3D Pointcloud Registration In-the-wild

Peter Ørnulf Ivarsen [1], Marianne Bakken [1] and Ahmed Mohammed[1*]

Abstract— **This study assesses two state-of-the-art (SOTA) pointcloud registration approaches on industrially challenging datasets, focusing on two specific cases. The first case involves the application of Lidar-based Simultaneous Localization and Mapping (SLAM) in a tunnel environment, while the second case revolves around aligning RGBD scans from intricately symmetrical cast-iron machine parts within the domain of small-scale industrial production. Our evaluation involves testing state-of-the-art pointcloud registration approaches both with and without fine-tuning, and comparing the results to a classical hand crafted feature extractors. Our experimental findings reveal that existing SOTA models exhibit limited generalization capability when confronted with the more challenging pointcloud data. Moreover, robust generalizable methods beyond training are currently unavailable, highlighting a notable gap in addressing challenges associated with industrial datasets in pointcloud registration.**

I. Introduction

Pointcloud registration is critical for different applications such as SLAM (Simultaneous localization and mapping), 3D reconstruction, robotic interaction, and more. For an optimal point-cloud registration framework, key attributes such as superior robust generalizability to unseen data, accuracy, and acceptable efficiency are crucial. Nevertheless, striking the right balance proves highly challenging as existing registration techniques often fall short in terms of generalization, accuracy or efficiency. The fundamental question remains: How can we develop a method that strikes a balance among these essential elements to meet the requirements of various applications? Compounding the challenge, state-of-the-art registration approaches are typically tailored for a predefined set of scenes and 3D sensors, limiting their adaptability and usability in diverse settings. This limitation necessitates additional training or fine-tuning to address alternative scenes or sensors.

Several papers have reviewed pointcloud registration, concentrating on aspects like generalization, accuracy, or efficiency [1], [2]. However, most of these evaluations either focus on specific technical components rather than the entire registration pipeline [1], or they use datasets of limited scale, such as synthetically-generated data or LiDAR scans of indoor objects [2]. These evaluations typically include performance analysis of keypoint detection and description algorithms [2], as well as cross-source pointcloud registration between different sensor types, like Kinect and Lidar [1]. While these evaluations identify the best-performing algorithms and registration strategies in terms of accuracy and speed, their conclusions are often based on findings reported in the respective papers (datasets). While these assessments provide valuable insights into specific technical aspects, it's important to acknowledge the disparity in achieving successful registration between real-world scenarios and academic benchmarks, particularly when dealing with complex and diverse datasets.

In this paper, we aim to provide an alternative brief comparison of existing pointcloud registration methods, encompassing both feature-based registration and deep learning-based approaches, along with their performance on two novel real-world datasets. We assess three methods for pointcloud registration: traditional hand-crafted feature approaches (FPFH) with RANSAC (Random sample consensus) [3], deep learning-based learned feature with RANSAC [4], and end-to-end registration approach [5]. These datasets consist of two highly challenging test cases: tunnel lidar scans with high self-similarity and RGBD scans of symmetrical cast-iron machine parts with low degree of overlapping 3D features due to self-occlusion. We use these test cases to investigate the generalizability of models across datasets, and to what extent the proposed approaches are able to produce useful features for these challenging use-cases. Our results and discussion highlight some of the remaining challenges in pointcloud registration for scans from real-world scenarios with limited training data.

II. Background

In this section, we start by examining the constituent elements of traditional pointcloud registration pipelines (Correspondence-based approaches) before delving into more recent End-to-end pointcloud registration algorithms.

A. Correspondence based approaches

The general pipeline for correspondence based pointcloud registration follows a typical two-step process [6], [7], [8]. The first step is to extract correspondences between two pointclouds. Subsequently, it recovers the transformation between the clouds by aligning these correspondences using robust pose estimators, such as RANSAC. These methods can be further categorized into two classes according to how they extract correspondences. The first class aim to detect locally unique keypoints and learn more powerful descriptors for the keypoints. While the second class retrieves correspondences without keypoint detection by considering all possible matches. In earlier works on keypoint based descriptors, the focus was on characterizing local geometry through the use of handcrafted features [3]. Although these features often lacked robustness against clutter and occlusions, they generalize across diverse datasets. In recent years, there has

* Smart Sensors and Microsystems, SINTEF Digital, Oslo, Norway
*corresponding author, ahmed.mohammed@sintef.no

been a shift towards learned 3D feature descriptors, which have consistently outperformed the traditional handcrafted ones in terms of performance, on the other hand, these methods are oftentimes more specific to the data they were trained on.

