# Learning-based 6 DOF Camera Pose Estimation Using BIM-generated Virtual Scene for Facility Management

**Thai-Hoa Le[1], Ju-Chi Chang[1], Wei-Yi Hsu[1], Tzu-Yang Lin[2], Ting-wei Chang[2] and Jacob J. Lin[1]**

[1]Department of Civil Engineering, National Taiwan University, Taiwan
[2]Lab for Service Robot Systems, Delta Research Center, Taiwan
D10521027@ntu.edu.tw, yuiochang@caece.net, R12521610@ntu.edu.tw,
TZUYANG.LIN@deltaww.com, TINGWEI.CHANG@deltaww.com, jacoblin@ntu.edu.tw

**Abstract –**
**Image-based indoor localization is a promising approach to enhancing facility management efficiency. However, ensuring localization accuracy and improving data accessibility remain key challenges. Therefore, this research aims to automatedly localize images captured during facility inspections by matching the viewpoint of the camera with a corresponding viewpoint in a Building Information Modeling-based (BIM) simulated environment. In this paper, we present a framework that generates photorealistic synthetic images and trains a deep learning model for camera pose estimation. Synthetic datasets are generated in a simulation environment, allowing precise control over scene parameters, camera positions, and lighting conditions. This allows the creation of diverse and realistic training data tailored to specific facility environments. The deep learning model takes RGB images, semantic segmented maps, and corresponding camera poses as inputs to predict six-degree-of-freedom (6DOF) camera poses, including position and orientation. Experimental results demonstrate that the proposed approach can enable indoor image localization with an average translation error of 5.8 meters and a rotation error of 69.05 degrees.**

**Keywords –**
**Synthetic data; Image localization; Pose estimation; Facility management; BIM**

## 1 Introduction

In facility management, accurate camera pose estimation is crucial for tasks such as building surveying, indoor navigation, and asset tracking. Within indoor built environments, reliable positioning data is the foundation for numerous location-based services. By determining the precise orientation and location of the camera, facility managers can automate the detection of structural changes, schedule efficient maintenance, and enhance occupant safety. Additionally, camera pose estimation enables advanced applications like augmented reality guidance and remote inspections, ultimately improving the overall efficiency and effectiveness of facility operations.

A traditional approach to getting a camera pose from a given 3D model of the scene is to match 2D image features with their corresponding 3D points using an n-point pose solver in a RANSAC loop [1]. Another common method is reconstructing a 3D scene via Structure-from-Motion (SfM), in which 3D points are triangulated from multiple local features and can be associated with the corresponding image descriptors [2, 3]. In recent years, the integration of SfM with Multi-View Stereo (MVS) has transformed 3D reconstruction by making data collection and processing more broadly accessible [4]. By capturing and computing multiple overlapping images, it is possible to derive 3D information without relying on knowing camera poses, camera calibration, or surveyed reference markers in the scene. However, in construction, capturing ordered photo collection typically follows a strict schedule and is implemented at predetermined intervals. Furthermore, reconstructing a reality model is time-consuming, and its quality may decrease if random photos are taken at different times. On the other hand, deep learning approaches have demonstrated strong capabilities in providing reliable and real-time image localization [5, 6]. Deep learning-based methods can automatically learn discriminative representations that are robust to large variations in scene appearance and viewpoint changes. Their ability to handle complex environments, varying lighting conditions, and scene dynamics leads to improved accuracy and efficient inference speeds,

making them suitable for real-time deployment [7].

Nowadays, Building Information Modeling (BIM) model is commonly available in the project, and it serves as a rich source for generating synthetic imagery of the scene. By rendering the model from multiple viewpoints, synthetic data can be produced to augment or replace labor intensive image capture. Such synthetic datasets are particularly beneficial for training deep learning–based (e.g., CNN-based) localization models, as they provide diverse examples of the scene without requiring extensive onsite photo documentation. Especially this approach can work with a low level-of-detail BIM model [8], thereby reducing the required time for the indoor camera pose regression process.

In this paper, we present a deep learning-based image localization approach that is suitable for dynamic and complex construction indoor environments. The framework includes two main phases: synthetic data generation and image localization.

