslide areas $\mathbf{P}_n$ by evaluating the Euclidean distance between the aligned point clouds.

$$\mathbf{P}_l = \left\{ \mathbf{p}_i^f \middle| \mathbf{p}_i^f \in \mathbf{P}^f \bigwedge \tau_b < \left\| \mathbf{R} \cdot \mathbf{p}_i^f + t - \mathbf{q}_j^f \right\|_2^2 < \tau_t \right\} \tag{8}$$

where τ_t and τ_b are the upper and lower bounds of the distance.

Region merging and filtering. Finally, given a predefined area threshold and growth parameters, we perform region merging and filtering based on region growing segmentation. We consider only regions with an area larger than the specified threshold as landslide regions, ignoring smaller changes. Through this process of region merging and filtering, we ultimately obtain the set of landslide regions $\mathbf{S}$. In practical applications, the sensitivity of landslide detection can be adjusted according to specific requirements by modifying the parameters for region merging and filtering, as well as the area threshold.

5 Experiment

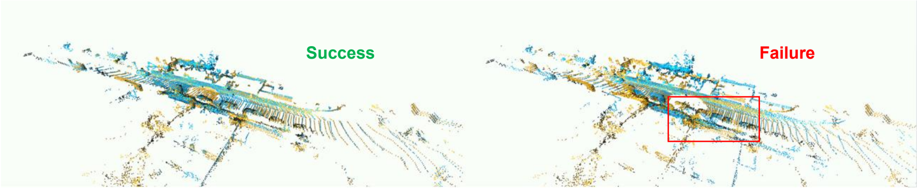
5.1 Point cloud registration

Dataset. We chose to evaluate the performance of our method on the outdoor KITTI dataset. KITTI odometry is an outdoor autonomous driving dataset, which contains 11 sequences. Following [14, 28], we use sequences 0∼5 for training, 6∼7 for validation and 8∼10 for testing. In addition, we refined the ground-truth pose by ICP following the official setting [1, 14].

Table 1: Registration results on KITTI odometry.

Model	RTE(cm)↓	RRE(°)↓	RR(%)↑	Time(s)↓
FCGF [5]	9.5	0.30	96.6	5.31
D3Feat [2]	7.2	0.30	**99.8**	5.89
SpinNet [1]	9.9	0.47	99.1	103.91
Predator [10]	**6.8**	0.27	**99.8**	2.86
CoFiNet [24]	8.2	0.41	**99.8**	2.64
GeoTransformer [14]	7.4	0.27	**99.8**	1.82
SPTransformer (ours)	8.6	0.27	**99.8**	**0.42**

Metrics. Following [1, 14], We use three metrics to evaluate the performance of our method. (1) *Relative Rotation Error* (RRE), the geodesic distance between



(a) Ours

(b) D3Feat

Fig. 3: Qualitative results on the KITTI dataset.

estimated and ground-truth rotation matrices, (2) *Relative Translation Error* (RTE), the Euclidean distance between estimated and ground-truth translation vectors, and (3) *Registration Recall* (RR), the fraction of point cloud pairs satisfies RRE<5° and TE<2m:

$$\mathrm{RR} = \frac{1}{H}\sum_{h=1}^{H} \mathbb{1}\left(\mathrm{RRE}_h < 2\ \mathrm{m} \wedge \mathrm{RTE}_h < 5^\circ\right). \tag{9}$$

where H is the number of point cloud pairs in KITTI.

Results. We compare our method with the current state-of-the-art algorithms: [5, 2, 1, 10, 14]. The results of the comparative experiments are shown in Table 1. As can be seen from the table, our method achieve performance on par with the advanced GeoTransformer [14]. We attain a registration success rate of 99.8%, the highest among all methods, and the second-lowest registration errors in terms of RRE and RTE, with accuracy only slightly lower than the computationally intensive GeoTransformer [14]. Notably, our method is RANSAC-free and, due to the design of the SparsePoint Transformer, is highly efficient with minimal computational overhead. Processing a pair of point clouds takes only 0.1 seconds, which is just 20% of the time required by [14]. The qualitative experimental results of our method on the KITTI dataset are shown in Figure 3. Our method demonstrates robust registration even in challenging scenarios.

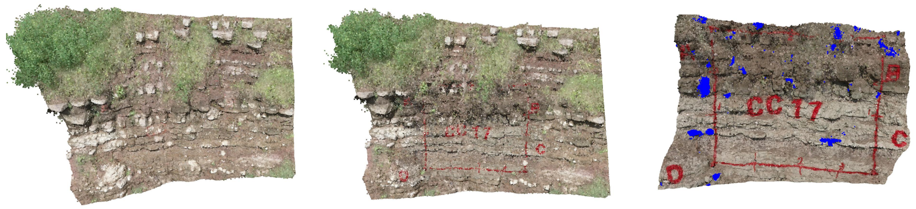
5.2 Landslide detection

Dataset. For the task of Landslide Detection, we utilized a self-constructed dataset. We employed a DJI L1 LiDAR sensor to collect point cloud data of a hillside at two different time points. To ensure an adequate number of true landslide samples, we set the time interval between the two data collection points to 6 months, allowing for observable changes in the terrain over this period. Data collection was conducted over two large-scale scenarios, resulting in a total of 80 pairs of valid point cloud data. We name our dataset as **Landslide-80**.

