# BIMCaP: BIM-based AI-supported LiDAR-Camera Pose Refinement

M.A. Vega-Torres[1], A. Ribic[1], B. García de Soto[2] & A. Borrmann[1]
[1]Chair of Computational Modeling and Simulation
Technical University of Munich, Munich, Germany
[2]S.M.A.R.T. Construction Research Group, Division of Engineering
New York University Abu Dhabi (NYUAD), Saadiyat Island, United Arab Emirates
miguel.vega@tum.de

**Abstract.** This paper introduces BIMCaP, a novel method to integrate mobile 3D sparse LiDAR data and camera measurements with pre-existing building information models (BIMs), enhancing fast and accurate indoor mapping with affordable sensors. BIMCaP refines sensor poses by leveraging a 3D BIM and employing a bundle adjustment technique to align real-world measurements with the model. Experiments using real-world open-access data show that BIMCaP achieves superior accuracy, reducing translational error by over 4 cm compared to current state-of-the-art methods. This advancement enhances the accuracy and cost-effectiveness of 3D mapping methodologies like SLAM. BIMCaP's improvements benefit various fields, including construction site management and emergency response, by providing up-to-date, aligned digital maps for better decision-making and productivity. Link to the repository: https://github.com/MigVega/BIMCaP.

## 1. Introduction

This research explores the convergence of 3D Building Information Modeling (BIM) with real-world 3D reconstruction, utilizing cost-effective RGB and LiDAR sensors. Traditional 3D data acquisition methods, such as terrestrial laser scanning, present challenges in terms of high costs and time-intensive procedures, particularly in the context of construction site monitoring or disaster relief. The practical implementation of faster 3D data acquisition methods has been made possible by the advancement of simultaneous localization and mapping (SLAM). However, state-of-the-art algorithms still encounter challenges to accurately map complex and dynamic environments, such as construction sites.

BIM has become a revolutionary technology within the Architecture, Engineering, and Construction (AEC) domain, offering comprehensive geometric and semantic information throughout a building's life cycle. In this context, BIMs provide a valuable foundation for rectifying data acquired through SLAM algorithms in real time using low-cost sensors. The automatic alignment of RGB and LiDAR data with the BIM holds significant potential to rapidly create precise 3D maps in GPS-denied environments (such as indoors). This alignment not only facilitates safety monitoring and quality management but also contributes to the quick development of a digital twin, providing an accurate 3D representation of actual asset states.

In this research, we address several critical questions central to advancing state-of-the-art technologies in the field of sensor pose correction for accurate 3D reconstruction. By utilizing the geometric and semantic information inherent in a BIM, we examine the potential of a bundle adjustment module to refine drifted sensor poses. Additionally, we explore methodologies for effectively combining calibrated sparse LiDAR data with RGB images to produce detailed depth maps, aiming for a comprehensive reconstruction. Furthermore, we evaluate semantic segmentation algorithms in complex indoor construction settings and develop a strategy for improving their performance. We demonstrated the improvement in performance through extensive experiments on the publicly available ConSLAM dataset (Trzeciak et al. 2023). In this research, we aim to contribute to the development of robust and efficient techniques for 3D

reconstruction, which has implications for a range of applications, including construction site management, emergency response, and beyond.

As we delve into the intricacies of our methodology, it is essential to acknowledge the scope and assumptions of our method. Since our goal is to have a robust method using sensors with a reduced field of view (FoV), such as solid-state LiDARs or RGB-D cameras, we only use the LiDAR information in the FoV of the camera, ignoring the rest of the available points. In our approach, we also assume that there is an initial rough alignment of the drifted trajectory with the reference map.

This paper is structured as follows: Section 2 offers an in-depth exploration of related research efforts. Section 3 delineates the proposed framework split into three main steps. Subsequently, Section 4 presents the findings derived from numerous experiments conducted on real-world construction site data. Finally, Section 5 provides the culmination of this work, offering conclusive insights and avenues for future research.

## 2. Related Work

Several studies have approached the alignment of RGB images with BIMs in two main ways: (1) as a global localization problem and (2) as a pose-tracking problem.

In the global localization problem, Acharya et al. (2022) introduced BIM-PoseNet, utilizing synthetic images from a 3D indoor model to achieve a 2-meter accurate camera pose without an initial position. Haque et al. (2020) localized a unmanned aerial vehicle in the BIM coordinate system by detecting doors and windows in RGB images, using You Only Look Once (YOLO) for object detection and ORB-SLAM2 for 3D mapping.