1) Direct point-to-point pipeline: An example of this approach is the FPFH method employed in this work. Here locally unique points, referred to as keypoints, are detected based on local geometry. Descriptors for each of these keypoints are calculated. By matching these with descriptors from a target pointcloud, point to point correspondences are established. Another widely used technique is the Iterative Closest Point (ICP) algorithm [9]. This method uses so called "soft" correspondences, which are established on the basis of closeness to a point in the target cloud. This means they are sensitive to initialization and will typically converge to the nearest local minima, which in many cases will not coincide with the "true" alignment.

2) Coarse-to-fine pipeline: Such approaches start by establishing initial correspondences at the level of patches and subsequently refine them to achieve a more precise matching of individual points. These refined correspondences are further extended to create dense point-to-point correspondences within the specified patch region. For this study, we focus on the course-to-fine pipeline specifically CofiNet [4], which shows better performance over direct point-to-point methods. CofiNet [4] addresses the challenge of extracting correspondences for 3D pointcloud registration. The proposed approach extracts hierarchical correspondences in a coarse-to-fine manner without relying on keypoint detection. The model initiates by learning to match down-sampled nodes, generating initial node correspondences. Subsequently, these node proposals are progressively expanded to form patches, each comprising groups of points along with associated descriptors. The correspondences at the patch level are further refined down to the point level through a density-adaptive matching module. The effectiveness of the proposed method is evaluated on standard benchmarks for both indoor and outdoor scenarios.

B. End-to-end approaches

The methods mentioned above are all establishing some local correspondence between two pointclouds and then perform alignment based on these correspondences in a separate step. The end-to-end registration methods on the other hand, estimate the transformation directly during the optimization process [10]. These methods can be further classified into two classes. The first class follows the idea of ICP, which iteratively establishes soft correspondences and computes the transformation with differentiable weighted SVD (singular value decomposition). The second class first extracts a global feature vector for each pointcloud and regresses the transformation with the global feature vectors. Although direct registration methods have achieved promising results on synthetic shapes, they are less robust for large-scale scenes. GeoTransformer [5] pointcloud registration method is another end-to-end approach that is both keypoint-free and RANSAC-free. Given a superpoint, [5] learns a non-local representation based on pair-wise distances and triplet-wise angles. The backbone downsamples the input pointclouds and learns features in multiple resolution levels. The features are iteratively encoded intra-point-cloud geometric structures and inter-point-cloud geometric consistency. The superpoint correspondences are then propagated to dense points. Finally, the transformation is computed with a local-to-global registration method.

Fig. 1: The Piloting platform in Coripe, Spain

C. Datasets

There are several publicly available datasets for testing and improving pointcloud registration algorithms, that has facilitated the recent success of learning-based methods. The 3DMatch dataset [11] provides real-world 3D pointcloud data specifically designed for registration tasks with primarily focuses on indoor scenes, such as living rooms and offices. It also provides a more challenging benchmark, 3DLoMatch, where the pointclouds are cropped such that there is less overlap. The KITTI dataset [12] offers data from LiDAR and cameras used in autonomous driving in urban and highway driving scenarios. The ModelNet40 [13] dataset provides a collection of simple synthetic 3D CAD models from 40 object categories for tasks like object recognition and pointcloud registration.

The next section introduces two novel test scenarios designed to assess the generalizability and real-world accuracy SOTA pointcloud registration methods. These scenarios deviate from existing publicly available datasets, aiming to provide a more challenging and realistic (out-of-domain) evaluation.