Metrics. We approach this task as a binary semantic segmentation task for distinguishing between landslide and non-landslide regions. Therefore, for evaluating Landslide detection, we selected standard semantic segmentation metrics,

Table 2: Experimental results of landslide detection.

Model	mIoU ↑	OA ↑	mAP ↑
Segmentation-based method			
RangeNet [13]	41.9	59.2	45.0
SalsaNext [6]	43.2	66.2	46.0
Registration-based method			
SpinNet [1]	76.2	82.9	77.9
GeoTransformer [14]	57.9	66.8	62.3
SPTransformer (ours)	**83.1**	**93.2**	**85.2**



(a) Original scene (b) Registration result (c) Landslide detection result

Fig. 4: Qualitative results of landslide detection. The landslide areas are visualized in blue.

including mean *Intersection over* Union (mIoU), *Overall Accuracy* (OA), and *mean Average Precision* (mAP).

Results. For the baseline comparison in our experiments, we selected two categories of algorithms. The first category includes semantic segmentation methods:RangeNet [13],SalsaNext [6], while the second category consists of registration error-based methods: SpinNet [1], GeoTransformer [14]. The comparison results are presented in Table 2. Our method belongs to registration error-based methods. The visualization results of the landslide detection are shown in Figure 4.

As observed from the table, the registration error-based methods significantly outperform the semantic segmentation methods. This superior performance can be attributed to the lack of sufficient data for adequately training the semantic segmentation models, which results in poor generalization performance. In contrast, registration error-based methods can be pre-trained on large public datasets and then fine-tuned on the actual dataset, leading to better generalization. Additionally, the registration error-based approach offers a more straightforward and effective solution for this specific task. Moreover, our proposed method achieved the best performance across multiple metrics: mIOU, OA, mAP, demonstrating both efficiency and accuracy in landslide detection.

6 Conclusion

In this work, we introduced the SparsePoint Transformer, a novel point cloud registration framework that addresses the limitations of existing methods by achieving a superior balance between accuracy and computational efficiency. By directly applying an attention mechanism to sparse points, our approach significantly enhances processing speed without compromising on accuracy. The inclusion of a cascaded feature aggregation encoder further improves the contextual understanding of point clouds, resulting in more precise registration. To tailor this framework for landslide detection, we developed the PCR4LD pipeline. This extension integrates effective solutions for mitigating vegetation interference, ensuring that environmental factors do not compromise detection accuracy. The pipeline also incorporates ICP-based pose refinement, which fine-tunes the alignment of point clouds, leading to more reliable detection outcomes. Finally, region merging and filtering techniques are applied to isolate and identify landslide-affected areas accurately. Our method was extensively evaluated on both the public KITTI dataset and real-world landslide detection tasks. The results demonstrate that our approach not only achieves state-of-the-art performance in terms of registration accuracy but also significantly reduces computational overhead, with processing times an order of magnitude faster than competing methods. Additionally, in practical applications, our method outperforms existing techniques by a substantial margin, particularly in challenging scenarios. These findings validate the potential of the SparsePoint Transformer and PCR4LD frameworks as robust solutions for efficient and accurate point cloud registration and landslide detection in real-world settings.

References

1. Ao, S., Hu, Q., Yang, B., Markham, A., Guo, Y.: Spinnet: Learning a general surface descriptor for 3d point cloud registration. In: CVPR. pp. 11753–11762 (2021)
2. Bai, X., Luo, Z., Zhou, L., Fu, H., Quan, L., Tai, C.L.: D3feat: Joint learning of dense detection and description of 3d local features. In: CVPR. pp. 6359–6367 (2020)
3. Bernreiter, L., Ott, L., Nieto, J., Siegwart, R., Cadena, C.: Phaser: A robust and correspondence-free global pointcloud registration. IEEE Robotics and Automation Letters **6**(2), 855–862 (2021)
4. Besl, P., McKay, N.D.: A method for registration of 3-d shapes. IEEE TPAMI **14**(2), 239–256 (1992)
5. Choy, C., Park, J., Koltun, V.: Fully convolutional geometric features. In: ICCV. pp. 8957–8965 (2019)
6. Cortinhal, T., Tzelepis, G., Erdal Aksoy, E.: Salsanext: Fast, uncertainty-aware semantic segmentation of lidar point clouds. In: Advances in Visual Computing: 15th International Symposium, ISVC 2020, San Diego, CA, USA, October 5–7, 2020, Proceedings, Part II 15. pp. 207–222. Springer (2020)
7. Fischler, M.A., Bolles, R.C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM **24**(6), 381–395 (1981)