In the pose-tracking approach, Kropp et al. (2018) focused on image-to-4D BIM registration using line segments as features, with manual intervention for initial registration. Boniardi et al. (2019) proposed a clutter-handling method using a convolutional neural network for layout prediction and a particle filter algorithm for pose tracking using a floor plan as a reference map. The method proposed by Dantas et al. (2022) aims at quickly correcting camera poses using vanishing points, lines, and synthetic renders created from a BIM.

Other methods addressed the challenge of creating a coherent 3D map of the environment aligned with a given reference map. Vega Torres et al. (2023) used a BIM to align and correct 360-degree LiDAR measurements, which initial poses were calculated with a LiDAR-based SLAM algorithm. Sokolova et al. (2022) presented the Floorplan-Aware Camera Poses Refinement (FACaP) method, aligning Visual-SLAM maps with floor plans using semantic segmentation and an optimization model considering geometric, floor-to-plane and wall-to-floorplan terms for map correction.

However, most of these methods were tested in indoor residential apartments without the level of clutter, dynamic elements, and changing lighting conditions present in real-world construction sites. Furthermore, the literature review shows accuracy metrics for diverse methods; however, all were evaluated on separate, unrelated case studies. This fact highlights the necessity for a standardized dataset (i.e., BIM and synchronized sensor information) tailored explicitly to real-world construction site environments, enabling fair and consistent ranking.

## 3. Methodology

We propose a framework designed to align a sequence of synchronized LiDAR scans and RGB images with a 3D BIM, thereby refining the initial approximated camera poses, which inherently suffer from drift owing to the characteristics of SLAM algorithms. Our framework can be divided

into three significant steps. **Step 1.** The initial step of our methodology involves fusing camera images and sparse LiDAR scans in precise depth maps. This process is facilitated through a hybrid approach employing interpolation and a deep learning (DL) technique, which then allows the projection of the pixel information (such as semantic information) into the 3D space. **Step 2.** Subsequently, in the second step, semantic segmentation is applied to the images, enabling the detection of permanent elements such as walls, columns, and floors within the reconstructed 3D map. Simultaneously, a point cloud and a vectorized floor plan with semantic information are created from the BIM. This vectorized semantic floor plan will be used as a reference map for the alignment of the real-world data. **Step 3.** In the third step, we employ a statistical approach to generate initial synthetic camera poses. These poses are then refined through a bundle adjustment (BA) module, which integrates custom cost functions. These functions are designed to iteratively enhance the accuracy of sensor poses, thereby ensuring optimal alignment between the generated map and the semantically vectorized floor plan from the BIM. This refinement process selectively considers only permanent elements, which are identified through semantic segmentation in real-world images and projected into three-dimensional space using the previously estimated depth maps. Fig. 1 illustrates the proposed semantic-aware pose optimization framework.