## 2 Related Work

### 2.1 Traditional Approaches for Camera Pose Estimate

The goal of camera pose estimation is to estimate 6 degrees of freedom (DoF) of a given camera, including 3D translations and 3D orientations. Classical methods for camera pose estimation primarily rely on geometric principles and feature-based matching. Techniques such as Perspective-n-Point (PnP) solvers, including Direct Linear Transform (DLT) [9], EPnP [10], and advanced variants like UPnP and OPnP [11, 12], estimate camera pose from 2D-3D correspondences with improvements in efficiency and robustness. Feature-based approaches, like Structure-from-Motion (SfM), reconstruct 3D scenes by detecting and matching local features (e.g., SIFT, SURF), performing triangulation, and refining camera pose and 3D structures through bundle adjustment [3]. Multi-view geometry techniques, such as essential or fundamental matrix estimation, recover relative camera motion between image pairs. Additionally, model-based methods align 2D image features with known 3D models, while marker-based systems, using fiducial markers like ARTag or AprilTag provide reference points for simplified pose estimation [13, 14]. These methods then form the theoretical foundation for modern deep learning solutions in camera pose estimation.

### 2.2 Deep Learning-based Image Localization

Compared to traditional methods, which rely heavily on geometric principles, recent advancements in deep learning have transformed the field, offering robust and scalable solutions. A comprehensive overview of state-of-the-art methods for deep learning-based image localization, particularly in indoor environments, is presented in the work of [15], showing numerous benefits of using convolutional neural network (CNN) models for camera pose estimation. CNNs trained on large datasets have demonstrated exceptional capability in learning feature representations. Pose regression can be performed using RGB-D images, which incorporate depth information to encode the 3D location of each pixel within the scene relative to a known environment. Pair-wise registration is a common method in which corresponding 3D key points between consecutive frames are utilized to estimate the relative camera pose [16]. However, depth cameras are not as popular in construction as ordinary cameras, leading to limitations for using RGB-D photos. Deep learning-based approaches using RGB images only, such as PoseNet and its variants, have become prominent in the construction domain. PoseNet directly regresses the camera's 6-DOF (degrees of freedom) pose from an input image, bypassing the need for explicit feature extraction and geometric matching [17]. Variants of PoseNet further enhance accuracy and robustness by introducing loss functions tailored to pose regression, attention mechanisms, or incorporating spatial constraints [8, 18, 19]. Data augmentation is also used to increase fine-tuning samples [20, 21]. Deep learning-based localization methods not only improve the robustness and scalability of image-based localization systems but also adapt well to diverse application domains. By leveraging neural networks' ability to learn feature representations directly from data, these approaches overcome many limitations of traditional methods, such as sensitivity to occlusion, lighting variations, and large-scale environments.

### 2.3 Synthetic Data for Model Training

Recent advances in synthetic data generation have significantly impacted various computer vision applications, including image localization and camera pose estimation. Synthetic datasets, created through graphics engines or procedural modeling, allow researchers to overcome the challenges posed by limited real-world data, expensive annotation processes, and environmental variability. In construction, synthetic data generated from BIM models can be useful for training CNN models for different tasks, including image localization. Acharya et al. [8] introduced a pose estimation approach to determine the 6-DOF camera pose, which can serve as an initial location input for techniques like PDR, SLAM, and visual odometry. A deep Bayesian recurrent CNN is fine-tuned with synthetic image sequences generated from a BIM model to predict the poses of real image sequences [22]. Hong et al. [23] proposed an image retrieval method for

