Empir3D : A Framework for Multi-Dimensional Point Cloud Assessment

Supplementary Material

The supplementary material extends the evaluation and discussion provided in the main manuscript. We show, in detail the efficacy and viability of using Empir3D as a general-purpose point cloud quality evaluation framework. The supplementary material is organized as follows, subsection I-A has additional observations about the SLAM experiments in section IV of the main paper including our method of collecting ground truth in the real world and the details about simulations, subsection I-B demonstrates how Empir3D can be used in learning pipelines further and subsection I-C describes the compute performance including memory and CPU utilization in details. Finally, in section II we discuss how Empir3D can be used example applications.

I. EVALUATION

A. Evaluation on Dense SLAM

Figure 3 shows different views of the *Warehouse* dataset, including magnified views for qualitative analysis. These images clearly show that SHINE and FAST-LIO2 significantly outperform LeGO-LOAM. Empir3D values corroborate with qualitative results by identifying SHINE's point cloud as the one with highest Q_c, Q_r, and Q_a whereas D_c identifies FAST-LIO2. Of note, SHINE has a lower artifact-score (compared to FAST-LIO2) indicating the presence of artifacts, observed in Figure 3 where the pillars appear distorted due to artifacts. Further, Empir3D also accurately quantifies the lower resolution in LeGO-LOAM compared to others.

1) Constructing ground truth point clouds: Acquiring ground truth for point cloud evaluation in a real-world setting is a challenging task. Although CAD and BIM files can be used, they're generally outdated and seldom represent the current environment. In outdoor settings, topographical maps and digital elevation models (DEMs) are sparse and lack rich 3D information. A common practice to build dense 3D models is the use of survey grade LiDAR scanners [1, 5], these use ground truth poses (from RTK GPS or Total-Stations) along with LiDAR scans to stitch a high-resolution 3D representation of the environment.

We used a similar approach to build ground truth point clouds. First, we capture ground truth poses using a Robotic Total-Station [4] which is essentially a theodolite with an integrated distance meter that can measure distances and angles. This enables extremely precise pose estimation (millimeter-level) and is widely used in infrastructure and geospatial surveying [2, 8]. Second, we capture LiDAR scans at the exact locations where poses are measured. This is achieved by placing a LiDAR mounted on a custom tripod, the tripod is equirped with a nadir pointed laser that projects a cross-hair onto carefully placed markers on the ground to align scans. For the *Davis* dataset (section IV in the main paper), over 400 poses (Figure 1) and corresponding scans were captured with this method. Finally, the scans are stitched into a dense point cloud of the environment with the recorded poses providing initial alignment. ICP was used for fine registration.
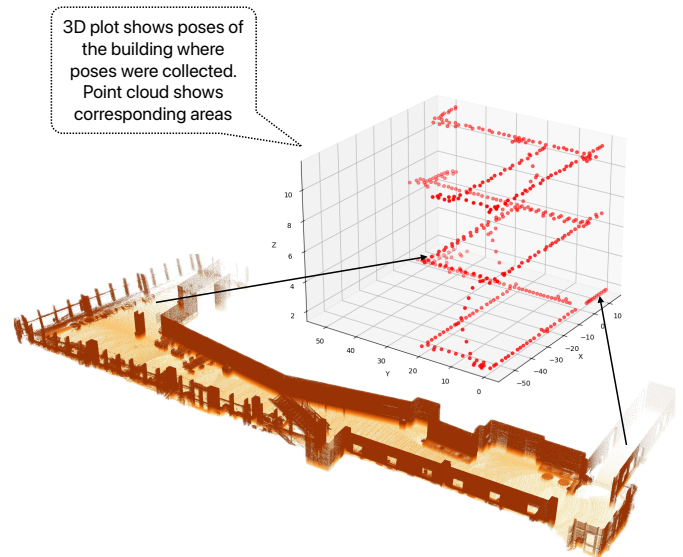

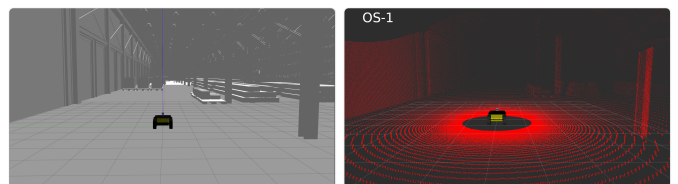**Fig. 1:** Poses and point cloud for *Davis* dataset