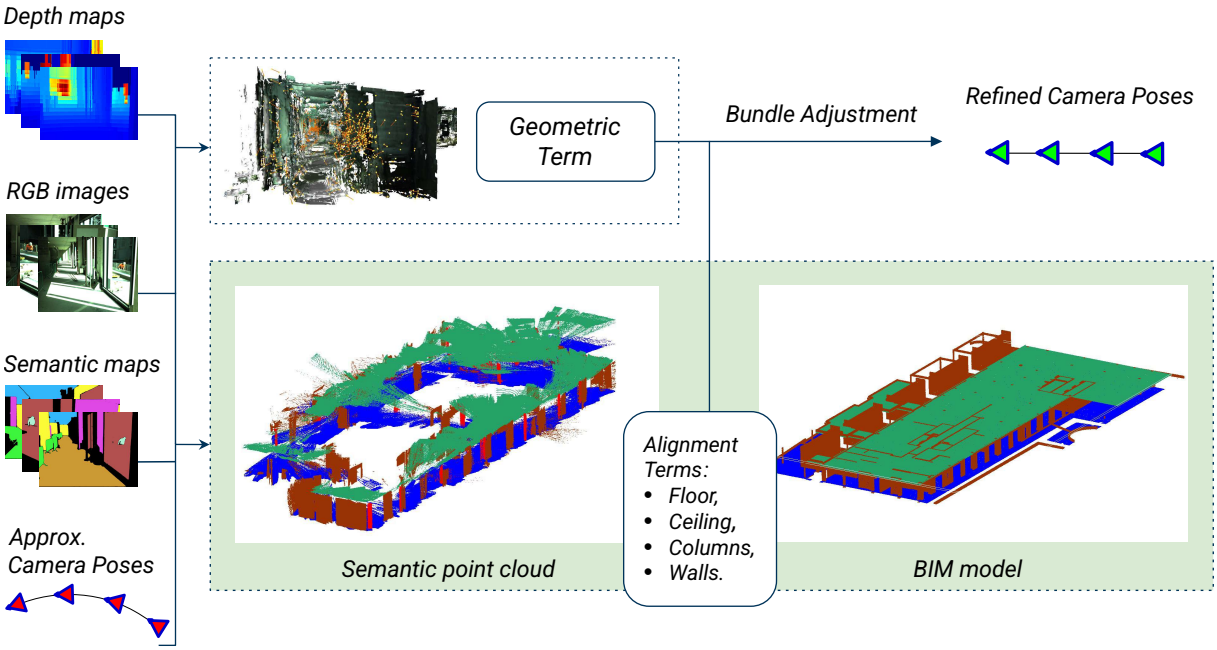



Figure 1: Overview of the proposed BIMCaP framework for sensor pose refinement. The depth maps are used to project the semantic maps (created from the images) into the 3D space using the approximated initial poses (drifted due to SLAM). Different terms aim to correlate the data measured over permanent elements (i.e., reliable landmarks) with the BIM. Moreover, a geometric term ensures geometric consistency among real-world images.

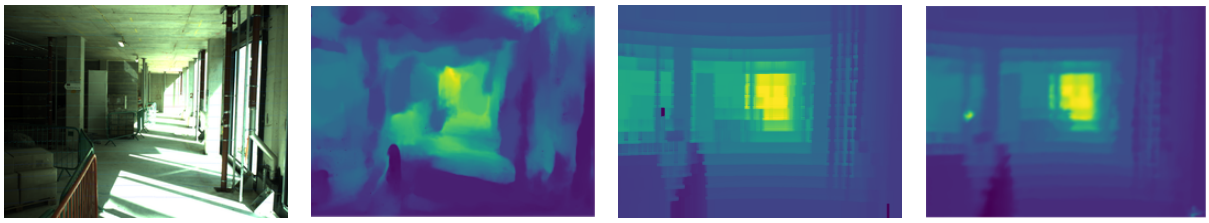
### 3.1. Step 1: LiDAR and camera fusion

To fuse the information from the LiDAR and the camera, we first project the visible point cloud (in the FoV of the camera) into the image, and then we aim to generate a dense map that is coherent with the image and LiDAR information. The projection of the LiDAR points to the camera image is made with the intrinsic and extrinsic parameters of the camera and with the package provided by Trzeciak et al. (2023); this package ensures the camera image is undistorted,

and only the corresponding (timestamped synchronized) LiDAR points of the small FoV of the camera are projected from the 3D space to the 2D image. Upon this step, we now have depth information for several pixels of the image. This depth is, however, very sparse since we are working with a 360° LiDAR. It is essential to mention that to ensure that the method works appropriately with sensors with a reduced FoV (such as solid-state LiDARs or RGB-D cameras), we only use the LiDAR information in the FoV of the camera.

The sparsity of the point cloud would not be sufficient to leverage all the information from the image in the 3D space; therefore, we subsequently aim to create a dense depth map using the point cloud and the corresponding camera image.

Currently, numerous DL methods serve for depth estimation, yet many of them are optimized for outdoor environments (such as the KITTI dataset). Therefore, their accuracy tends to decline in indoor settings, which constitutes the focus of our investigation. Following extensive experimentation with various methodologies, we chose to adopt a hybrid method that combines linear interpolation with CompletionFormer (CF) (Zhang et al. 2023). Fig. 2 illustrates the results of this hybrid approach, showing the original CF output alongside the refined outcome involving an initial linear interpolation.



(a) (b) (c) (d)

Figure 2: Depth completion with sparse LiDAR point cloud: (a) original image from the ConSLAM dataset with original sparse projected LiDAR scan; (b) depth map using only CompletionFormer; (c) depth map using only linear interpolation and (d) using linear interpolation and CompletionFormer. It is evident that (d) yields the best results since it is smother than (c) and more coherent with the measurements than (b).