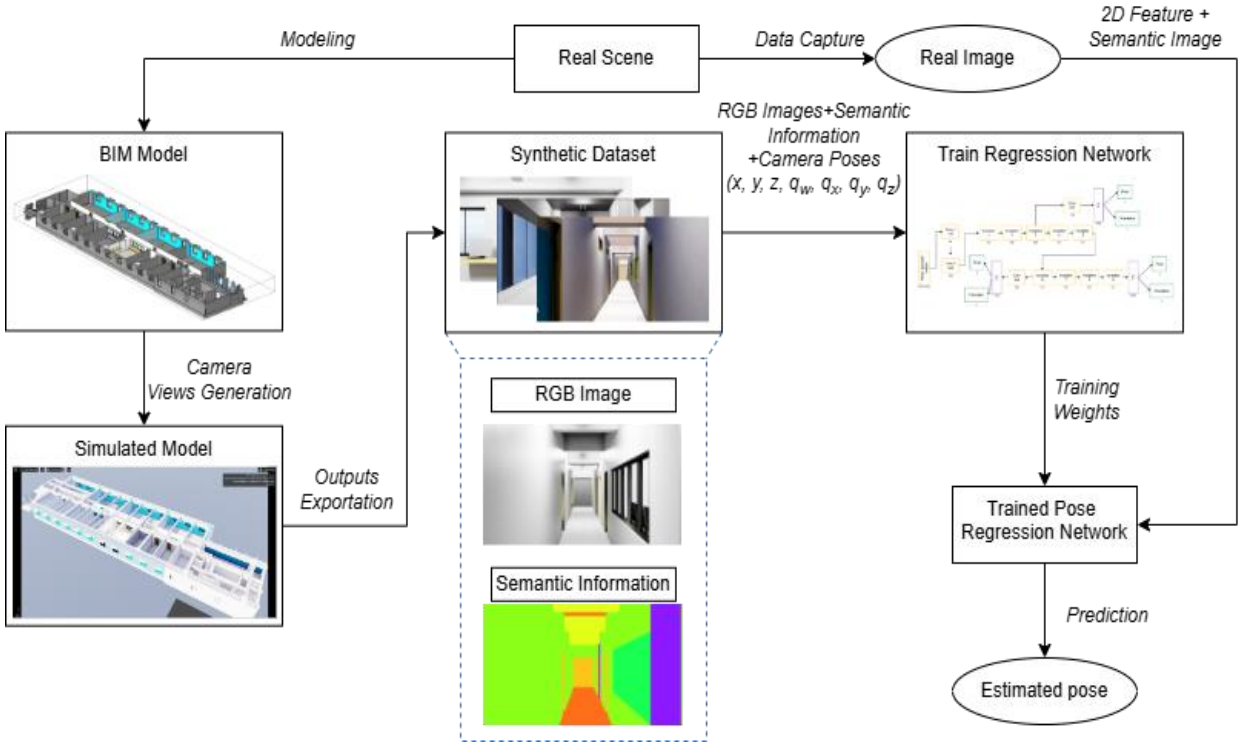


Figure 1. Overview of the framework for CNN-based image localization in this study

achieving coarse indoor localization utilizing BIM. It can be said that synthetic data has emerged as a flexible source for supplementing building image datasets and enhancing the performance and reliability of learning-based visual localization frameworks. The effectiveness of deep learning models trained on synthetic images may be influenced by the reality gap, which refers to discrepancies in color and texture between synthetic and real-world images [24].

In our framework, we combine different platforms to render photorealistic images with similar appearances to real-world images. The rendered outputs are used to train a PoseNet-based CNN model for image localization. The contributions of this study include: (1) propose an automated BIM-based localization for indoor inspection images, and (2) apply cross-domain feature integration to train a deep learning model to improve the indoor image localization.

## 3 Methodology

### 3.1 Framework of the CNN-based image localization

Figure 1 indicates our framework for CNN-based image localization, which combines synthetic data generation and deep learning to achieve accurate and robust image localization in dynamic construction scenarios. The BIM model is the basis of real-world environment simulations, including geometric, material, and spatial information, which are necessary to generate a synthetic dataset comprising images rendered from multiple viewpoints under diverse conditions. Furthermore, BIM software provides essential functionalities, such as its API, which allows automated creation and manipulation of camera poses within the model environment. These capabilities enable the efficient generation of diverse camera views, facilitating the development and testing of camera pose estimation techniques. The BIM model is then exported to an advanced simulation platform to generate the required synthetic dataset. This dataset is used to train a deep learning-based pose regression network, which learns to estimate the camera pose directly from input images. Real images captured from the construction scene are fed into the trained network, producing pose predictions.

### 3.2 Synthetic Data Generation

In this research, camera poses are generated from a 3D model in Revit by defining the positions and orientations of cameras to ensure the coverage of inner spaces in the building. A set of camera poses, as illustrated in Figure 2, is the result of our developed algorithm in Revit API. The process in Revit API begins by setting key parameters such as offset distance and camera position spacing. A transaction is started to

handle model modifications, and room boundaries are identified. For each room, the algorithm extracts boundary segments and generates viewpoints along the offset segments. It then determines valid points for camera placement and assigns viewpoints accordingly. Each viewpoint is processed to compute the camera's position and orientation.