Fig. 2: *Warehouse* environment in simulation. Robot third-person view (left) and sensor output (right)

2) Simulation environment: The simulator (Figure 2) built to study Empir3D's performance supports various environments, these are mesh files of objects and structures and can be built using popular tools like Blender [3]. The simulator supports three variants of the Ouster OS series LiDARs,
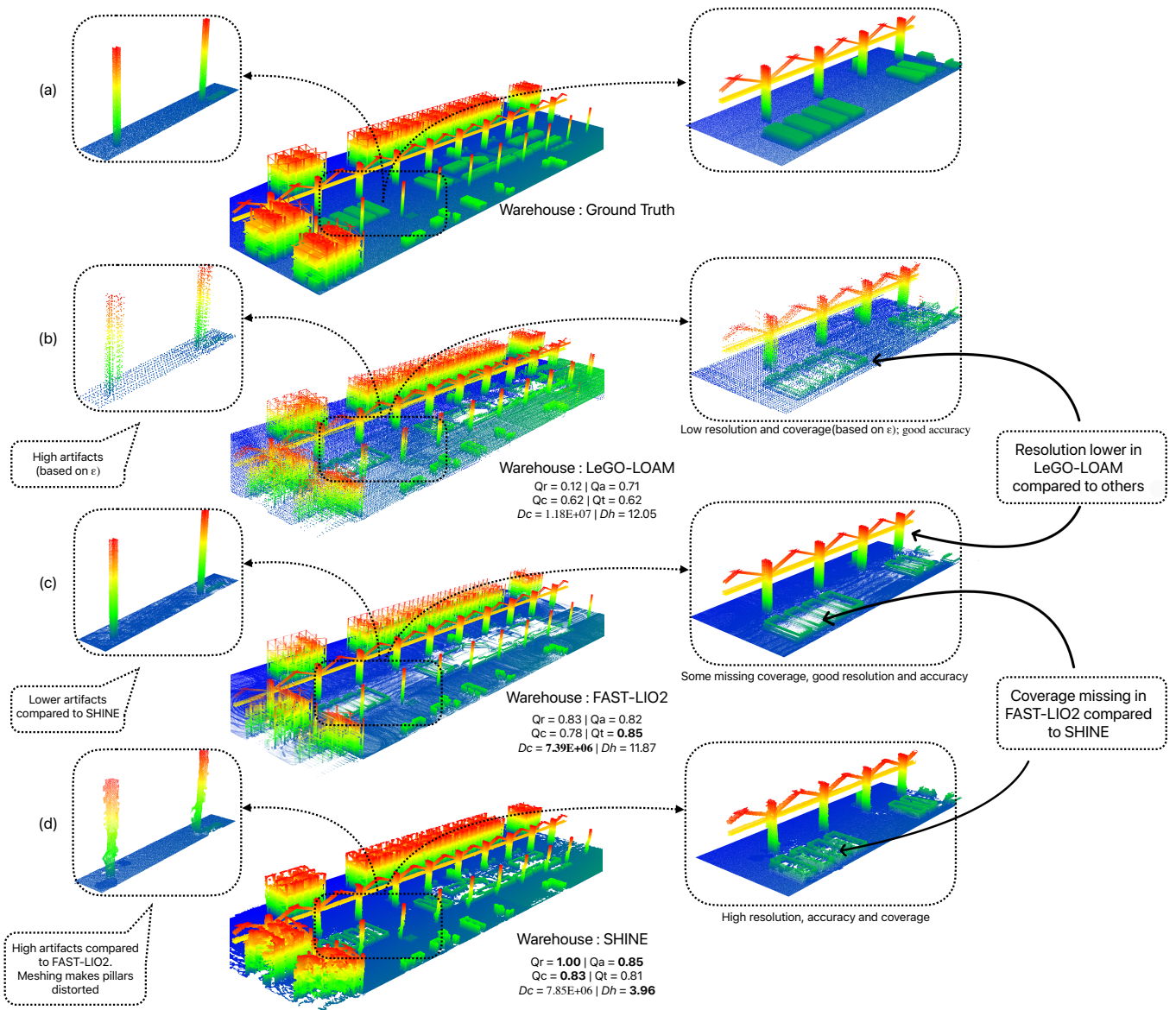
Fig. 3: Evaluation on *Warehouse* dataset (simulated); (a),(b),(c),(d) show point clouds ground truth, LeGO-LOAM, FAST-LIO2 and SHINE. (d) exhibits highest Q_c, Q_a, Q_a as shown in zoomed-in view but contains artifacts (lower Q_t compared to (c)). Empir3D accurately quantifies all aspects of quality while D_c and D_h identify (c) and (d) as the most similar