8. Guo, N., Zhang, B., Zhou, J., Zhan, K., Lai, S.: Pose estimation and adaptable grasp configuration with point cloud registration and geometry understanding for fruit grasp planning. Computers and Electronics in Agriculture **179**, 105818 (2020)
9. Han, K., Wang, Y., Chen, H., Chen, X., Guo, J., Liu, Z., Tang, Y., Xiao, A., Xu, C., Xu, Y., et al.: A survey on vision transformer. IEEE transactions on pattern analysis and machine intelligence **45**(1), 87–110 (2022)
10. Huang, S., Gojcic, Z., Usvyatsov, M., Wieser, A., Schindler, K.: Predator: Registration of 3d point clouds with low overlap. In: CVPR. pp. 4267–4276 (2021)
11. Huang, X., Mei, G., Zhang, J., Abbas, R.: A comprehensive survey on point cloud registration. arXiv preprint arXiv:2103.02690 (2021)
12. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. nature **521**(7553), 436–444 (2015)
13. Milioto, A., Vizzo, I., Behley, J., Stachniss, C.: Rangenet++: Fast and accurate lidar semantic segmentation. In: 2019 IEEE/RSJ international conference on intelligent robots and systems (IROS). pp. 4213–4220. IEEE (2019)
14. Qin, Z., Yu, H., Wang, C., Guo, Y., Peng, Y., Xu, K.: Geometric transformer for fast and robust point cloud registration. In: CVPR. pp. 11143–11152 (2022)
15. Rusu, R.B., Blodow, N., Beetz, M.: Fast point feature histograms (fpfh) for 3d registration. In: ICRA. pp. 3212–3217 (2009)
16. Rusu, R.B., Blodow, N., Marton, Z.C., Beetz, M.: Aligning point cloud views using persistent feature histograms. In: IROS. pp. 3384–3391 (2008)
17. Salti, S., Tombari, F., Di Stefano, L.: Shot: Unique signatures of histograms for surface and texture description. CVIU **125**, 251–264 (2014)
18. Segal, A., Haehnel, D., Thrun, S.: Generalized-icp. In: RSS. p. 435. Seattle, WA (2009)
19. Sharp, G.C., Lee, S.W., Wehe, D.K.: Icp registration using invariant features. IEEE TPAMI **24**(1), 90–102 (2002)
20. Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L.J.: Kpconv: Flexible and deformable convolution for point clouds. In: ICCV. pp. 6411–6420 (2019)
21. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. NeurIPS **30** (2017)
22. Xiong, K., Zheng, M., Xu, Q., Wen, C., Shen, S., Wang, C.: Speal: Skeletal prior embedded attention learning for cross-source point cloud registration. arXiv preprint arXiv:2312.08664 (2023)
23. Yew, Z.J., Lee, G.H.: Rpm-net: Robust point matching using learned features. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 11824–11833 (2020)
24. Yu, H., Li, F., Saleh, M., Busam, B., Ilic, S.: Cofinet: Reliable coarse-to-fine correspondences for robust pointcloud registration. NeurIPS **34**, 23872–23884 (2021)
25. Yu, H., Qin, Z., Hou, J., Saleh, M., Li, D., Busam, B., Ilic, S.: Rotation-invariant transformer for point cloud matching. In: CVPR. pp. 5384–5393 (2023)
26. Zeng, A., Song, S., Nießner, M., Fisher, M., Xiao, J., Funkhouser, T.: 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In: CVPR. pp. 1802–1811 (2017)
27. Zhang, J., Lin, X.: Advances in fusion of optical imagery and lidar point cloud applied to photogrammetry and remote sensing. International Journal of Image and Data Fusion **8**(1), 1–31 (2017)
28. Zhao, G., Guo, Z., Wang, X., Ma, H.: Spherenet: Learning a noise-robust and general descriptor for point cloud registration. IEEE TGRS **62**, 1–16 (2024)

29. Zhao, H., Jiang, L., Jia, J., Torr, P.H., Koltun, V.: Point transformer. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 16259–16268 (2021)
30. Zheng, Y., Li, Y., Yang, S., Lu, H.: Global-pbnet: A novel point cloud registration for autonomous driving. IEEE Transactions on Intelligent Transportation Systems **23**(11), 22312–22319 (2022)
31. Zhu, M., Ghaffari, M., Peng, H.: Correspondence-free point cloud registration with so (3)-equivariant implicit shape representations. In: Conference on robot learning. pp. 1412–1422 (2022)