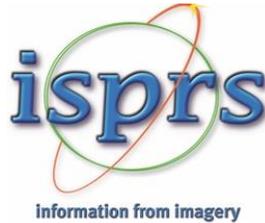# ISPRS Scientific Initiatives 2025 Final Report

PC2Model:
ISPRS Benchmark on 3D Point Cloud to Model Registration

Mehdi Maboudi[1], Kourosh Khoshelham[2], Yelda Turkan[3], Karam Mawas[1]

[1] Institute of Geodesy and Photogrammetry, Technische Universität Braunschweig, Germany
m.maboudi@tu-braunschweig.de

[2] Department of Infrastructure Engineering, University of Melbourne, Australia.
k.khoshelham@unimelb.edu.au

[3] Civil & Construction Engineering, Oregon State University, USA.
yelda.turkan@oregonstate.edu

Summary

Point cloud registration is the task of aligning one point cloud with another or with a three-dimensional (3D) model, enabling the integration of multimodal data into a unified representation. This is essential in applications such as construction monitoring, autonomous driving, robotics, and virtual or augmented reality (VR/AR). As point cloud acquisition technologies like Light Detection and Ranging (LiDAR), and structured light scanning, have become more accessible alongside recent advances in deep learning, research focus has shifted toward downstream tasks, particularly point cloud-to-model registration. While data-driven methods aim to automate this process, they face challenges such as sparsity, noise, clutter, and occlusions, which limit performance. The PC2Model benchmark is introduced to address these challenges. Developed under the leadership of ICWG II/I b the PC2Model benchmark provides a publicly available dataset for 3D point cloud-to-model registration[1]. It adopts a hybrid design, combining simulated point clouds with, in some cases, real-world scans and their corresponding 3D models. Simulated data offer precise ground truth and controlled conditions, while real-world data introduce sensor and environmental artifacts. This setup supports robust training and evaluation across domains and enables systematic analysis of model transferability from simulated to real-world scenarios.

---

[1] https://doi.org/10.5281/zenodo.17581812

Project goals

The overall aim of the PC2Model benchmark dataset is to provide a diverse and representative resource for training and evaluating machine and deep learning models for 3D point cloud processing, with a particular emphasis on point cloud-to-model registration. The dataset enables systematic evaluation of conventional optimization-based methods, such as iterative closest point (ICP), as well as investigation of machine learning approaches aimed at improving registration accuracy, robustness, and computational efficiency. In addition, the inclusion of real-world data supports validation studies to assess whether performance gains achieved using simulated data transfer effectively to real-world scenarios. This combination of simulation-based training and real-world validation is designed to facilitate the development of robust and generalizable registration solutions.

Dataset

The PC2Model benchmark dataset comprises both simulated and real-world point clouds organized into seven categories. Six categories are synthetically generated, and one category consists of real-world data, as summarized in

Table 1 along with the corresponding number of samples. Each sample includes a 3D model, the corresponding point cloud generated by   simulating the scanning of that model and applying a rigid transformation to the scan, and the associated transformation matrices.

Table 1: Categories and number of samples

Category	Number of samples
Mechanical objects (simulated)	25
Furniture (simulated)	25
Home décor (simulated)	25
Houses (simulated)	25
Vehicles (simulated)	25
Indoor spaces (simulated)	6
Indoor spaces (real dataset[2])	6

Simulation software

The 3D models to generate simulated point clouds were sourced from a range of publicly available datasets, including ABC (Koch et al. 2019), ModelNet (Wu et al. 2015), Fusion360 (Willis et al. 2022), and Thingi10k (Zhou and Jacobson 2016), as well as user-contributed, open-licensed 3D models hosted on the Sketchfab[3] platform. In addition, for real-world scans, the ISPRS indoor

---

[2] ISPRS indoor modeling benchmark: https://dpv.uvigo.es/index.php/s/edzBCDDGkHjSWGC
[3] https://sketchfab.com

modelling dataset (Khoshelham et al. 2017) was included in the benchmark. All models were carefully curated to satisfy the diversity and quality requirements of the PC2Model dataset.

To simulate the scanning of the 3D models, Helios++ (Winiwarter et al. 2022) was selected from several available simulation tools. Helios++ provides high fidelity recreation of laser-scanning environments, producing point clouds that closely resemble real-world scans. The software is open source and supports the import of custom 3D models in Wavefront Object (OBJ) format. In combination with configurable simulation parameters and transparent file handling, these features enable realistic scan behavior at a manageable computational cost.