namely the OS-0, OS-1 and OS-2 sensors. These allow a max vertical field-of-view of 90, 45 and 22.5 degrees respectively. Each of the sensor can be configured in three resolutions - 128, 64 and 32 channels. This totals to nine distinct LiDARs in simulation. We also add noise to simulate realistic outputs which can be tuned to match the sensor. Additionally, the simulator outputs IMU data and ground truth odometry for use in SLAM methods that need IMU or need external pose information.

B. Evaluation on point cloud completion

Figure 4 shows evaluation of point clouds generated using three point cloud completion networks, namely, PCN [10], TOPNET [9] and ECG [6]. PCN and ECG output rela-tively high quality point clouds while TOPNET lags behind, producing noisy and incomplete results. These observations are corroborated by Empir3D's metrics which denote ECG's output as the one with highest quality while D_c and D_h show PCN as the most similar. At the same time, D_h indicates lower similarity for ECG's point clouds contradicting visual observations and other metrics. This illustrates D_c and D_h's limitations in quantitative quality evaluation.

C. Compute Performance

Empir3D compute performance is evaluated by calculating quality metrics for a sample point cloud. We compute Empir3D for a wide range of region sizes while D_c and D_h are calculated on the entire point clouds. Region sizes of 1, 2,
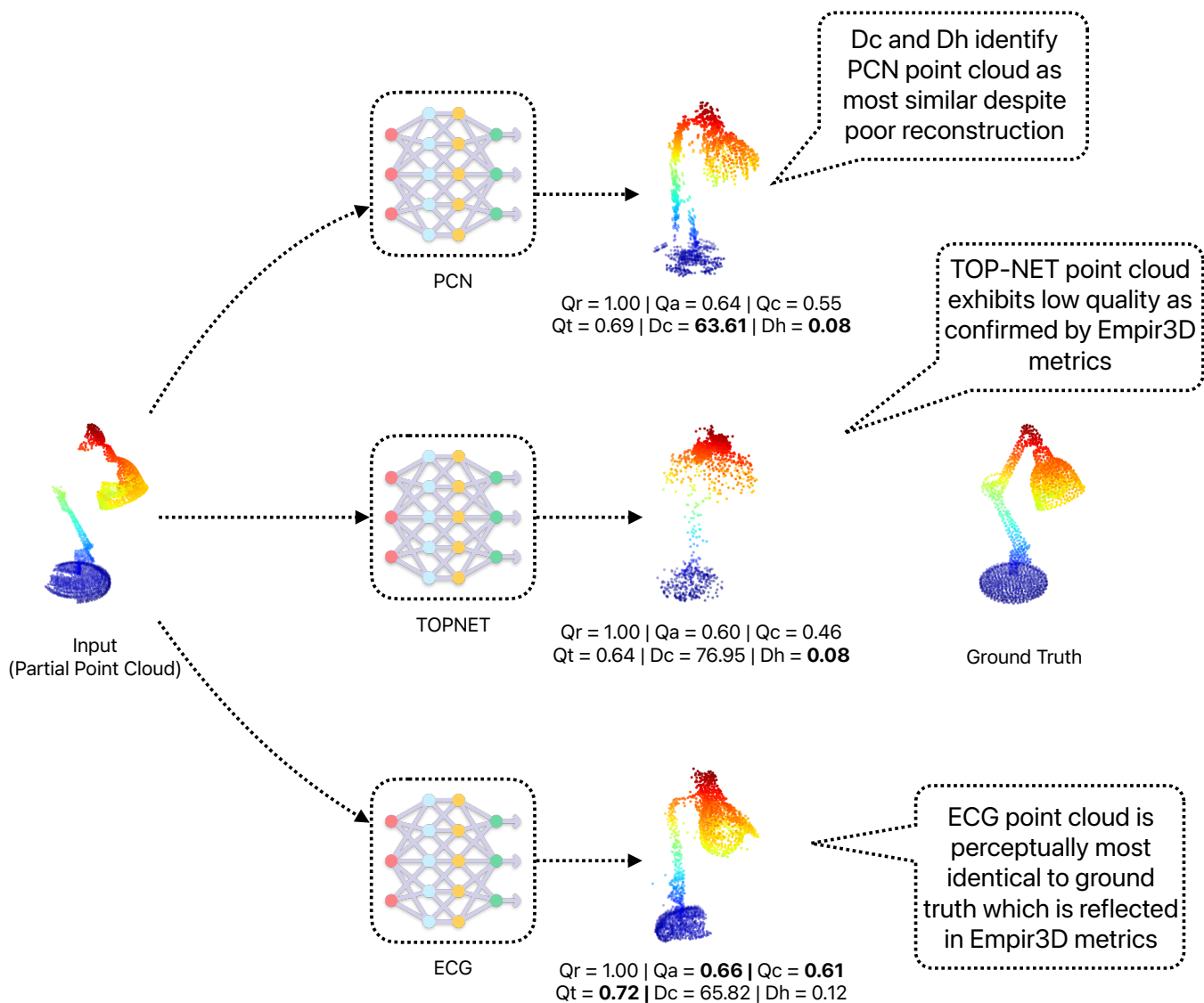
Fig. 4: Evaluation on point cloud completion, Lamp from MVP [7] reconstructed using PCN [10], TOPNET [9], ECG [6]. ECG generates highest quality point clouds which is confirmed by Empir3D metrics while D_c and D_h identify PCN despite presence of artifacts and poor reconstruction.