In facility management, the inspectors often need to capture numerous images following a specific path. Thereby, camera poses should be determined on such a camera path that simulates the actual inspection path. After that, the model with camera poses is exported to Nvidia Isaac Sim to add more textures of objects and simulated lights and adjust the camera's parameters. The export process from RVT file in Revit to USD file in Isaac is directly supported by the NVIDIA plugin in Revit, ensuring optimal model preservation. Figure 3 shows the simulated environment in Isaac Sim software, in which camera parameters can be adjusted to match the real camera parameters. Other adjustments can also be executed, such as adding realistic materials and changing light intensity. Using this simulation is necessary because the current camera setting in Revit lacks customized functions to simulate the different cameras. The simulation environment in Isaac Sim is then used to render outputs, including RGB images, depth maps, and instance segmentation maps, which were saved with filenames corresponding to the camera names and timestamps for traceability. This process enabled the generation of high-fidelity visualizations of the architectural model, with realistic camera perspectives derived from the Revit-defined poses. Simultaneously, the position (x, y, z) and orientation (in quaternions) of the virtual cameras are exported into a CSV file as preparation for our CNN model training.

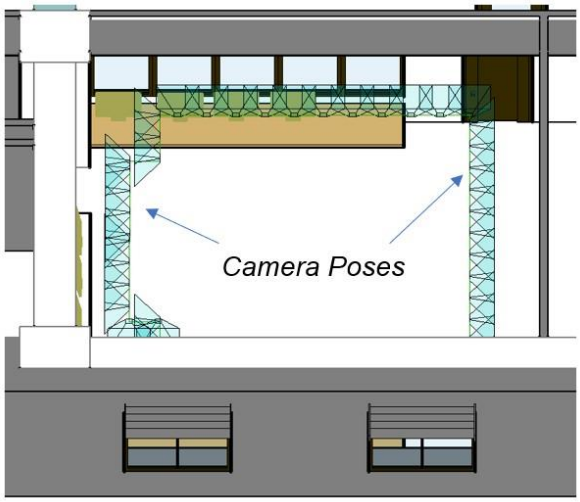


Figure 2. Camera poses defined in Revit

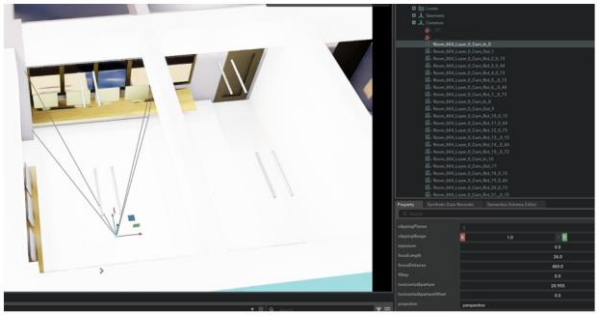

Figure 3. Simulated environment with generated camera poses in Isaac Sim

### 3.3 Image Localization

The localization of the camera pose is performed using a PoseNet-based CNN network, which has been proposed in our previous research [21]. This network is designed with 22 layers in its deep neural network architecture (Figure 4). In this study, the dataset used for training includes RGB and semantic images generated by Isaac Sim software, as well as their relevant 6-DOF poses. The poses are stored in a CSV file and linked to the relevant input images.

At first, the images are resized to 224×224 pixels and normalized using ImageNet-specific mean and standard (mean = [0.485, 0.456, 0.406] and standard = [0.229, 0.224, 0.225]) as required by the pre-trained GoogLeNet. The encoder, which includes convolutional layers and inception modules, extracts features from input images. This module uses pre-trained weight from the original GoogLeNet. An illustration of the inception module can be found in Figure 5. Each inception module captures a deeper level of abstraction from the input data. Originally in GoogLeNet architecture, the final layer of the network, representing the highest level of abstraction, along with