III. In-the-wild test cases

Extracting correspondences from pointcloud data for the purpose of registration is an active field of research and new methods are being presented continuously. Commonly some way of sparcifying the data is performed, (keypoints, uniform down sampling, coarse to fine), before feature estimation and a correspondence search is performed. In cases where pointclouds contain repeated geometric structures, symmetries, or no locally unique geometries at all, correspondence search will be prone to errors as viewpoint can be the dominating contribution towards the uniqueness of a descriptor.

In the following, two such challenging cases will be investigated. One case is tied to performing Lidar based Simultaneous Localization and Mapping (SLAM) in a tunnel environment, and the other is aligning RGBD scans from highly symmetrical cast-iron machine parts in the context of automating small-scale production.

A. Tunnel Case

A challenge for any pointcloud registration algorithm is the case when the geometry of a pair of scans are not sufficiently constraining a rigid transform (6DoF) along all degrees of freedom. This is the case when aligning simple shapes such as planes, tubes and spheres and also more complex shape containing repeated structures or symmetries. Such a case was encountered in connection with the H2020 project PILOTING [14].

1) Tunnel dataset: The tunnel dataset was recorded in connection with [15], where an autonomous robot (Fig 1) which performs visual inspection of tunnels was developed. To allow the robot to navigate autonomously as well as report the position of damages that were detected, an accurate localization solution was required. A hardware / software solution which estimates ego motion based partly on scan registration was developed. To capture sufficient information to solve for localization, data from a Ouster OS0-128 LiDAR and a forward-facing FLIR BFS-U3-17S7M-C camera was combined. Sensors were time synchronized and intrinsics / extrinsics were estimated through calibration. The project made available a dataset consisting of three runs through a 175 meter straight tunnel-stretch outside the city of Coripe, Spain. In addition to lidar scans and images, the dataset contain ground truth positional data recorded with a Leica robotic totalstation.

The dataset present a particularly challenging case as geometry is almost identical for each scan along the tunnel except for small geometric features in the shape of lighting armatures in the roof and small 20x20 concrete blocks supporting a drainage pipe along the tunnel wall. See fig. 4 for example pointclouds. To create scan pairs, we have set the dataloader to pair scans which are between 1 meter and 4 meter apart. This interval secures that we have a high density overlap between the scans, while ensuring that they are reasonably spaced.

B. Cast Iron parts Case

Registration of cast manufactured parts is a prerequisite for automation of tasks such as sanding, welding and assembly in low-volume production. Typically, the part is placed with an arbitrary pose on a table surface or bin, and scanned

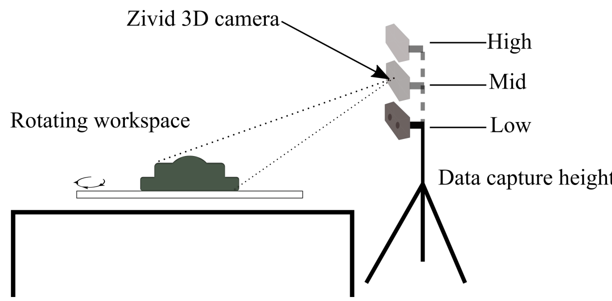


Fig. 2: Sensor setup for 3D scanning of cast parts.

once from one angle only. The pointcloud from the scan should be matched against a target scan or a CAD model to facilitate downstream tasks like computation of robotic tool path. While scan-to-CAD matching problems can utilize correspondences from the whole source pointcloud to compute the registration, scan-to-scan problems can suffer from low overlap between source and target. Other typical challenges are rotation symmetry and lack of distinctive local geometric features.

1) Cast manufactured parts dataset: We use 3D data of cast manufactured parts acquired for the scan-to-cad dataset in [16] to construct a scan-to-scan dataset. The cast parts are brass parts from from Mjøs Metallvarefabrikk, approximately 30-40 cm in diameter, and 20 individual physical items. The capture setup is shown in fig. 2. The parts were put on a turntable and scanned with a Zivid 3D camera (with HD resolution RGBD output) from 8 different angles, with 3 different heights of the camera, giving 24 scans per unique part, i.e. 480 scans in total. The ground truth registration was found through RGB pose estimation with aruco markers, but only the depth information is used in the datasets for this paper.