### 3.2. Step 2: Semantically enriched maps

In this step, we aim to create maps that will allow the sensor pose correction in the subsequent step. This step is divided into two sub-steps: Firstly, we create a reference semantic vectorized floor plan from the BIM, and secondly, we enrich the 3D map created with real-world data with semantic information. This semantic enrichment serves a pivotal role in distinguishing permanent elements within real-world data, such as walls, columns, and floors, which can be reliably aligned with the BIM.

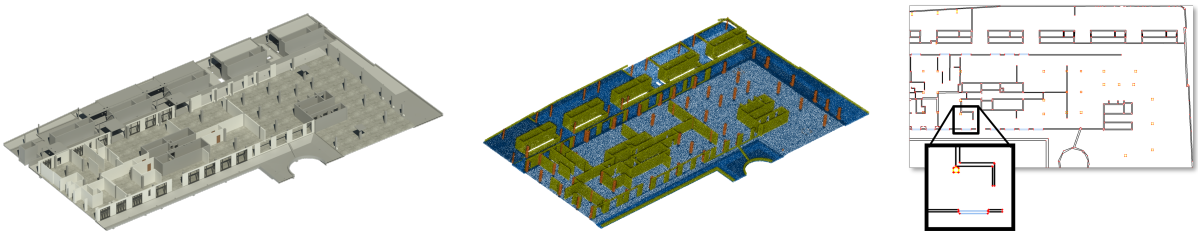
#### 3.2.1. Reference map

To prepare for implementing the pose correction module, we simplify the 3D BIM into a 2D semantic vectorized floor plan. Since walls and columns are perpendicular to the XY plane, this reduction not only retains all vertical structural element information but also allows efficient pose optimization in subsequent stages.

To generate the 2D semantic vectorized floor plan, the BIM undergoes conversion from Industry Foundation Classes (IFC) format to OBJ format using ifcConvert. Following this,

distinct OBJ files are generated for each entity within the model (e.g., walls, columns, floor, ceiling, windows, and doors). Then, uniform point cloud sampling is applied to each OBJ file, and the resulting semantically enriched synthetic point clouds are merged into a single one. An illustration of such a point cloud can be observed in Fig. 3b.

The created synthetic 3D point cloud is projected vertically into 2D images within a specified height range, typically within ± 20 cm from the floor level. Semantic labels are utilized to filter each element in the point cloud. Subsequently, image processing methods such as contour and line detection are employed to identify line segments representing individual elements in the 2D projection. These detected lines are then consolidated, including their start and end points, to form the vectorized semantic floor plan. A resulting floor plan is depicted in Fig. 3c.



(a) (b) (c)

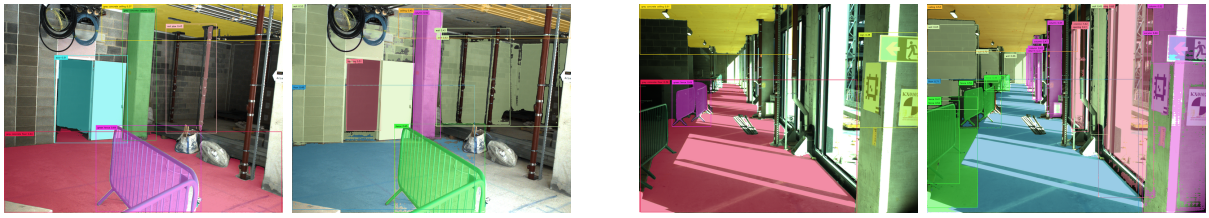
Figure 3: Reference map preparation: (a) original 3D BIM (without ceiling); (b) uniformly sampled 3D point cloud with semantic information from the BIM; and (c) vectorized semantic floor plan, from which the walls and columns (in black and yellow) are used for pose refinement in the subsequent pose optimization step.