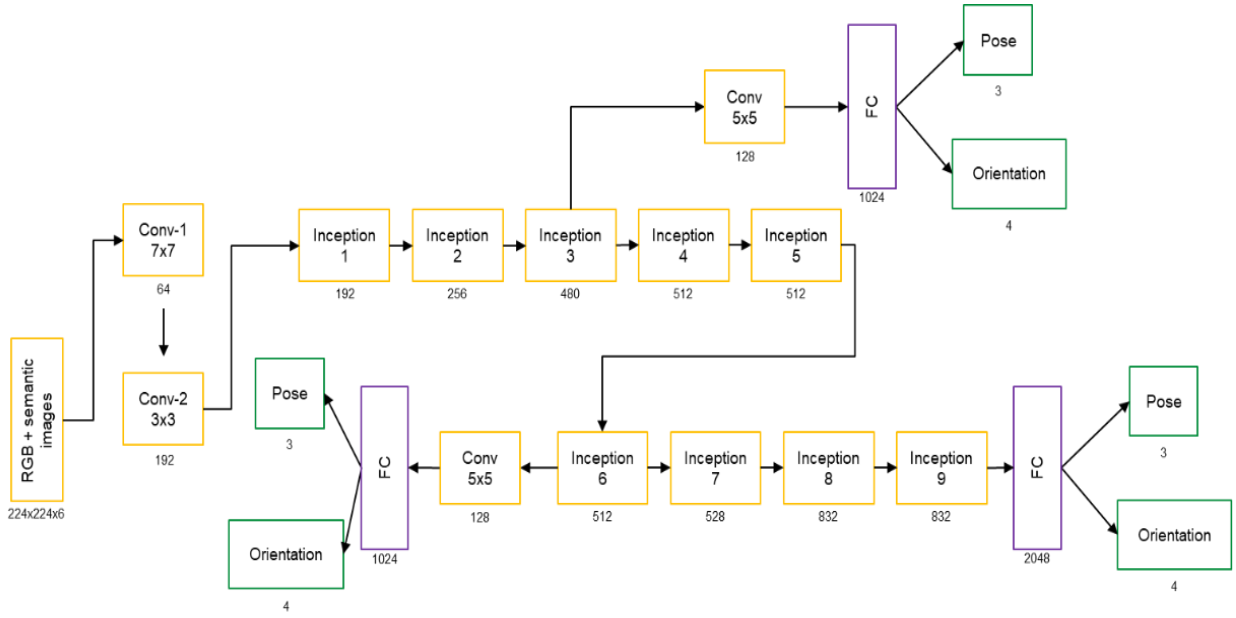


Figure 4. The architecture of CNN-based pose regression network

two intermediate abstraction levels, is passed to fully connected and Softmax layers to classify the objects.

PoseNet, however, replaces the Softmax classifier with two parallel fully connected layers with 3 and 4 units, respectively. These regressors output a 7-dimensional vector representing the camera's 6-DOF pose, which includes the 3D position $\hat{X}_c$ and orientation $\hat{q}_c$ in quaternion format. The network is trained using a loss function that minimizes the position error $\|\hat{X}_c - X_c\|$ and the orientation error $\|\hat{q}_c - q_c\|$ with a balancing hyperparameter to ensure optimal performance. $Xc$ and $qc$ are the ground truth position and orientation of the input camera poses, respectively.

$$Loss(I) = \|\hat{X}_c - X_c\|_2 + \beta\|\hat{q}_c - q_c\|_2 \qquad (1)$$

The hyperparameter $\beta$ is used to balance the weights of position and orientation losses, keeping them with in a similar range. Since position errors and orientation errors are measured in different units and magnitudes, directly optimizing them together may lead to an imbalance, where one dominates the loss function. Thereby, $\beta$ acts as a scaling factor that adjusts the relative contribution of orientation loss compared to position loss. If $\beta$ is too high, orientation accuracy improves at the cost of position accuracy, and vice versa. Previous experiments suggest that $\beta$ is typically assigned values between 250 and 2,000 [17]. The closest camera pose is determined by calculating the distance in position (Euclidean distance) and angular difference in orientation (quaternion angular distance) for each camera pose in the dataset

## 4 Initial Result and Discussion

In our study, an academic building model was used, containing different offices and a long hallway. The synthetic dataset generated by Isaac Sim includes 1922 RGB images and 1922 semantic images. Each camera position has five different orientations. These images' poses are controlled by creating camera paths offset from the room walls. These camera paths simulate the tracks when implementing inspections for facility management. The integration of Revit for camera pose generation, and Isaac Sim for advanced rendering provides an effective workflow for applications such as architectural visualization, training datasets for AI models, or construction progress monitoring. The resulting outputs demonstrate the feasibility of leveraging a combined pipeline for precise and visually rich simulation data.

When preparing the dataset, 80 percent of the images were used to train the pose regression model. Testing and validation used 10% of the data relatively. The testing results show an error of 5.81 meters in translation and 69.05 degrees in orientation. This section aims to evaluate the accuracy of our network for single-image localization using only synthetic images. The goal is to analyze the network's performance when trained exclusively on synthetic data and tested directly on real images without employing any techniques to bridge the domain gap.