5, 10, 15, 20, 30, 50, 100 meters are chosen and run-time along with CPU and memory utilization is measured. Further, we repeat the experiments by limiting number of CPU cores for multi-threading providing some insights on computing on resource constrained hardware.

Using D_c and D_h as baselines for reference, we observe that their CPU utilization is low and correspondingly the time taken to complete the evaluation is high. In contrast, our implementation of Empir3D, trades CPU utilization for execution time. In Figure 5, we demonstrate this trade-off along with the effect of choosing different region sizes for parallelization on a benchmark point cloud. We observe that for smaller region size configuration, the CPU utilization is higher. As the region size is increased, the utilization dips as the region size grows closer to the size of the point cloud.

Eventually, Empir3D's parallel implementation will reach a CPU utilization parity with D_c and D_h where entire point clouds are loaded on a single thread.

On the other hand, we observe a trend in the execution time that might appear unusual at first glance. The time taken by Empir3D starts to decrease with increase in region size but rises again for even higher values. We note that the reasons for this effect are two-fold. With much smaller region sizes, the individual region comparisons are quick. The major reason for this is that there are fewer points to consider for the closest point match. But, there are many regions to compute leading to a longer time. As we increase the region size, the number of regions to compare reduces giving Empir3D a performance boost. At the same time, the time taken for the closest point search for individual regions increases. Finally,
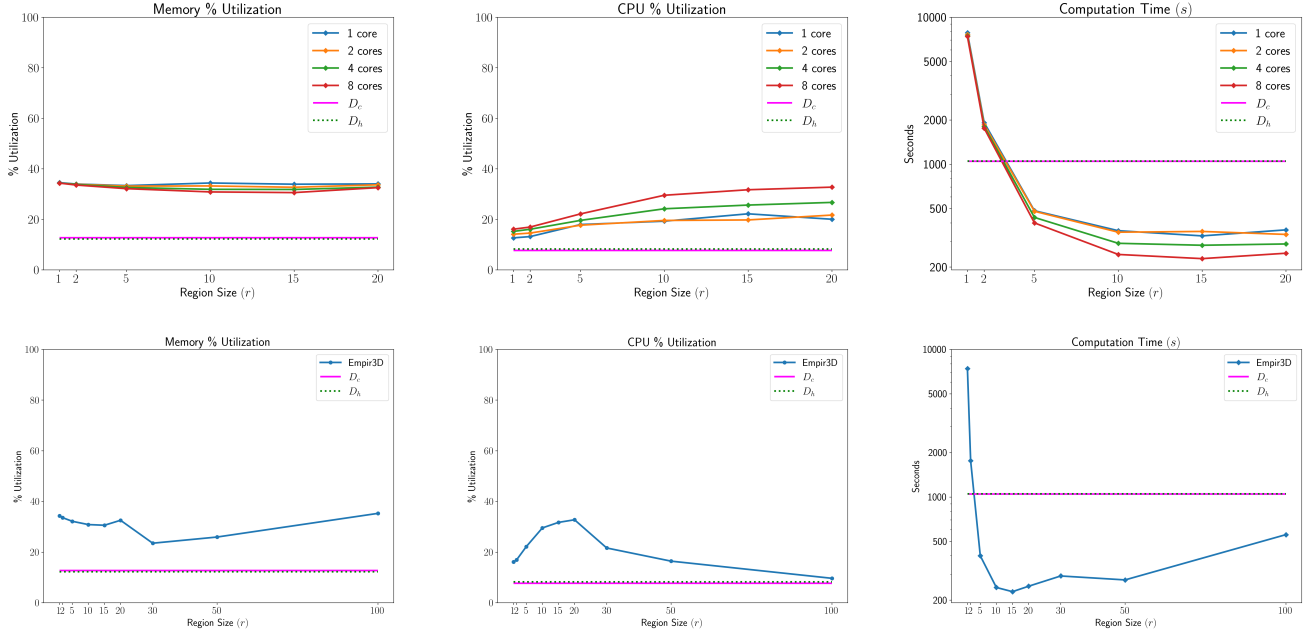
Fig. 5: Comparative analysis of memory, CPU utilization and computation time across different region sizes (r) and number of processor cores.1 Memory utilization remains relatively constant across various core counts, whereas CPU utilization increases with more cores, especially for smaller region sizes before decreasing again when regions become too large for multi-threading. Computation time decreases significantly with the increase in region size up to a certain point, beyond which the benefits plateau, as shown in the benchmarks for 1, 2, 4, and 8 core computation of Empir3D, compared to the baseline single-threaded implementations D_c and D_h.