### 3.2.2. Semantic segmentation of real-world data

To filter permanent elements that we can match from the real-world data with the BIM, we leverage state-of-the-art image semantic segmentation algorithms. More specifically, we use a modified version of Grounding DINO (Liu et al. 2023). However, for the object detection task, we replace the DINO algorithm with a tiny version of the RTMDet algorithm (Lyu et al. 2022) pre-trained with the COCO dataset and 250 labeled images of the ConSLAM dataset, which contains custom classes typical of a construction site. These images were labeled semi-automatically using the Computer Vision Annotation Tool (CVAT).Thus, our approach enables the detection of objects of interest, expanding beyond the foreground elements identified by the original Grounding DINO version. Fig. 4 illustrates the results of the semantic enrichment before and after the proposed enhancement, and Fig. 5a shows the top view of the resulting semantically enriched 3D point cloud after projecting the semantic labels to the 3D space with the previously generated depth maps. In this last figure, it is also visible that we can now filter walls, columns, floor, and ceiling points in the depth maps, which can reliably be used for registration with the BIM and, therefore, for camera pose optimization. It is worth mentioning that the floor and ceiling predicted labels were also used to optimize the depth maps, creating smoother surfaces in these regions with blurring operations in the 2D depth maps.

## 3.3. Step 3: Sensor pose calculation and refinement

The initial approximations of sensor poses are ideally determined using a Visual-SLAM framework. However, our experimentation with cutting-edge SLAM algorithms such as DROID-SLAM or Go-SLAM yielded unsatisfactory results when applied to the ConSLAM dataset,



(a) (b) (c) (d)

Figure 4: Semantic segmentation over 2D images of the ConSLAM dataset: (a) inference with the original Grounding DINO algorithm (b) inference result after replacing DINO with pre-trained RTMDet for object detection. (b) comprehends predicted labels for the walls in the background, which are critical for our camera pose refinement framework. Similarly, (c) and (d) are, respectively, the results of Grounding DINO and the results of our proposed pipeline.

functioning correctly only for limited segments of the trajectory. Despite these limitations, advancements in odometry systems suggest that addressing this challenge will become feasible in the future. Therefore, and since we aim to refine slightly drifted poses and experiment under different magnitudes of drift, we opted to create a synthetic trajectory instead, simulating the output of a SLAM framework. This process is explained in the following subsection. Subsequently, we introduce the method that is used to improve the accuracy of these poses.

### 3.3.1. Synthetic pose calculation

To make sure our approach matches the usual trajectory patterns seen in existing SLAM systems and to have the flexibility to study how stable our method is when it comes to convergence with different starting positions, we carefully engineer synthetic trajectories. When creating these trajectories, we focus on replicating the gradual drifting feature of SLAM-generated trajectories. In other words, we want the error at each pose to slowly increase over time.

Therefore, we model the translation offset from the original sensor pose $\Delta T_{i+1}$ as a normally distributed random variable with mean $\Delta t_i$ and variance $\sigma_t^2$. Formally, $\Delta T_{i+1} \sim \mathcal{N}\left(\Delta t_i, \sigma_t^2\right)$ with $\Delta T_1 \sim \mathcal{N}\left(0, \sigma_t^2\right)$ where $\Delta t_i$ is the previously sampled offset value, and the variance $\sigma_t^2$ is an adjustable hyper-parameter which would determine the offset of the trajectory from the ground truth poses. Regarding the camera rotation, we randomly sample degree offsets $\Delta\phi \sim \mathcal{N}\left(0, \sigma_p^2\right), \Delta\theta \sim \mathcal{N}\left(0, \sigma_{th}^2\right)$ around pitch and yaw directions. Fig. 5b presents the resulting map using a synthetically drifted trajectory.

### 3.3.2. Pose optimization

Inspired by the FaCAP framework (Sokolova et al. 2022), we consider several terms in our cost function to achieve sensor pose refinement with the BIM using BA. To create a consistent 3D map from the real-world sequential images, we use a geometric term that encapsulates the divergence between 3D point estimations from two distinct viewpoints. In other words, the geometric term considers photogrammetric constraints, and in our case, we use COLMAP to obtain features and correspondences among sequential images. Fig. 5b and 5c visualize some of these features.

Furthermore, we use a floor term designed to ensure that segmented points corresponding to the floor lie within a single plane and close to the floor surface defined in the BIM. In a parallel manner, we introduce a ceiling term that incorporates information from the model's ceiling

for optimization purposes. In addition, we include wall and column terms. These elements are crucial for correcting rotations around the vertical axis (i.e., yaw variations) and horizontal translations.