For each unique part, we pick two sets of scan-pair combinations for scan-to-scan registration: 1) scans with *high overlap* (less than 50° difference in rotation), and 2) scans with *low overlap* (more than 50° difference in rotation). Example pointclouds for the two different datasets are shown in fig. 3. Due to occlusions, the scans with low overlap have very few common 3D features, which creates a very challenging dataset.

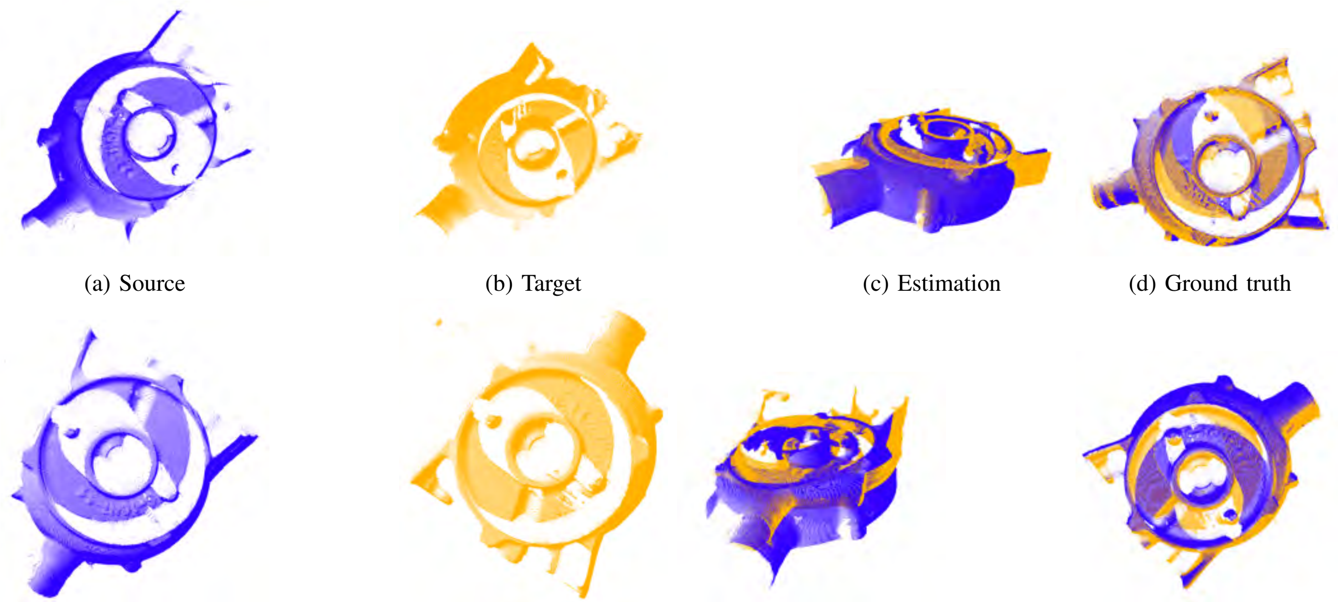
IV. Evaluation Methodology

A. Overview of methods

We evaluate three existing methods on our two test cases, representing different approaches; a classical hand-crafted approach (FPFH [3]) combined with RANSAC for pose estimation, a learning-based approach (CoFiNet [4]) combined with RANSAC for pose estimation, and an end-to-end learning-based approach without RANSAC ([5]).

To assess generalizability, we evaluate the performance of models pretrained on public datasets [1] that are aquired

[1] Pretrained models for COFiNet: `https://github.com/haoyu94/Coarse-to-fine-correspondences` and GeoTransformer `https://github.com/qinzheng93/GeoTransformer/releases`



(a) Source (b) Target (c) Estimation (d) Ground truth

(e) Source (f) Target (g) Estimation (h) Ground truth

Fig. 3: Example pointcloud pairs, registration results and ground truth registration for cast parts data. a)-d): Example pointcloud pair with high overlap (small angle between aquisitions), e)-h): example with low overlap (aquisition from opposite sides of the part). In c) the estimated transform has only a small error, but in g), the estimated transform places the part upside down.

with similar sensor types as our test data (but from different domains). For the tunnel case, we use models trained on the LIDAR part of the well known KITTI dataset [17], containing scenes from driving in urban environments. For the cast parts case, we use models trained on 3DMatch [18], which consists of a large collection of RGBD scans from indoor scenes.