as the region size grows close to the full map, the time taken for closest point matches dominates leading to longer execution times. We hope that this analysis will help our readers make a judicious choice for the region size parameter for parallelization. Further, we note that as an added benefit of using lower region sizes is the lower memory usage.
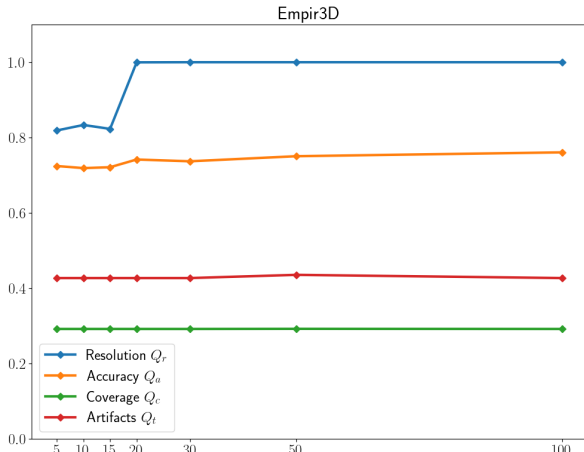

Fig. 6: Plot shows change in Empir3D metrics with increase in region size (r). Resolution and accuracy are affected by averaging while coverage and artifact-score are not.

Since Empir3D metrics are averaged, a choice of region size will affect the actual values of some metrics (Figure 6).

While coverage and artifact-score are unaffected by region size, accuracy and resolution are affected by the averaging. However, this should not affect comparisons between different point clouds as long as the same region size is used for the comparisons.

Region size affects CPU usage (thus time) and the memory usage. From our analysis, we make these recommendations:

- Use lower regions sizes on memory constrained devices
- Use higher region sizes in the order of map size on compute constrained devices
- On unconstrained devices, use the region size to split the map into as many chunks in the order of the number of threads available on the device

II. APPLICATIONS AND USE-CASES

To illustrate Empir3D's usage, consider the problem of selecting a suitable LiDAR sensor for dense mapping. Given Ouster OS-0 and OS-1 LiDARs, with vertical FOV of 90 and 45 degrees respectively, a range of 35m and 90m respectively and a vertical resolution of 128 channels, we simulate both sensors in our simulator (Figure 2) and map the *Warehouse* environment using FAST-LIO2. Figure 7 shows the point clouds and corresponding Empir3D metrics. FAST-LIO2 with OS-0 generated pcd_{OS-0} and OS-1 generated pcd_{OS-1}.