Helios++ supports simulations for terrestrial, airborne, and mobile laser scanners, in this project, terrestrial scanning was used. Among the available scanning methods, polygonal mirror deflection was selected. A key feature of Helios++ is its waveform-based laser beam simulation. Instead of modeling beams as straight lines, beam footprints are represented with a finite spatial extent, and reflected energy is integrated to derive surface characteristics. Beam divergence is modeled using multiple sub-rays sampled around a central ray, each of which is simulated individually, enabling control over parameters that affect scanning accuracy.

Ray tracing is used to compute intersections between rays and scene geometry, and additional noise sources, including range measurement noise and sensor position uncertainty, are incorporated to generate more realistic point clouds. Leveraging these capabilities, a virtual scanner emulating the *Leica ScanStation P40* was implemented in Helios++. The scanner specifications were derived from the datasheet and are summarized in Table 2. To achieve a uniform sampling density, the identical vertical and horizontal angular resolutions were set to the same value of 0.174°. Accordingly, the head rotation speed was set to 13.05 m/s, the pulse frequency to 125,000 Hz, and the scan frequency to 75 Hz.

Table 2: Simulated Leica ScanStation P40 scanner settings

Property	Value	Unit
Accuracy	0.0012	m
Beam divergence	0.00023	rad
Maximum head rotation speed in	18.0	°/s
Pulse frequencies	125000, 250000, 500000, 1000000	Hz
Pulse length	0.75	ns
Minimum scan range	0.4	m
Maximum vertical scan angle	145	°
Wavelength	1550	nm
Minimum scan frequency	4	Hz
Maximum scan frequency	100	Hz

Blender add-on

The setup of the simulation environments includes precise placement, translation, and scaling of the 3D objects, as well as careful positioning of the laser scanner around, and where applicable

within, the objects. For large-scale dataset generation, an efficient and intuitive workflow with visual verification was necessary. To address this, a Blender[4] add-on was developed that enables configuration of Helios++ scanning environments directly within Blender's viewport. Using the add-on, scanners can be positioned accurately at appropriate locations, and scanner parameters can be interactively adjusted while the resulting angular resolutions are displayed immediately. Upon execution of a simulation, the add-on automatically generates the required scene and survey files in Extensible Markup Language (XML) format, which serve as inputs to Helios++. After the simulation is completed, the point clouds from individual scanners are converted to ASTM E57 (E57) format, with scanners location embedded in the files. These point clouds are then merged into a single point cloud containing the xyz-coordinates and scanner positions. Finally, all output files are organized according to the PC2Model directory structure.

The add-on is maintained on GitHub, and includes the scripts used to compute the dataset statistics: https://github.com/saidharb/PC2Model

Scanning strategy

A consistent set of scanning principles was applied across all categories in the dataset. When importing OBJ files into Blender, scanning strategies were adapted based on object type. For mechanical objects, furniture, vehicles, and home décor items, scanners were placed on the ground plane around the objects. As a result, upper surfaces may not always be fully captured. These objects were intentionally not scaled to their real-world dimensions in order to introduce additional geometric and scale variability into the dataset.

For houses and indoor scenes, models were scaled to approximately reflect real-world dimensions. Houses vary in their level of furnishing, with some containing fully furnished interiors and others consisting of empty rooms; however, a consistent scanning strategy was applied in all cases. Scanners were positioned at approximately tripod height (1.5 m to 2 m) above ground level within each room. When visually accessible areas such as roofs were present (e.g., through doors or openings), additional scanners were placed to capture these regions. Exterior walls of houses were scanned by placing additional scanners at ground level outside the structures.

For indoor scene models, scanners were placed within each room, but no exterior scanning was performed; consequently, exterior walls were not captured for this category. Real-world indoor point clouds were sourced from the ISPRS indoor modeling benchmark (Khoshelham et al. 2017) and consist of six buildings. A visual overview of representative samples of each category is shown in Figure *1*.