(a) (b) (c)

Figure 5: Features used for optimization (a) Top view semantic segmented map generated with the ground truth poses and the segmentation results of walls (in red) and floor (in blue) as explained in Section 3.2.2; (b) map created with synthetic poses of Exp. 1 (obtained as explained in Section 3.3.1), here the COLMAP features are visible; (c) view from an indoor observer's perspective of the point cloud with highlighted Scale-Invariant Feature Transform (SIFT) features used in the geometric term for optimization.

## 4. Experiments and results

### 4.1. Dataset and evaluation details

To ensure reproducibility and enable benchmarking, we tested the developed method on the ConSLAM dataset (Trzeciak et al. 2023). ConSLAM represents a pioneering effort, offering the first open-access dataset acquired in an indoor cluttered construction site. This dataset encompasses sequences of RGB and LiDAR data together with Terrestrial Laser Scanning (TLS) point clouds. The latter was leveraged as a resource for the generation of a BIM with centimeter-level accuracy. The GT poses of ConSLAM were calculated using SLAM2REF, an enhanced version of BIM-SLAM (Vega Torres et al. 2023) and OGM2PGBM (Vega Torres et al. 2022) for large-scale maps, which is robust to LiDAR motion distortion and Scan-Map deviations.

To quantify the quality of the whole trajectory before and after pose optimization, we used the standardized root mean square error (RMSE) of the absolute trajectory error (ATE) in position (also referred to as translation) and in rotation. Moreover, for better comparison, we incorporated some of the metrics introduced in (Sokolova et al. 2022) including the Map Mean Entropy (MME), Mean Plane Variance (MPV), and the Nearest Neighbor Distance (NND). The MME serves to assess the quality of 3D maps, with a higher MME signifying favorable alignment between the input cloud and the reference map. The MPV evaluates the variance among planes within the map, with lower MPV values indicating more uniform and well-defined surfaces. The NND quantifies the average distance between adjacent points in the point cloud, with smaller NND values indicating denser point clouds. The MME value of 0.761, calculated using the GT poses (as shown in 1), represents the optimal alignment between the real-world point cloud and the BIM model. For understanding, this value would be zero if no deviations between the actual environment and the model (Scan-BIM deviations) exist.

Table 1: Comparison of validation measurements for Exp. 1 using the different methods. All values are in meters except for the rotational ATE, which is given in degrees. The best overall results are highlighted in bold, while the best results per method are underlined. G, F, W, Co, and Ce stand for the geometric, floor, wall, column, and ceiling terms, respectively.

| **Source/Method** | **G** | **F** | **W** | **Co** | **Ce** | **MME↓** | **MPV↓** | **NND↓** | **$ATE_{pos}$↓** | **$ATE_{rot}$↓** |
|---|---|---|---|---|---|---|---|---|---|---|
| GT poses | - | - | - | - | - | 0.761 | 0.040 | 0 | 0 | 0 |
| Exp. 1 | - | - | - | - | - | 1.027 | 0.059 | 0.557 | 1.391 | 9.99 |
| FACaP | ✓ | ✓ | ✓ | - | - | 0.979 | 0.054 | 0.503 | 1.321 | 15.40 |
| | ✓ | - | - | - | - | 1.013 | 0.058 | 0.566 | 1.385 | **8.84** |
| | - | ✓ | - | - | - | 0.966 | 0.053 | 0.503 | 1.358 | 16.50 |
| | - | - | ✓ | - | - | 1.031 | 0.059 | 0.545 | 1.378 | 9.82 |
| BIMCaP | ✓ | ✓ | ✓ | ✓ | ✓ | **0.956** | **0.052** | 0.460 | **1.281** | 11.84 |
| | ✓ | ✓ | ✓ | - | ✓ | 0.959 | 0.052 | **0.456** | **1.281** | 11.81 |
| | ✓ | ✓ | ✓ | - | - | 0.975 | 0.053 | 0.505 | 1.311 | 12.58 |
| | - | ✓ | - | - | - | 0.966 | 0.054 | 0.519 | 1.351 | 13.63 |
| | - | - | ✓ | ✓ | - | 1.034 | 0.059 | 0.549 | 1.378 | 9.75 |
| | - | - | - | - | ✓ | 0.982 | 0.054 | 0.480 | 1.387 | 12.71 |

### 4.2. Pose refinement results

The results of our framework are compared against the state-of-the-art FaCAP pipeline (Sokolova et al. 2022) and evaluated meticulously with three different experiments. The first experiment consists of a synthetic trajectory that has an offset of around 1.4 meters in translation and 10 degrees in rotation (Exp. 1), the second one has an offset of only 30.3 cm in translation and 8.82 deg in rotation (Exp. 2), and the third one only has rotation offset of 9.6 degrees (Exp. 3).