pcd_{OS-0} has higher coverage at 90.5% compared to 89.36% in pcd_{OS-1}, this is primarily due to the larger V-FOV of the OS-0 LiDAR. pcd_{OS-1} on the other hand, has higher resolution at 97.86% compared to 95.02% in pcd_{OS-0}, this
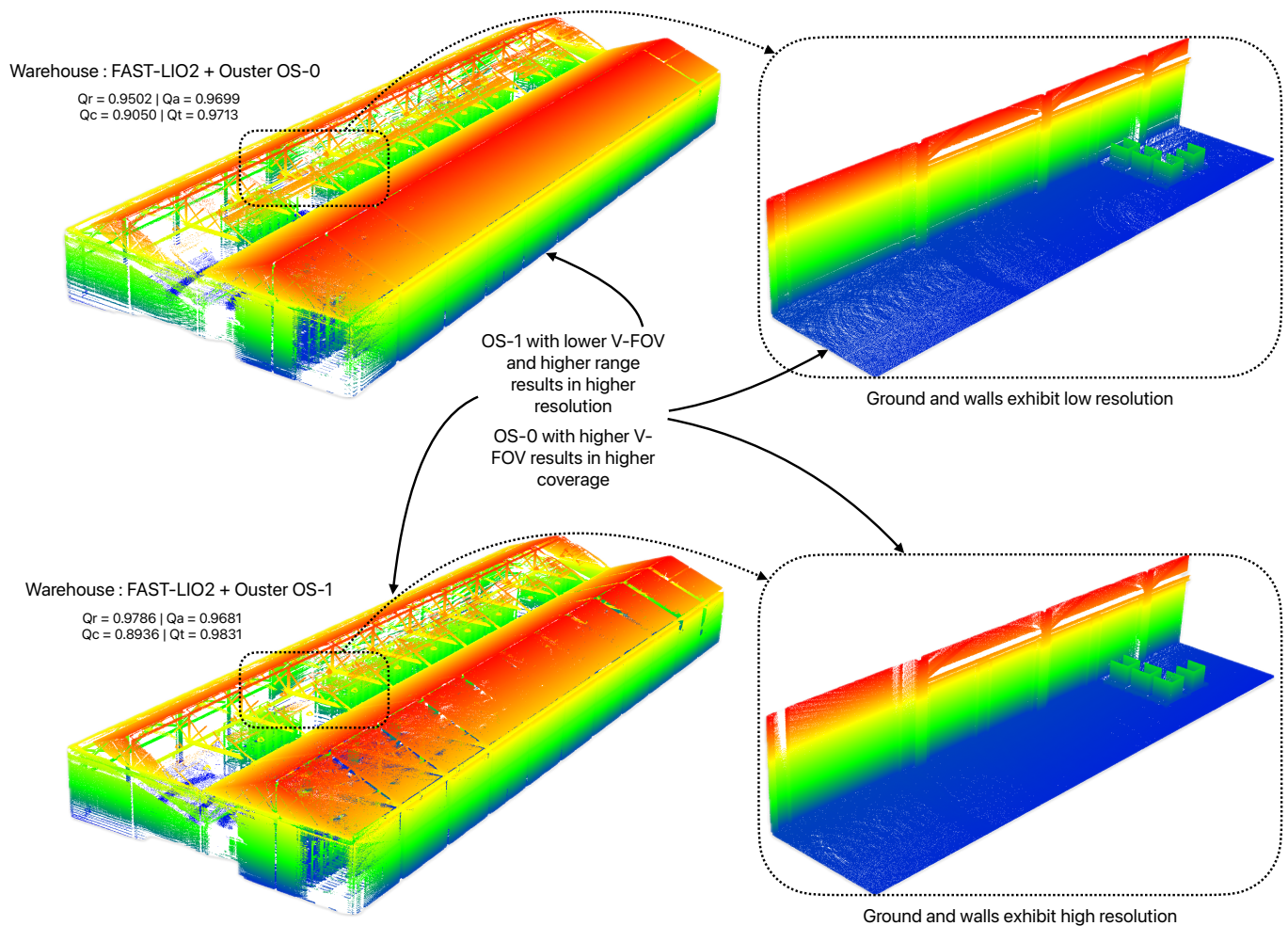
Fig. 7: Choosing sensors based on coverage and resolution; study shows Empir3D used to quantify performance of two LiDARs, namely the Ouster OS-0 and OS-1 128 channel. When *Warehouse* environment is mapped using FAST-LIO2 with OS-0 the resulting point cloud exhibits higher overall coverage thanks to the increased vertical FOV while OS-1's point cloud is higher resolution increased vertical scan density.

can be attributed to the spreading out of the 128 available vertical channels over different FOVs that produce different densities. Results indicate that the OS-1 is a better choice in this case due to the higher resolution in the maps despite the expanded coverage of OS-0.