---

[4] https://www.blender.org

3D models   Simulated point clouds
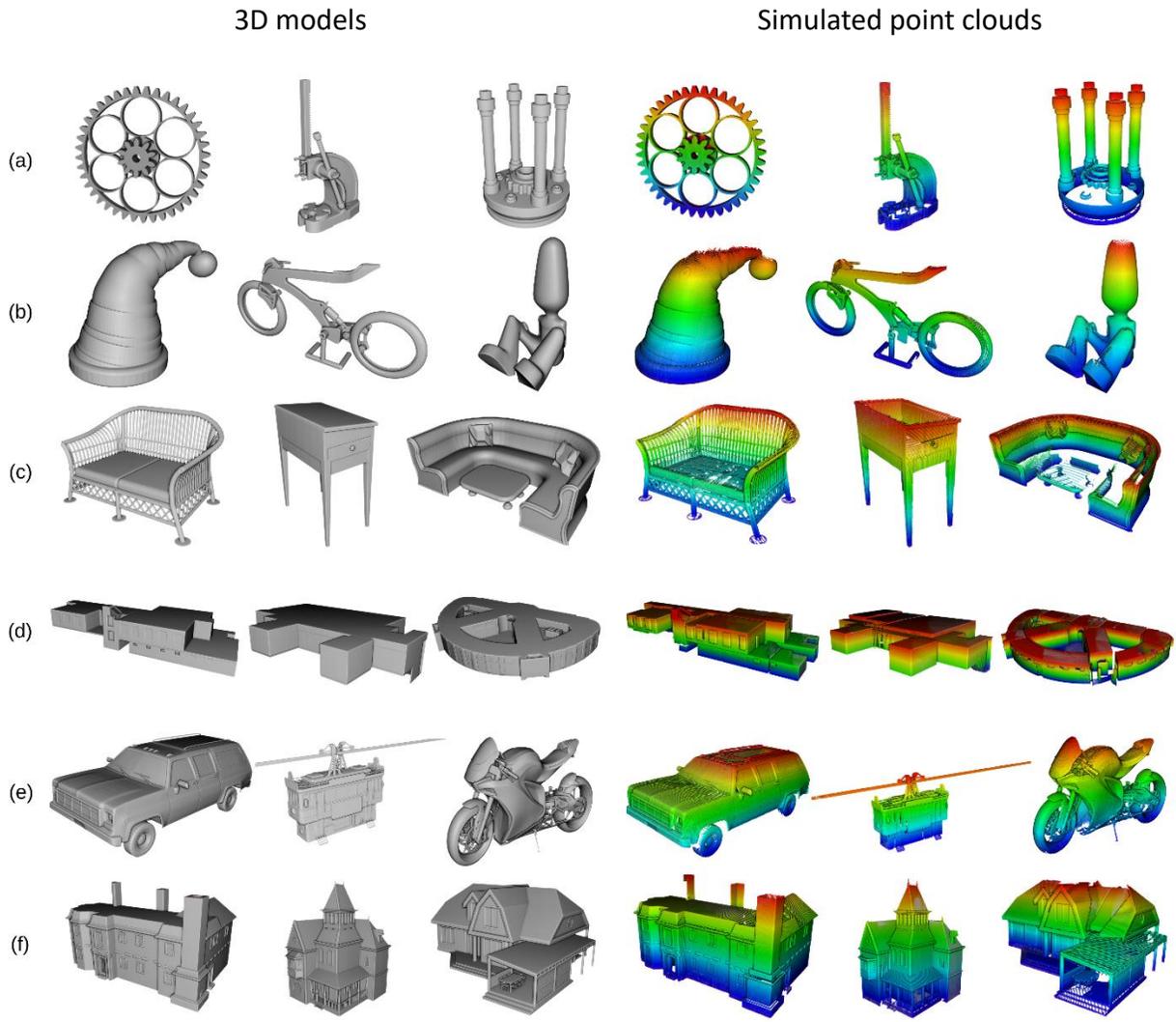
(a)

(b)

(c)

(d)

(e)

(f)

Figure 1: Selected examples from the dataset. a) mechanical objects, b) home décor, c) furniture, d) simulated indoor spaces (real point cloud data is also available via ISPRS indoor modeling benchmark), e) vehicles, f) houses

Dataset characteristics and preprocessing

Summary statistics of the dataset are presented in Table 3. Houses and indoor scenes constitute the largest objects with the most complex geometry, therefore, they typically require a greater number of scanner positions, contain a larger number of points, and exhibit higher average median point densities. In contrast, object-level categories, mechanical objects, furniture, vehicles, and home décor, generally require fewer scanners to achieve adequate coverage, as these objects are smaller in scale and present fewer occlusions than multi-room, multi-level structures. Because laser beams emitted within enclosed spaces inevitably intersect with surrounding surfaces, houses and indoor scenes exhibit substantially higher point densities than object categories, which are scanned in open environments where only beams intersecting the object contribute to the point count. Among all categories, furniture scans exhibit the lowest point densities.