Table 1 shows initial metrics based on ground truth and synthetic poses for Exp. 1, along with results after pose optimization using various terms of the FACaP pipeline and the proposed BIMCaP framework.

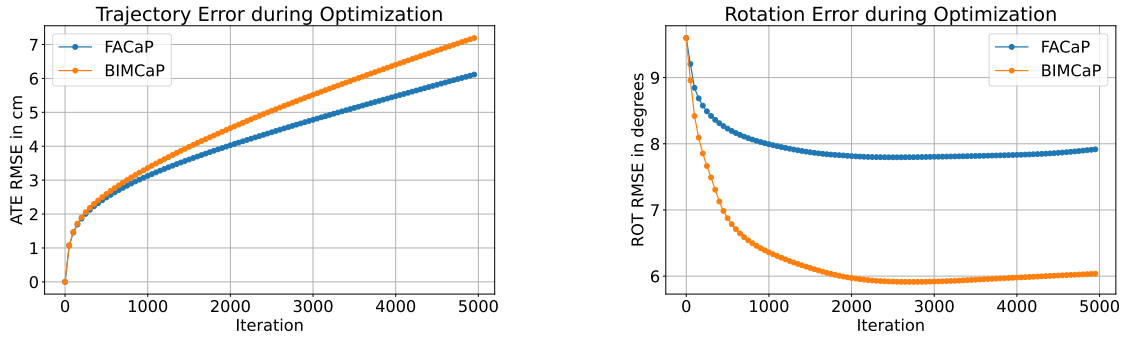
The findings from Exp. 1 (Tab. 1) emphasize the efficacy of utilizing all terms for optimizing translational errors. However, this approach may not consistently yield optimal results when addressing rotational errors. Notably, while BIMCaP demonstrates a superior reduction in translational error by 4 cm compared to FACaP, both methodologies become trapped in a local minimum, impeding the accurate optimization of the poses. This issue can be attributed to the substantial difference between the synthetic poses and the ground truth.

Exp. 2 and 3 results (Table 2 and Fig. 6) indicate superior performance of both methods in optimizing rotational errors over translational errors. Notably, BIMCaP significantly enhances yaw and pitch angles during the pose optimization process. Fig. 7 illustrates how BIMCaP aligns the floor and ceiling points to the correct planes, contrary to FACaP, which tries to fit a plane among the given measurements without any reference.

Exp. 3 exposes a limitation in our approach, as optimizing trajectory with only rotational offsets resulted in unintended translations. This could be due to the simultaneous optimization of both translation and rotation, causing discrepancies. Additionally, the challenge of accurately calculating sensor poses is intensified by the reduced FoV and the sparse ground truth poses.

Table 2: Pose optimization results for Exp. 2 and 3: given a small translational and rotational offset. In addition to the $ATE_{pos}$ and $ATE_{rot}$, we provide the RMSE for the Yaw, Pitch, and Roll axes separately in degrees.

| **Source/Method** | **$ATE_{pos}$ (cm)↓** | **$ATE_{rot}$ (deg)↓** | **Yaw↓** | **Pitch↓** | **Roll↓** |
|---|---|---|---|---|---|
| Exp. 2 before optim. | 30.3 | 8.82 | 4.31 | 4.31 | 0.29 |
| FACaP (Sokolova et al. 2022) | **30.2** | 7.70 | 3.79 | 3.79 | **0.21** |
| BIMCaP | 30.4 | **5.61** | **2.73** | **2.75** | 0.23 |
| Exp. 3 before optim. | 0 | 9.60 | 4.70 | 4.72 | 0.29 |
| FACaP (Sokolova et al. 2022) | **6.2** | 7.92 | 3.91 | 3.90 | **0.19** |
| BIMCaP | 7.2 | **6.04** | **2.96** | **2.96** | 0.22 |




(a) (b)

Figure 6: Development of the translational (a) and rotational error (b) given only a rotational offset as described for Exp. 3.