To assess whether the proposed methods are able to produce relevant features for these challenging use-cases, we also evaluate performance of models trained on our datasets.

B. Learning-based methods

1) Tunnel case: As we test pointcloud registration techniques based on pure geometry, only the lidar data was used for benchmarking. The data was split in three equal parts for training/testing/validation. We have evaluated the performance of CofiNet [4] and GeoTransformer models [5], both using weights from training on the kitty dataset, as well as weights from training on the lidar data from the above mentioned tunnel data. The tolerances for qualifying a registration as a success was set to 30 cm and 3 degrees.

2) Cast parts case: We used the same overall network architectures as for the tunnel test case (CoFiNet combined with RANSAC and GeoTransformer end-to-end and with RANSAC). It should be noted that the GeoTransformer with and without RANSAC are both trained in the same end-to-end manner, but are evaluated using different pose estimation methods.

The pretrained methods were trained on the 3DMatch dataset [18]. To compensate for different scales in our scenes and the 3DMatch scenes, the pointcloud coordinates of our dataset were scaled by 0.1. Otherwise, the same parameters and metrics were used as for 3DMatch in [4]; RANSAC with an inlier threshold of 3 cm and 5000 samples, and a registration recall threshold of 5 degrees rotation error and 20 cm translation error.

For training models on cast parts data, we used both high overlap and low overlap scan pairs, as described in section III-B.1. The models were finetuned with the same parameters as in the original code, except from the parameters mentioned above.

The two test datasets consists of 235 high overlap pointcloud pairs and 433 low overlap pointcloud pairs from other physical parts than those seen during training.

C. Classical method

For comparison with a classical pointcloud feature extraction method, we use FPFH in combination with RANSAC, using the same parameters as described in the previous section. As this is not a learning-based method, the results are not affected by training data.

D. Evaluation metrics

Registration results are reported using 1) registration recall (RR), the fraction of successful registrations (with a transformation error smaller than a certain threshold) and 2) transformation error between estimated and ground truth transformations.

More specifically, the transformation error is defined as the relative error between the estimated transformation $\hat{\mathbf{T}}$ and the ground truth pose $\mathbf{T}$. We report it as Relative Rotation Error (RTE) and Relative Translation Error, which are defined as

$$\text{RRE} = \arccos(\frac{\text{trace}(\hat{\mathbf{R}}^T\mathbf{R}) - 1}{2}) \quad (1)$$

and

$$\text{RTE} = \|\hat{\mathbf{t}} - \mathbf{t}\| \quad (2)$$

where $\hat{\mathbf{t}}$ and $\hat{\mathbf{R}}$ are the estimated translation vector and rotation matrix, and $\mathbf{t}$ and $\mathbf{R}$ are the ground truth equivalents.

Following the definition in [4], the mean rotation and translation errors are computed using *only the point cloud pairs with a successful registration*, by the same definition as for registration recall. When interpreting the results, it should be noted that RRE and RTE only capture the smaller differences of the registrations that are considered a success, while the large errors are captured by the recall value. As mentioned, for the tunnel case, we set the threshold values at RTE < 0.3 m and RRE < 3 degrees. While we used a similar threshold values as for the 3DMatch data in [4] for the cast parts case: RTE < 0.2 m and RRE < 5 degrees.

V. Results

A. Tunnel Case

Registration results for the tunnel test case are shown in table I. We see that the FPFH approach is struggling with alignment and achieve a 5% registration recall. This was expected as the method rely on locally unique geometrical shapes, which are lacking in the tunnel. As for the pretrained learning based approaches, we see CoFiNet is completely failing with a 0.5% recall ratio. While GeoTransformers is also performing poorly it is outperforming FPFH with a 14.5% registration recall, indicating that this method is able to pick up some useful information from the scans.