Point density was computed by constructing a spherical neighborhood with the radius r = 0.5 (in model unit) around each point in the point cloud and counting the number of neighboring points within this volume. The resulting count was normalized by the volume of the sphere to obtain a local point density estimate. For most house-scale datasets, the radius can be interpreted in meters; however, this interpretation does not necessarily apply to object level categories, which were not uniformly scaled to real-world dimensions.

Table 3: Dataset statistics. All scans were performed with simulated terrestrial laser scanners, except for the real-world scans, which were taken with a mobile laser scanner.

Category	Average # scan positions	Average number of points	Average point densities ($r = 0.5$)
Mechanical objects (Synthetic)	6	189,560	5,427
Furniture (Synthetic)	6	221,275	67
Home décor (Synthetic)	5	244,270	4,881
Houses (Synthetic)	19	24,823,458	47,542
Vehicles (Synthetic)	4	277,288	19,228
Indoor spaces (Synthetic)	13	21,009,545	43,872
Indoor spaces (real)	N/A	19,507,569	30,127

All provided point clouds underwent a rigid body transformation, which serves as the ground truth for point cloud-to-model registration algorithms. The transformed point clouds, the original 3D models, and the corresponding homogeneous transformation matrices are provided for the training set to support the training and fine-tuning of data-driven methods.

The transformations were applied as follows. For each sample, the original 3D model was first rotated around the x-, y-, and z-axes using independent random angles uniformly sampled between 0° and 360°. Subsequently, the point cloud was translated. To account for variations in object scale, a maximum translation factor of 5 (times of object extents) was defined. For each spatial direction

(x, y, z), a random value within the range of the negative and positive maximum translation factor was sampled and multiplied by the spatial extent of the point cloud in that axis. This procedure ensures that the magnitude of the applied translations is scale-aware and are meaningful across point clouds of varying sizes.

Conclusion

The PC2Model benchmark dataset offers a controlled, scalable, and cost-effective resource for generating paired point clouds and corresponding 3D models to support the training and evaluation of both conventional and deep learning-based registration methods. By integrating high-fidelity simulated point clouds with real-world scans, the dataset mitigates common limitations of exclusively real-world data, including noise, occlusions, and limited availability. Designed as a standard benchmark, PC2Model enables systematic comparison, validation, and advancement of more robust, accurate and generalizable solutions. The dataset is publicly available at: https://doi.org/10.5281/zenodo.17581812 . As a future work, we plan to benchmark the existing methods on PC2Model dataset. Furthermore, an online evaluation platform will be developed.

Outcomes

- The dataset is publicly accessible via: https://doi.org/10.5281/zenodo.17581812

- A GitHub repository containing the source codes of the python tool developed from this project: https://github.com/saidharb/PC2Model

- A conference paper accepted for oral presentation at ISPRS 2026 congress in Toronto, Canada: PC2Model: ISPRS benchmark on 3D point cloud to model registration Maboudi, Mehdi; Harb, Said; Ferrao, Jackson; Khoshelham, Kourosh; Turkan, Yelda; Mawas, Karam

Acknowledgement

References

Koch, S. et al., 2019. ABC: A Big CAD Model Dataset for Geometric Deep Learning. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 9601-9611. https://doi.org/10.1109/CVPR.2019.00983.

Willis, K. D. et al., 2022. JoinABLe: Learning Bottom-Up Assembly of Parametric CAD Joints. Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 15849-15860. https://arxiv.org/abs/2111.12772.

Wu, Z. et al., 2015. 3D ShapeNets: A Deep Representation for Volumetric Shapes. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1912-1920. https://arxiv.org/abs/1406.5670.

Zhou, Q. & Jacobson, A. et al., 2016. Thingi10K: A Dataset of 10,000 3D-Printing Models. https://doi.org/10.48550/arXiv.1605.04797.

Khoshelham, K. et al. 2017. The ISPRS benchmark on indoor modelling. Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci., XLII-2/W7, 367–372, https://doi.org/10.5194/isprs-archives-XLII-2-W7-367-2017.

Winiwarter, L. et al., 2022. Virtual laser scanning with HELIOS++: A novel take on ray tracing-based simulation of topographic full-waveform 3D laser scanning. Remote Sensing of Environment, 269, 112772. https://doi.org/10.1016/j.rse.2021.112772.