## 5. Conclusion and future directions

We have demonstrated that BIMCaP enables alignment and correction of a sequence of the camera and reduced FoV LiDAR measurements with a BIM; the technique takes into consideration only reliable selected semantic landmarks (in our case, floor, walls, columns, and ceiling) for the drift correction.Moreover, we evaluated our technique in the open-access ConSLAM dataset and compared it against a state-of-the-art method, ensuring reproducibility and benchmarking. In future work, we aim to improve the accuracy of the optimization process and add global registration and change detection capabilities to our framework.

(a) (b)

(c) (d)

Figure 7: Side views of the different maps. (a) ground truth map; (b) map created with synthetic poses of Exp. 1; (c) map after FACaP optimization and (d) after BIMCaP optimization. The BIMCaP result shows better alignment with the real floor and ceiling planes.

## 6. Acknowledgement

This research is part of the INTREPID project funded by EU's Horizon 2020 program (Grant agreement ID: 883345) and the Research Unit 5672 funded by the German Research Foundation (DFG) (Grant ID: 517965147). This work has also benefited from the collaboration with the NYUAD Center for Interacting Urban Networks (CITIES), funded by Tamkeen under the NYUAD Research Institute Award CG001. Shaowen Qi's contributions to data labeling and to the semantic segmentation step are also acknowledged.

## References

Acharya, D., Tennakoon, R., Muthu, S., Khoshelham, K., Hoseinnezhad, R. & Bab-Hadiashar, A. (2022), 'Single-image localisation using 3D models: Combining hierarchical edge maps and semantic segmentation for domain adaptation', *Automation in Construction* **136**, 104152.

Boniardi, F., Valada, A., Mohan, R., Caselitz, T. & Burgard, W. (2019), 'Robot localization in floor plans using a room layout edge extraction network'.

Dantas, R., Peter, S., Wang, X., Vega, M. & Dugstad, A. (2022), Towards real-time image localization with bim models, *in* 'Proceedings of 33. Forum Bauinformatik'.

Haque, A., Elsaharti, A., Elderini, T., Elsaharty, M. A. & Neubert, J. (2020), 'UAV autonomous localization using macro-features matching with a CAD model', *Sensors (Basel, Switzerland)* **20**(3).

Kropp, C., Koch, C. & König, M. (2018), 'Interior construction state recognition with 4d BIM registered image sequences', *Automation in Construction* **86**, 11–32.

Liu, S., Zeng, Z., Ren, T., Li, F., Zhang, H., Yang, J., Li, C., Yang, J., Su, H., Zhu, J. et al. (2023), 'Grounding dino: Marrying dino with grounded pre-training for open-set object detection', *arXiv preprint arXiv:2303.05499* .

Lyu, C., Zhang, W., Huang, H., Zhou, Y., Wang, Y., Liu, Y., Zhang, S. & Chen, K. (2022), 'RTMDet: An empirical study of designing real-time object detectors', *arXiv preprint arXiv:2212.07784* .

Sokolova, A., Nikitin, F., Vorontsova, A. & Konushin, A. (2022), Floorplan-aware camera poses refinement, *in* '2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)', pp. 4857–4864.

Trzeciak, M., Pluta, K., Fathy, Y., Alcalde, L., Chee, S., Bromley, A., Brilakis, I. & Alliez, P. (2023), 'Conslam: Construction data set for SLAM', *Journal of Computing in Civil Engineering* **37**(3), 04023009.

Vega Torres, M., Braun, A. & Borrmann, A. (2022), Occupancy grid map to pose graph-based map: Robust BIM-based 2D-lidar localization for lifelong indoor navigation in changing and dynamic environments, *in* S. F. S. Eilif Hjelseth & R. Scherer, eds, 'eWork and eBusiness in Architecture, Engineering and Construction: ECPPM 2022', CRC Press, Trondheim, Norway, pp. 265–289.

Vega Torres, M., Braun, A. & Borrmann, A. (2023), BIM-SLAM: Integrating BIM models in multi-session SLAM for lifelong mapping using 3D LiDAR, *in* 'Proceedings of the 40th International Symposium on Automation and Robotics in Construction (ISARC 2023)', International Association for Automation and Robotics in Construction (IAARC), Chennai, India.

Zhang, Y., Guo, X., Poggi, M., Zhu, Z., Huang, G. & Mattoccia, S. (2023), Completionformer: Depth completion with convolutions and vision transformers, *in* 'Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition', pp. 18527–18536.