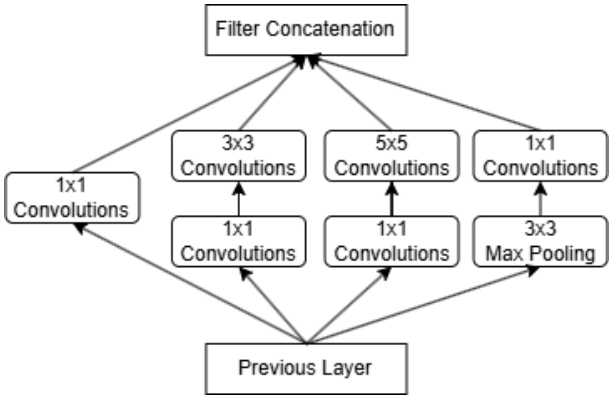


Figure 5. The inception module based on [17]

The test result can be considered acceptable in predicting the pose of an actual image, especially when the model was trained with a small dataset (less than 4000 images compared to other common models). Some causes can be listed for this result. At first, only images of the virtual environment and semantic information were used to train the mode. It is noticeable that there is a vast difference between the simulated environment and real-world space, including the furniture and objects in the scene. We acknowledge the complexity of the environment captured in the real image, including different finishing materials and furniture. Therefore, the images in our synthetic dataset subsequently lack features compared to the images captured in the real world. When using the pre-trained GoogLeNet backbone, the convolutional layers learn to recognize lines and geometric patterns as part of the training process. These features do not exist in the images in the synthetic dataset, thereby issuing challenges for the network. Furthermore, the network's ability to estimate the camera's rotation is less accurate, likely due to the complexity of capturing fine-grained orientation details. Without refinement techniques, the network struggles to bridge this gap, leading to suboptimal performance on real-world data.

For future work, additional features should be utilized to train the regression model. The simulation software Isaac Sim can generate depth information from images, which is essential for predicting spatial details. This input will support the prediction process. On the other hand, PnP and RANSAC algorithms are preferred for refining the poses predicted by the CNN in the earlier stage. The process will use a set of virtual images taken from the initial predicted poses, which serve as the starting point for the pose refinement process. The PnP algorithm leverages 2D-3D point correspondences to estimate the camera's position and orientation, minimizing reprojection error. In detail, it solves the transformation between a set of 2D image points and their corresponding 3D points in a known reference model such as a 3D point cloud. Then, PnP estimates the projection matrix, which aligns the image features with their 3D world coordinates. However, the presence of mismatched or noisy correspondences can degrade accuracy. This is where RANSAC becomes essential, as it ensures that only inliers contribute to the pose estimation. The algorithm repeatedly selects small random subsets of point matches to estimate the pose using PnP and evaluates the consistency of the remaining points. By using RANSAC, mismatched feature correspondences or erroneous keypoints are rejected, ensuring that only inliers contribute to the final pose estimation. As validated in our previous study [21], PnP and RANSAC enhance the robustness and precision of camera pose predictions, making them particularly effective for scenarios involving noise or incomplete data.

## 5 Conclusion

In conclusion, this study introduces an automated method for creating photorealistic synthetic images of BIM scenes depicting building interiors. In addition, a deep learning-based framework for image localization tailored to challenging and dynamic construction environments has been presented. The proposed framework includes two stages: the first stage involves synthetic dataset generation using BIM software and an advanced simulation platform; the second is a CNN-based pose regression. The pose regression model can use inputs, including RGB images and semantic images for prediction. The findings demonstrated that currently, using only a synthetic dataset generated in advanced simulation software can be reliable for training a CNN-based pose regression model. The network performs well for the initial coarse localization. However, in order to improve the performance of localizing a real-world image, implementing pose refinement will be the next step in our research. Future work will improve this initial prediction by matching the query image with virtual images generated from the BIM model, using geometric pose optimization methods such as PnP and RANSAC. The refined poses ensure higher accuracy, enabling the alignment of real images with the virtual model. This combination of synthetic data, deep learning, and geometric optimization creates a robust pipeline for localizing images in complex and evolving construction environments.