When comparing with the results of both methods trained on the tunnel data we see a big improvement in performance, particularly GeoTransformer which has a registration recall of almost 60% on this challenging data.

One of the main challenges with this dataset is that most of the points represent only a smooth wall, and does not contribute to any distinct features useful for localizing the tunnel along its length axis. This is a possible explanation why the classical method and pretrained methods fail, as they are not sufficiently amplifying the sparse useful information contained in the scans. Two examples of successful registrations can be seen in fig. 4

TABLE I: Registration results tested on our tunnel (PILOTING) dataset, comparing a classical approach, pretrained models trained on the KITTI dataset and models trained on our PILOTING data.

Method - training data	***Mean RRE* [deg]**	***Mean RTE* [m]**	***RR***
FPFH - N/A	1.038	0.174	0.050
CoFiNet - Kitti	1.082	0.159	0.005
CoFiNet - PILOTING	0.788	0.114	0.458
GeoTransformer - Kitti	0.764	0.152	0.145
GeoTransformer - PILOTING	0.525	0.141	0.591

B. Cast parts

Results for registration on our cast parts data are shown in table II for the high overlap case and table III for the low overlap case. We see that this use case has an overall higher recall score than the tunnel use-case.

For the high overlap case, the best performing method is GeoTransformer finetuned on our data and with RANSAC pose estimation, with a registration recall of 0.93. In general, finetuning gives an increase in registration recall of around 0.1, which is expected. Even without finetuning, the end-to-end approach outperforms the others with a significant margin.

The low overlap case is more challenging, and without finetuning best recall is only 0.28 (with ransac in evaluation). CoFiNet gets a small improvement after finetuning, but GeoTransformer gets a significant performance increase to 0.73 (with RANSAC in evaluation). A closer inspection reveals that most of the errors are around 180 degrees, which corresponds to a flipped part. By looking at the *overall* rotation errors (not the recall errors reported by the standard metrics), we see that GeoTransformer (without RANSAC) has an *overall* mean rotation error of 127,3 degrees, which is reduced to 55,7 degrees after finetuning. There are also fewer examples of flipped parts after finetuning.

Visualizations of two example registration results are shown in fig. 3; one with high overlap (45 degree rotation) between the scans and one with low overlap (scanned from opposite sides), to illustrate typical errors for the two cases. For the pointclouds with high overlap, there is a slight angular error in the estimation, while for the pointclouds with low overlap, the estimated transformation has flipped the part upside down. This is because the two scans contain points from opposite sides of the symmetric part, which will give similar features, and an erroneous solution in the RANSAC step. This is a particularly challenging feature of this problem, which is different from for instance the 3DLoMatch benchmark, which contains cropped pointclouds from similar viewpoints.

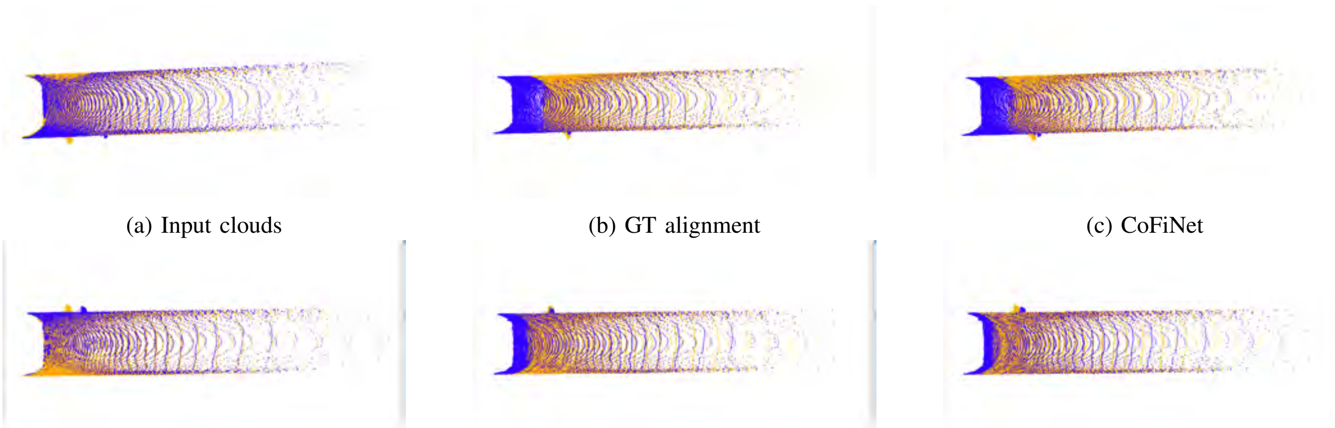
TABLE II: Registration results on our cast parts dataset, high overlap test case.