This style of assessment can help pick sensors and algorithms based on expected coverage and resolution. Higher coverage and resolution generally results in improved SLAM performance, both in localization and mapping. This is because an increase in these metrics indicates an increase the number of features detected in the point cloud. To demonstrate this further, we compute ISS features [11] on both pcd_{OS-0} and pcd_{OS-1}. pcd_{OS-0} contains 1,531,103 features while pcd_{OS-1} contains 1,509,107, this confirms the hypothesis that an increase of 1.14% in coverage results in 1.45% increase in the number of points detected.

REFERENCES

[1] Yusuf Arayici. An approach for real world data modelling with the 3d terrestrial laser scanner for built environment. *Automation in Construction*, 16(6):816–829, 2007. ISSN 0926-5805. doi: https://doi.org/10.1016/j.autcon.2007.02.008. URL https://www.sciencedirect.com/science/article/pii/S0926580507000350.

[2] Solomon Dargie Chekole. Surveying with gps, total station and terresterial laser scaner: a comparative study, 2014.

[3] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. URL http://www.blender.org.

[4] U Kizil and L Tisor. Evaluation of rtk-gps and total station for applications in land surveying. *Journal of Earth System Science*, 120:215–221, 2011.

[5] G. Mandlburger, M. Kölle, F. Pöppl, and M. Cramer. Evaluation of consumer-grade and

survey-grade uav-lidar. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLVIII-1/W3-2023: 99–106, October 2023. ISSN 2194-9034. doi: 10.5194/isprs-archives-xlviii-1-w3-2023-99-2023. URL http://dx.doi.org/10.5194/isprs-archives-XLVIII-1-W3-2023-99-2023.

[6] Liang Pan. ECG: Edge-aware Point Cloud Completion with Graph Convolution. *IEEE Robotics and Automation Letters*, 5(3):4392–4398, July 2020. ISSN 2377-3766, 2377-3774. doi: 10.1109/LRA.2020.2994483. URL https://ieeexplore.ieee.org/document/9093117/.

[7] Liang Pan, Xinyi Chen, Zhongang Cai, Junzhe Zhang, Haiyu Zhao, Shuai Yi, and Ziwei Liu. Variational Relational Point Completion Network. 2021. doi: 10.48550/ARXIV.2104.10154. URL https://arxiv.org/abs/2104.10154. Publisher: arXiv Version Number: 1.

[8] Aivars Alt Tarvo Mill and Roode Liias. Combined 3d building surveying techniques – terrestrial laser scanning (tls) and total station surveying for bim data management purposes. *Journal of Civil Engineering and Management*, 19(sup1):S23–S32, 2013. doi: 10.3846/13923730.2013.795187. URL https://www.tandfonline.com/doi/abs/10.3846/13923730.2013.795187.

[9] Lyne P. Tchapmi, Vineet Kosaraju, Hamid Rezatofighi, Ian Reid, and Silvio Savarese. TopNet: Structural Point Cloud Decoder. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 383–392, Long Beach, CA, USA, June 2019. IEEE. ISBN 978-1-72813-293-8. doi: 10.1109/CVPR.2019.00047. URL https://ieeexplore.ieee.org/document/8953650/.

[10] Wentao Yuan, Tejas Khot, David Held, Christoph Mertz, and Martial Hebert. PCN: Point Completion Network. In *2018 International Conference on 3D Vision (3DV)*, pages 728–737, Verona, September 2018. IEEE. ISBN 978-1-5386-8425-2. doi: 10.1109/3DV.2018.00088. URL https://ieeexplore.ieee.org/document/8491026/.

[11] Yu Zhong. Intrinsic shape signatures: A shape descriptor for 3d object recognition. In *2009 IEEE 12th international conference on computer vision workshops, ICCV Workshops*, pages 689–696. IEEE, 2009.