## References

[1] M. A. Fischler and R. C. Bollers. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381-395, 1981

[2] M. Pollefeys, et al. Visual Modeling with a Hand-Held Camera. *International Journal of Computer Vision*, 59(3):207-232, 2004

[3] T. Sattler, B. Leibe and L. Kobbelt. Efficient & Effective Prioritized Matching for Large-Scale Image-Based Localization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(9):1744-1756, 2017

[4] M. W. Smith, J. L. Carrivick and D. J. Quincey. Structure from motion photogrammetry in physical geography. *Progress in Physical Geography: Earth and Environment*, 40(2):247-275, 2016

[5] A. Kendall and R. Cipolla. Geometric loss functions for camera pose regression with deep learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5974-5983, 2017

[6] J. M. H. Noothout, et al. Deep Learning-Based Regression and Classification for Automatic Landmark Localization in Medical Images. *IEEE Transactions on Medical Imaging*, 39(12):4011-4022, 2020

[7] Y. Shavit and R. Ferens. Introduction to camera pose estimation with deep learning. *arXiv preprint arXiv:1907.05272*, 2019

[8] D. Acharya, K. Khoshelham and S. Winter. BIM-PoseNet: Indoor camera localisation using a 3D indoor model and deep learning from synthetic images. *ISPRS Journal of Photogrammetry and Remote Sensing*, 150:245-258, 2019

[9] Y. Liu, T. S. Huang and O. D. Faugeras. Determination of camera location from 2-D to 3-D line and point correspondences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):28-37, 1990

[10] V. Lepetit, F. Moreno-Noguer and P. Fua. EPnP: An Accurate O(n) Solution to the PnP Problem. *International Journal of Computer Vision*, 81(2):155-166, 2009

[11] L. Kneip, H. Li and Y. Seo. UPnP: An Optimal O(n) Solution to the Absolute Pose Problem with Universal Applicability. In pages 127-142,Cham, 2014

[12] Y. Zheng, Y. Kuang, S. Sugimoto, K. Astrom and M. Okutomi. Revisiting the pnp problem: A fast, general and optimal solution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2344-2351, 2013

[13] M. Fiala. ARTag, a fiducial marker system using digital techniques. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, pages 590-596 vol. 2, 2005

[14] E. Olson. AprilTag: A robust and flexible visual fiducial system. In *2011 IEEE International Conference on Robotics and Automation*, pages 3400-3407, 2011

[15] X. Bai, M. Huang, N. R. Prasad and A. D. Mihovska. A Survey of Image-Based Indoor Localization using Deep Learning. In *2019 22nd International Symposium on Wireless Personal Multimedia Communications (WPMC)*, pages 1-6, 2019

[16] D. R. d. Santos, M. A. Basso, K. Khoshelham, E. d. Oliveira, N. L. Pavan and G. Vosselman. Mapping Indoor Spaces by Adaptive Coarse-to-Fine Registration of RGB-D Data. *IEEE Geoscience and Remote Sensing Letters*, 13(2):262-266, 2016

[17] A. Kendall, M. Grimes and R. Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE international conference on computer vision*, pages 2938-2946, 2015

[18] D. Jia, Y. Su and C. Li. Deep convolutional neural network for 6-dof image localization. *arXiv preprint arXiv:1611.02776*, 2016

[19] R. Zhang, Z. Luo, S. Dhanjal, C. Schmotzer and S. Hasija. PoseNet++: A CNN framework for online pose regression and robot re-localization. *Tech. Rep*, 2018

[20] M. S. Müller, S. Urban and B. Jutzi. Squeezeposenet: Image based pose regression with small convolutional neural networks for real time uas navigation. *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-2/W3:49-57, 2017

[21] T. H. Wang, A. Pal, J. J. Lin and S.-H. Hsieh. Construction Photo Localization in 3D Reality Models for Vision-Based Automated Daily Project Monitoring. *Journal of Computing in Civil Engineering*, 37(6):04023029, 2023

[22] D. Acharya, S. Singha Roy, K. Khoshelham and S. Winter. A Recurrent Deep Network for Estimating the Pose of Real Indoor Images from Synthetic Image Sequences. *Sensors*, 20(19):5492, 2020

[23] Y. Hong, S. Park, H. Kim and H. Kim. Synthetic data generation using building information models. *Automation in Construction*, 130:103871, 2021

[24] H. Ying, R. Sacks and A. Degani. Synthetic image data generation using BIM and computer graphics for building scene understanding. *Automation in Construction*, 154:105016, 2023