Method - training data	***Mean RRE* [deg]**	***Mean RTE* [m]**	***RR***
FPFH - N/A	2.316	0.031	0.664
CoFiNet - 3DMatch	1.734	0.028	0.696
CoFiNet - Cast parts	1.634	0.026	0.779
GeoTransformer end-to-end - 3DMatch	1.306	0.012	0.793
GeoTransformer w/RANSAC - 3DMatch	0.888	0.008	0.802
GeoTransformer end-to-end - Cast parts	0.723	0.0073	0.894
GeoTransformer w/RANSAC - Cast parts	0.843	0.009	0.930

VI. Discussion

We have in this work assessed the performance and generalizabillity of two state of the art in learning based registration methods, on two real world datasets which both present particular challenges.

The tunnel dataset contain scans with a high degree of similarity. In this dataset viewpoint dependent artifacts outnumber the subtle geometrical details containing the information



(a) Input clouds (b) GT alignment (c) CoFiNet

(d) Input clouds (e) GT alignment (f) CoFiNet

Fig. 4: Example registration results from two samples of the tunnel dataset, comparing ground truth to CoFiNet registration trained on the tunnel data. Note how the small protruding structures on the side of each sample are roughly aligned.

TABLE III: Registration results on our cast parts dataset, low overlap test case.

Method - training data	*Mean RRE* [deg]	*Mean RTE* [m]	*RR*
FPFH - N/A	2.920	0.0610	0.106
CoFiNet - 3DMatch	3.120	0.074	0.150
CoFiNet - Cast parts	2.628	0.0607	0.319
GeoTransformer end-to-end - 3DMatch	1.606	0.0176	0.213
GeoTransformer w/RANSAC - 3DMatch	2.193	0.0208	0.280
GeoTransformer end-to-end - Cast parts	0.892	0.0095	0.679
GeoTransformer w/RANSAC - Cast parts	1.427	0.0144	0.731

necessary for a successful registration. On the other hand, in the cast parts dataset, distinct features are abundant, but the particular symmetries of the part can be a source of noise when aligning scans.

When examining the tunnel data, it becomes evident that a model trained on the kitti dataset struggles to perform well in tunnel scenarios. This highlights the fact that learning-based registration remains largely influenced by the specific training data it has encountered. However, it's noteworthy that one of the two methods still surpasses handcrafted feature-based registration by nearly threefold, illustrating the promise of learning based methods. Furthermore, despite both the kitti datset and the tunnel dataset was captured using a 360-degree rotating lidar, the significant performance enhancements that were observed when training the models specifically on tunnel data could be attributed the the network parameters adapting to the particularities of this kind of environment.

As for the cast parts dataset, we see that all approaches perform well on the large overlap scans, but that both models has a superior performance when they are trained on similar data. It is also interesting to note that in contrast to the findings in [19] we see that GeoTransformer perform better when we use a RANSAC based outlier rejection scheme then if we use the learned scheme embedded in the end to end method. For the low overlap dataset, we see a significant improvement when training both models on the cast parts dataset, compared to the pretrained model, but we also see a much improved performance of Geotranformers over CoFiNet. A possible explanation could be that training in an end-to-end manner (as in GeoTransformer) results in features that are more specific for the registration problem. This enables the model to optimize the features to give a small registration error on this challenging use-case, which requires features with more global context the original 3DMatch case due to symmetries and occlusions. Even though strong features are found by the pretrained model, these lead to erroneous matching due to the symmetric properties of the cast parts.

VII. Conclusion

In summary, while learning-based pointcloud registration methods are demonstrating enhanced performance, even in difficult scenarios such as tunnels and symmetrical cast iron parts, their specificity becomes evident when evaluated with out-of-sample or out-of-distribution data. To address this limitation, further refinement of these approaches or the implementation of a more inclusive training regimen will be essential for these models to effectively accommodate a broader range of domains. There is also a need for more challenging large-scale open datasets and benchmarks that address challenges specific to real-world use-cases and realistic 3D sensor setups.

References

[1] Xiaoshui Huang, Guofeng Mei, Jian Zhang, and Rana Abbas, "A comprehensive survey on point cloud registration," *arXiv preprint arXiv:2103.02690*, 2021.

[2] Yulan Guo, Mohammed Bennamoun, Ferdous Sohel, Min Lu, Jianwei Wan, and Ngai Ming Kwok, "A comprehensive performance evaluation of 3d local feature descriptors," *International Journal of Computer Vision*, vol. 116, pp. 66–89, 2016.

[3] Radu Bogdan Rusu, Nico Blodow, and Michael Beetz, "Fast point feature histograms (fpfh) for 3d registration," in *2009 IEEE international conference on robotics and automation*. IEEE, 2009, pp. 3212–3217.

[4] Hao Yu, Fu Li, Mahdi Saleh, Benjamin Busam, and Slobodan Ilic, "CoFiNet: Reliable Coarse-to-fine Correspondences for Robust Point-Cloud Registration," in *Advances in Neural Information Processing Systems*. 2021, vol. 34, pp. 23872–23884, Curran Associates, Inc.

[5] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, Slobodan Ilic, Dewen Hu, and Kai Xu, "Geotransformer: Fast and robust point cloud registration with geometric transformer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

[6] Christopher Choy, Jaesik Park, and Vladlen Koltun, "Fully convolutional geometric features," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 8958–8966.

[7] Christopher Choy, Wei Dong, and Vladlen Koltun, "Deep global registration," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 2514–2523.

[8] 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*, 2022, pp. 11143–11152.

[9] Paul J Besl and Neil D McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*. Spie, 1992, vol. 1611, pp. 586–606.

[10] Yue Wang and Justin M Solomon, "Prnet: Self-supervised learning for partial-to-partial registration," *Advances in neural information processing systems*, vol. 32, 2019.

[11] 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*, 2017, pp. 1802–1811.

[12] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[13] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao, "3d shapenets: A deep representation for volumetric shapes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 1912–1920.

[14] "PILOTs for robotic INspection and maintenance Grounded on advanced intelligent platforms and prototype applications | PILOTING Project | Fact Sheet | H2020," .

[15] Peter Ivarsen, Jan Sramota, Martin Gerhardsen, Henrik Lundqvist, and Richard Moore, "Multi-modal slam for accurate localisation in self-similar environments," 11 2023, pp. 302–307.

[16] Ahmed Mohammed, Johannes Kvam, Ingrid Fjordheim Onstein, Marianne Bakken, and Helene Schulerud, "Automated 3D burr detection in cast manufacturing using sparse convolutional neural networks," *Journal of Intelligent Manufacturing*, vol. 34, no. 1, pp. 303–314, Jan. 2023.

[17] A Geiger, P Lenz, C Stiller, and R Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, Sept. 2013.

[18] Andy Zeng, Shuran Song, Matthias Niessner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser, "3DMatch: Learning Local Geometric Descriptors from RGB-D Reconstructions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, July 2017, pp. 199–208, IEEE.

[19] Zheng Qin, Hao Yu, Changjian Wang, Yulan Guo, Yuxing Peng, Slobodan Ilic, Dewen Hu, and Kai Xu, "GeoTransformer: Fast and Robust Point Cloud Registration With Geometric Transformer," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 8, pp. 9806–9821, Aug. 2023, Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence.