2023 European Conference on Computing in Construction
40th International CIB W78 Conference
Heraklion, Crete, Greece
July 10-12, 2023

Towards fully automatic Scan-to-BIM: A prototype method integrating deep neural networks and architectonic grammar

Yijie Wu[1], Maosu Li[1], and Fan Xue[1]

[1]Faculty of Architecture, The University of Hong Kong, Pokfulam, Hong Kong SAR, China

This is the post-print version of the conference paper:

Wu, Y., Li, M., & Xue, F. (2023). Towards fully automatic Scan-to-BIM: A prototype method integrating deep neural networks and architectonic grammar. Proceedings of the 2023 European Conference on Computing in Construction and the 40th International CIB W78 Conference, 257, Heraklion, Crete, Greece. https://doi.org/10.35490/EC3.2023.257

The final version: https://ec-3.org/publications/conference/paper/?id=EC32023_257 .

Abstract

Building Information Modeling (BIM) has presented great potential in the construction industry. Scan-to-BIM is demanded to verify as-designed models and to digital twin many existing buildings without BIMs. This paper focuses on an extreme case without human interferences like data cleansing and partitioning – fully automatic Scan-to-BIM, on which recent advances in 3D scanning and deep neural networks (DNNs) shed light. This paper presents a prototype FLKPP that integrates DNNs with architectonic grammar for fully automatic Scan-to-BIM. FLKPP won the 2nd place in 3D reconstruction ($F_1^{5cm} = 0.316$, $F_1^{10cm} = 0.454$, $F_1^{20cm} = 0.584$) and the 3rd place in the 2D reconstruction in the 2nd International Scan-to-BIM Challenge. Nevertheless, the results of all methods in the challenge were still limited in training data quality, DNN architecture, and inconsistency between DNN predictions and BIM reconstruction; the observations indicate a long way leading to the complete automation of Scan-to-BIM.

Introduction

Building Information Modeling (BIM) is becoming an essential practice to optimize the whole building life cycle from planning to demolition (NBIMS-US, 2022). The adoption of BIMs at the urban scale can help with fine-scale city information modeling and urban analytics (Batty, 2013; Xue et al., 2020; Li et al., 2022). However, the restricted efficiency of creating and updating the as-is or as-built Building Information Models (BIMs) of existing or even aged buildings is still a crucial hinder against the trend of BIM (Esfahani et al., 2021). To create as-is/as-built BIMs, 3D scanning, in particular Light Detection and Ranging (LiDAR), is nowadays prevalent to capture the realistic and detailed surface status of building interior rapidly (Bosché et al., 2015). Yet, it is still fairly time-consuming to manually create BIMs from 3D scans through burdensome editing. Hence, automatic Scan-to-BIM is highly desired in the construction field for years, which attracts significant efforts from both academia and industry in developing this automation (Wang et al., 2019; Xue et al., 2021; Wu et al., 2021).

However, the full automation of Scan-to-BIM remains challenging due to the complexity and diversity of the building interior as well as the data deficiencies of 3D scanning. The automation of Scan-to-BIM can be roughly decomposed into two fundamental tasks. The first task is to understand the semantics of the 3D scans, such as recognizing the walls, slabs, and columns in the point cloud. It could be extremely difficult and onerous to manually craft rules for recognizing different architectural structures with diverse styles or flexible designs. The second task aims at the topologically-consistent and parametric reconstruction of geometry, such as estimating the shapes, locations, orientations, and dimensions of wall solids without conflicts. The challenges encountered by this task are to guarantee geometric accuracy despite the inevitable noise, outliers, nonuniform sampling, and missing parts of scans (Berger et al., 2017).

In recent years, point cloud processing has been largely promoted by deep learning by the community of computer vision (CV), which led to advanced tools especially for semantics understanding to equip the Scan-to-BIM. One of the most significant advances is the 3D semantic segmentation based on deep neural networks (DNNs). On top of the public and large-scale 3D datasets (Armeni et al., 2016; Dai et al., 2017), DNNs, such as PointNet++ (Qi et al., 2017) and KPConv (Thomas et al., 2019), has been designed and trained to predict the semantic categories of each point in the 3D scans of building interior, equivalently segmenting a scan into clusters belong to different physical objects. Another main track that also aims at semantics understanding is the 3D object detection (Qi et al., 2018) which would benefit the recognition and reconstruction of furniture or small objects in building interior. Meanwhile, the features of points can be learned through DNNs for correspondence matching and registration between multiple scans (Aoki et al., 2019; Choy et al., 2020). Besides, existing studies also parsed point clouds from the top views to simultaneously extract the semantics and reconstruct floor plans (Liu et al., 2018; Chen et al., 2019).

Although the DNN-based point cloud processing developed by the CV community inspires Scan-to-BIM in semantics understanding and geometry reconstruction, there are still crucial gaps in adopting DNNs for fully automatic Scan-to-BIM. In this paper, we dissect these gaps from the input and output sides. The common issue on the input side is that the present large-scale datasets for benchmarking DNN-based algorithms are substantially simpler and smaller than those used in AEC practice. More particular, the datasets for 3D semantic segmentation are limited to room scale, although in practice, an initial scan may include parts of multiple rooms and a finished scan may span the entire building story. Meanwhile, there could

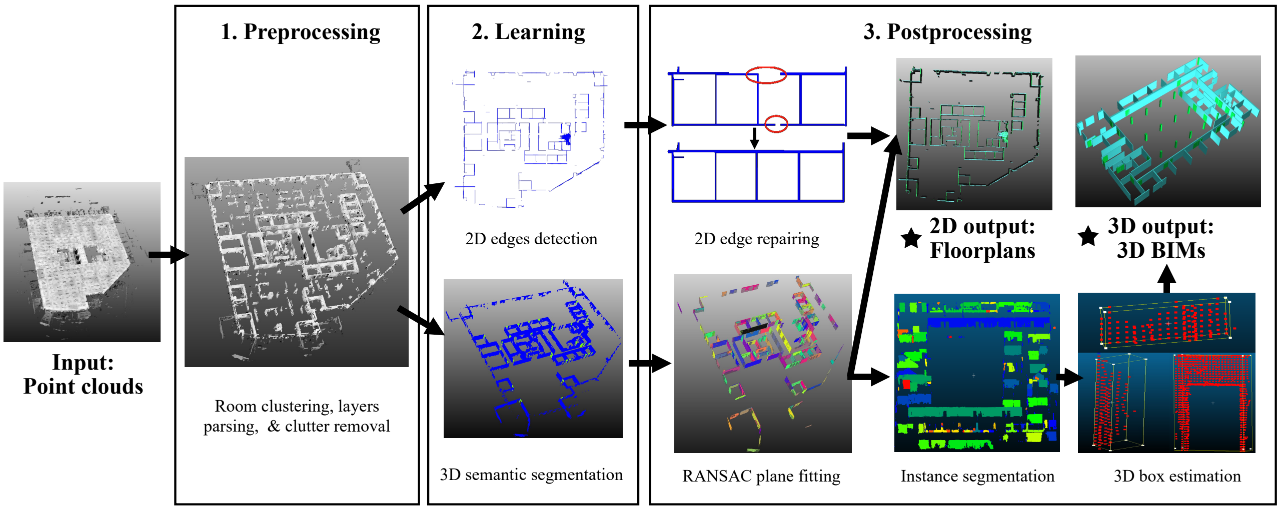

Figure 1: Overview of the prototype: Floor Layer-based Kernels and Pillars of Points (FLKPP).

be fewer noises, outliers, or missing parts in the current datasets, which also contributes to the high accuracy reported by the current DNN-based 3D semantic segmentation. Hence, without careful preprocessing, there could be accuracy gaps between the reported ones and the actual adoption in Scan-to-BIM. From the output side, the target results as 3D BIMs or 2D floor plans presented different characteristics from the 3D point-wise prediction. The end branches of DNNs should be modified to generate line drawings or 3D parametric primitives, or postprocessing should be used to 'translate' the point-wise prediction into parametric 2D or 3D components.

To fill the above-mentioned gaps and maximize the learning ability of DNNs, this paper presents a prototype integrating DNNs and architectonic grammar for fully automatic Scan-to-BIM. Architectonic grammar, such as the planarity, angle regularity, alignment, and adjacency of architectural structures as well as the furniture distribution of building interior, could be harnessed as useful *a prior* in the preprocessing, network design, and postprocessing of DNNs. In this paper, the proposed prototype involves preprocessing to clean and simplify the input data for DNNs based on furniture distribution. For the adoption of DNNs, except for embedding state-of-the-art (SOTA) DNNs directly in the prototype, the DNN to reconstruct floor plans has been specially designed for estimating 2D line drawings more efficiently. After the learning procedure, postprocessing has been designed to finalize or repair the prediction of DNNs and output the floor planes and 3D BIMs. The prototype, FLKPP, is named after the **f**loor-**l**ayer based preprocessing, semantic segmentation through **KP**Conv (Thomas et al., 2019), and the inputs as **p**oint **p**illars (Lang et al., 2019) for the DNN to reconstruct floor plans.

The proposed FLKPP is also the entry that won the second and third places, respectively in the 3D and 2D track of the 2nd International Scan-to-BIM challenge hosted by the 2nd Workshop on Computer Vision in the Built Environment, which was held in conjunction with IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2022. Meanwhile, the workshop aims at connecting fields AEC and CV and devotes to benchmarking, promoting, and communicating the fully automatic solutions of Scan-to-BIM. A large-scale dataset aligned to the practical scenarios in AEC has been released, and the Scan-to-BIM challenge, including floor plan reconstruction (2D track) and 3D building model reconstruction (3D track), has been held based on the dataset.

The remainder of this paper introduces FLKPP, reports the results of our entry on the 2nd Scan-to-BIM Challenge, and discusses the integration of DNNs with architectonic grammar for fully automatic Scan-to-BIM as well as the limitations and future improvement of the current FLKPP.

The proposed FLKPP

Overview

Fig. 1 shows the overview of our prototype. The pipeline takes the point cloud of a whole building story as input and outputs 2D floor plans and 3D BIMs. Three stages are designed to prepare the inputs of DNNs, i.e., **Stage 1 'Preprocessing'**, involve DNNs, i.e., **Stage 2 'Learning'**, and finalize their prediction into floor plans and 3D BIMs, i.e., **Stage 3 'Postprocessing'**.

Preprocessing

The preprocessing is designed according to architectonic grammar for cleaning and simplifying the input point clouds by removing the furniture clutters and preserving the architectural structures. More specifically, the preprocessing extracts the points of or close to the walls, ceilings, and edges. Then, the remaining points can be grouped into clusters of different rooms and removed to let the DNNs focus on the architectural structures for Scan-to-BIM. For computational efficiency, the point cloud is first voxelized to label the occupied voxels close to the edges, ceilings, and walls, as illustrated in the left part of Figure 2. The labeling is performed iteratively in terms of whether the neighborhoods on the top and sides of a voxel are also occupied. After removing the clutters, the remaining point cloud mainly covering the architectural structures can be separated into the upper, middle, and lower layers. The point densities of the three layers are distributed according to the locations and dimensions of architectural structures, which provides hints for the following semantic understanding and geometry reconstruction.

Learning

In this stage, we explored different schemes to use DNNs for 2D and 3D reconstruction. 3D reconstruction can be derived naturally and directly from the 3D semantic segmentation based on DNNs, while 2D reconstruction does not need to rely on the 3D segmentation. Hence, for 2D reconstruction, the 3D scans are flattened features of 2D grids from the top view. We then designed a more compact representation as the input of the DNN and chose the proper top branches to predict the line drawings directly.

Learning for 2D reconstruction

As illustrated in Figure 3, a horizontal 2D grid is created for the input point cloud. The input point cloud is then converted into the feature of each cell in the grid, forming a compact representation that can be the input of any standard 3D convolutional network architecture. The feature of each cell looks like a vertical column and is thus named as point pillar (Lang et al., 2019), which presents much higher efficiency in semantics understanding task from the top view. As the preprocessing cleans and separate the point cloud into three floor layers, the normalized distribution of the point density on each floor layer can serve as a simple yet distinguishable feature to recognize the architectural structures. Meanwhile, considering the reconstruction of floor plans are equivalent to predicting a set of lines and thanks to point pillars embedded in the 2D grids, the DNNs designed for the line detection of 2D images can be naturally used in predicting the line drawings as floor plans, as shown at the far right of Figure 3. Therefore, the 'ConvNet' in Figure 3 to predict the corners and edges from the point pillars can be replaced with the SOTA DNNs for line detection, such as L-CNN (Zhou et al., 2019) and deep Hough Tranformation-based line detection (Zhao et al., 2021). In the implementation of the Challenge, L-CNN was used.

Learning for 3D reconstruction

For 3D reconstruction, we directly adopt the SOTA DNNs of 3D semantic segmentation to label the points into different categories, including walls, doors, columns, and others. The input of the DNN is the point cloud cleaned by preprocessing. Pilot tests on the Challenge dataset proved that the IoU of 3D semantic segmentation could be significantly improved with data after cleaning. Furthermore, KPConv (Thomas et al., 2019) was adopted in our prototype because it achieved the highest Intersection-of-Union (IoU) in our pilot experiments on the dataset of the Scan-to-BIM Challenge. Besides, the instance segmentation among the same categories and the 3D geometric reconstruction are left to the postprocessing.

Postprocessing

Postprocessing for 2D reconstruction

Thanks to the DNN for 2D reconstruction predicting the line drawings directly, only repairing is required in the postprocessing for the reconstruction of floor plans. As shown in the upper of Figure 4, there are gaps between the predicted lines. The postprocessing reconnects them based on the probability of the line map simultaneously estimated by the DNN for 2D reconstruction and the room map generated by the room clustering in the preprocessing (as presented in the right-hand side of Figure 2). Figure 5 presents the mentioned maps of lines and rooms, while the repaired floor plan is shown in the lower of Figure 4.

Postprocessing for 3D reconstruction

The postprocessing of 3D reconstruction should 'translate' the points labeled as wall, door, and column into 3D parametric instances. For points labeled as the wall, the postprocessing splits them into different groups according to their orientations. Next, in each group, the points are further clustered by DBSCAN (Ester et al., 1996) to separate them into individual walls. The instance segmentation of doors and columns is conducted in a similar way as the wall segmentation. For the doors connected to each other as shown in the middle of Figure 6, we threshold the width of the door instances and clip those larger than the threshold into parts. After the instance segmentation, the postprocessing estimates the Manhattan bounding box of each instance and store the results in the IFC standard.

Entry results on 2nd Scan-to-BIM Challenge

Dataset, tracks, and evaluation metrics

The whole dataset of this Challenge, including those for the training and validation of both the 2D and 3D tracks, contains about 80 point clouds, each covering one building story. Examples are shown in Figure 7. The number of points in each point cloud could be up to a billion. Office and parking lots are the top 2 scenes in the dataset. Besides, as presented in Figure 8, floor plans as line drawings and the parametric 3D instances with semantic tags are provided as the ground truth for the 2D and 3D tracks respectively. The evaluation of 2D reconstruction involves both the geometric and topological metrics, including the IoU, endpoint accuracy and orientation deviation of the predicted segments, warping error (Jain et al., 2010), and Betti number error. Meanwhile, the current 3D evaluation metrics only concern the geometry, including the 3D IoU of the walls' bounding boxes and the accuracy of endpoints. Readers can refer to the Challenge website (`https://cv4aec.github.io/`) for more details about the dataset, tracks, and evaluation.

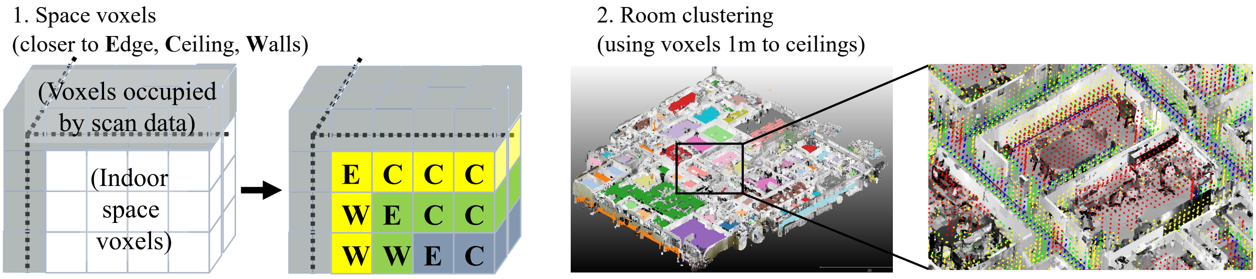

Figure 2: Illustration of the preprocessing.

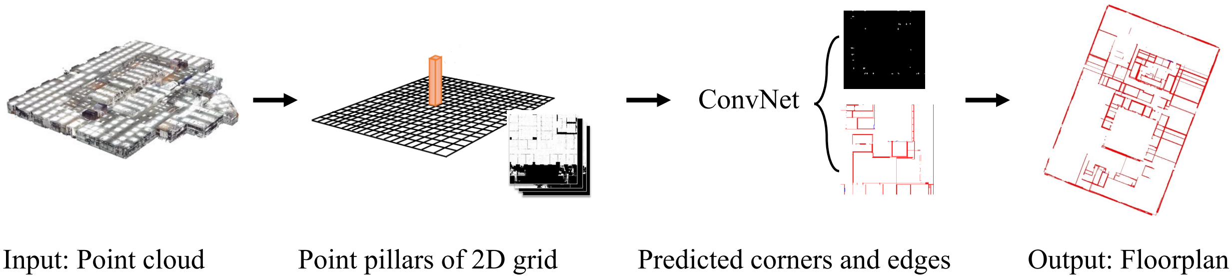

Figure 3: Pipeline of the learning for 2D reconstruction.

Figure 4: Postprocessing of the 2D reconstruction.

Figure 5: Line map and room map for repairing the floor plans predicted by the DNN.

2D reconstruction

Table 1 shows the evaluation results of FLKPP and other entries. Our prototype came 3rd in the 2D track. The precision and recall of endpoints at all the evaluation resolutions, i.e., 5, 10, and 20 cm, are still far from satisfactory. The highest value is still lower than 0.4, regardless of those metrics less than 0.1 at the highest resolution which is still much more coarse than the mm accuracy requirement common in AEC. The remaining three metrics in Table 1 mainly reflect the closeness of space split by the lines of floor plans. The IoU of FLKPP is 0.374, while the winner's achieved 0.657. Fig. 9 presents selected results of our 2D reconstruction. FLKPP has captured most main architectural structures on the floor plans. Yet, there still are some gaps between the reconstructed segments, resulting in an insufficient IoU. Meanwhile, the clusters of messy lines can be spotted in the upper two examples in Fig. 9 due to the unstable line prediction at stairs by the DNN.

3D reconstruction

The evaluation results of the top 4 entries are reported in Table 2, and FLKPP ranks second. Compared with the 2D reconstruction, the precision and recall of endpoints, reported as F1-measure in Table 2, are much higher. Yet, the 3D IoUs of architectural structures reconstructed by FLKPP and other entries still should be much improved in the future. As shown in Fig. 10, despite fairly reconstructing most architectural structures, some small structures and connections between the adjacent walls are still missing in the current reconstruction.

Discussion

Integrating DNNs with architectonic grammar for fully automatic Scan-to-BIM

The rapid development of DNNs presents spectacular potential in promoting point cloud processing and automatic Scan-to-BIM. However, careful integration is needed to

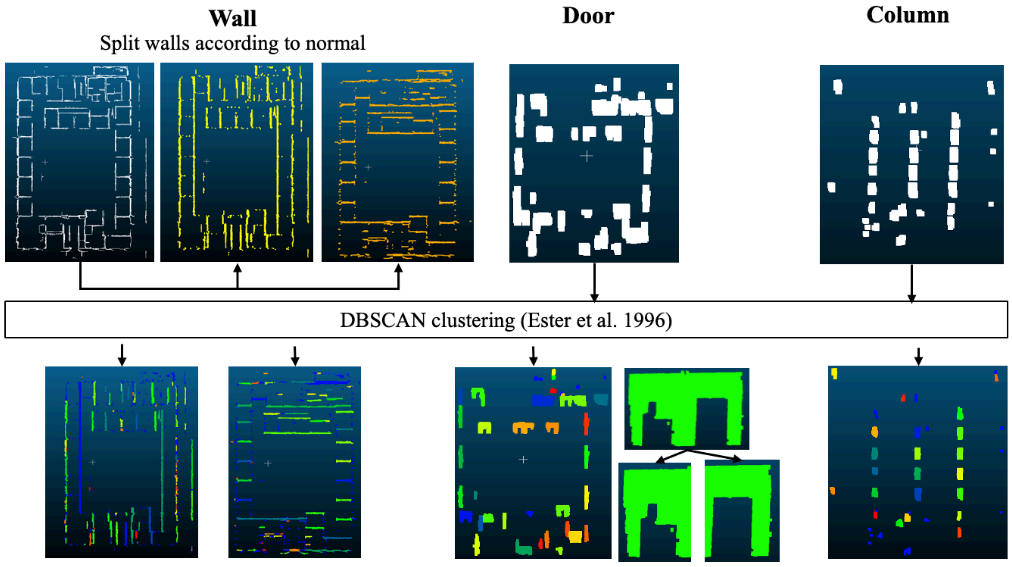

Figure 6: Postprocessing of the 3D reconstruction.

Table 1: Top 4 entries of the 2D reconstruction.

Method	Precision			Recall			IoU	Warping Error	Betti Error
	5cm	10cm	20 cm	5cm	10cm	20cm			
Seg2Plan	**0.052**	**0.203**	**0.335**	0.015	0.065	0.114	**0.657**	0.249	**1.076**
S2FP	0.02	0.085	0.146	**0.048**	**0.22**	**0.375**	0.517	**0.188**	1.14
FLKPP	0.016	0.068	0.132	0.032	0.129	0.253	0.374	0.258	1.128
VecIM	0	0	0	0	0	0	0	0.731	1.646

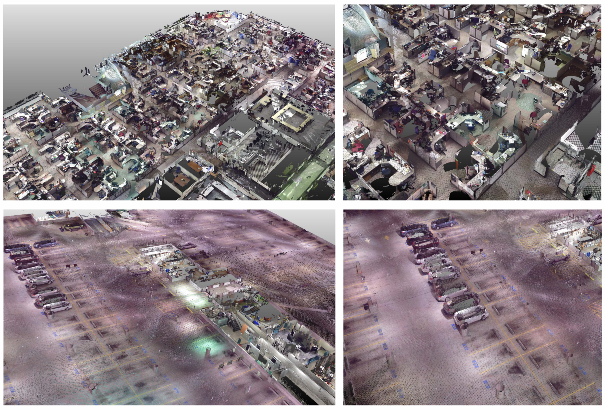

Figure 7: Screenshots of the dataset. The right column presents the zoom-in views of their left.

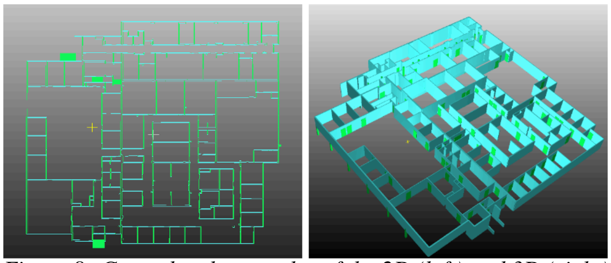

Figure 8: Groundtruth examples of the 2D (left) and 3D (right) tracks.

handle the issues raised by the data quality, choose and adjust the network architectures, and link the prediction of networks to the final reconstruction. The proposed FLKPP is a prototype that explores solutions for optimal integration.

1. Training data quality. The SOTA DNNs in point cloud processing, in particular the 3D semantic segmentation, are mainly trained on datasets that are much simpler than the practical scans in AEC. Many of them are confined to the room scale, while the practically finalized scans usually cover the whole building story. Meanwhile, those datasets have much less noise, outliers, and occlusions. Such data condition is difficult for the real scanning in AEC to guarantee, resulting in the accuracy gaps between the reported metrics by SOTA DNNs and real applications. Besides, the relevant DNNs usually need supervision with large-scale labeling. The existing CAD drawings and 3D models of buildings can serve as an off-the-shelf data source to significantly ease the labeling of 3D scans. However, due to the building innovation and own errors in the drawings or models, there could be inevitable deviations from actual situations captured by the scans, which would undermine the prediction of DNNs.

2. Network architecture. Although existing DNNs on both 3D and image processing provide numerous options to embed the deep learning in Scan-to-BIM, it is beneficial for efficiency, robustness, and simplicity to

Table 2: Top 4 entries of the 3D reconstruction.

Method	IoU				Average F1			10cm F1		
	Average	Column	Door	Wall	5cm	10cm	20cm	Column	Door	Wall
Seg2BIM	**0.309**	**0.47**	**0.26**	**0.266**	**0.417**	**0.515**	0.577	0.618	**0.494**	**0.477**
FLKPP	0.231	0.372	0.23	0.152	0.316	0.454	**0.584**	0.608	0.367	0.452
PointToBIM	0.17	0.396	0.061	0.15	0.276	0.366	0.448	**0.633**	0.165	0.415
BoxDetector	0.024	0.038	0.006	0.033	0.109	0.171	0.258	0.167	0.144	0.197

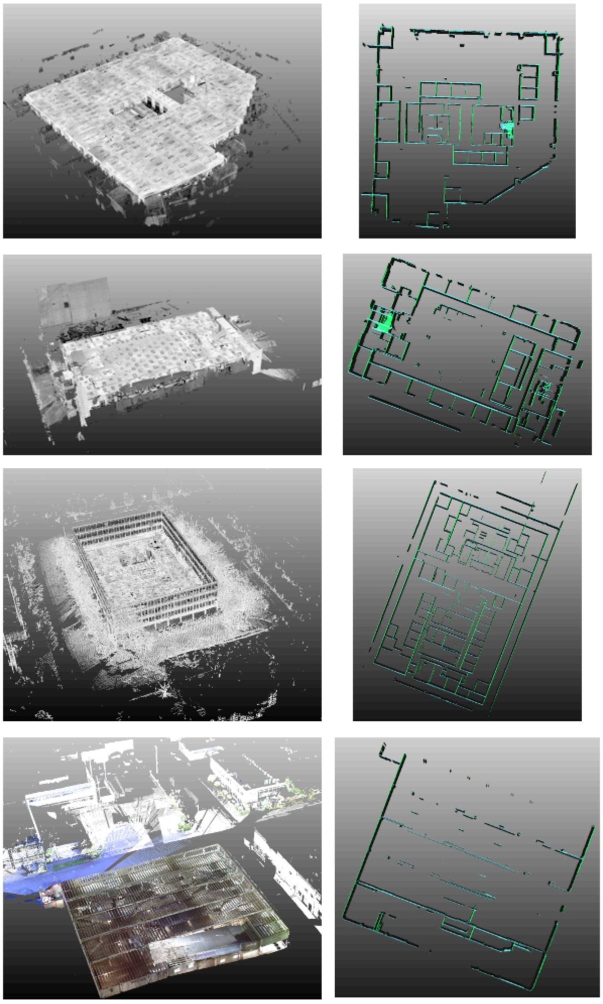

Figure 9: Examples of 2D reconstruction by FLKPP.

make an optimal choice and integration. 3D semantic segmentation and object detection are the most direct choices for Scan-to-3DBIM. For the 2D reconstruction, however, it is not necessarily the same case. We believe that it could be more efficient to flatten the 3D data as 2D grids and use the DNNs of image processing, which is recognized as more mature than that of 3D processing.

3. Inconsistency between DNN predictions and BIM reconstruction. Without altering the output layers of DNNs, there could be 'a last mile' from the prediction of DNNs to the final 2D and 3D reconstruction. Two options to complete this last mile are 1) reframing the last layers of DNNs to generate floor plans or 3D parametric instances directly or 2) postprocessing DNNs' prediction into floor plans or 3D parametric instances according to appropriate architectonic grammar. Note that there may still be a need for further refinement based on architectonic grammar even choosing the first option, which depends on the quality of network prediction and the requirement on the reconstruction quality.

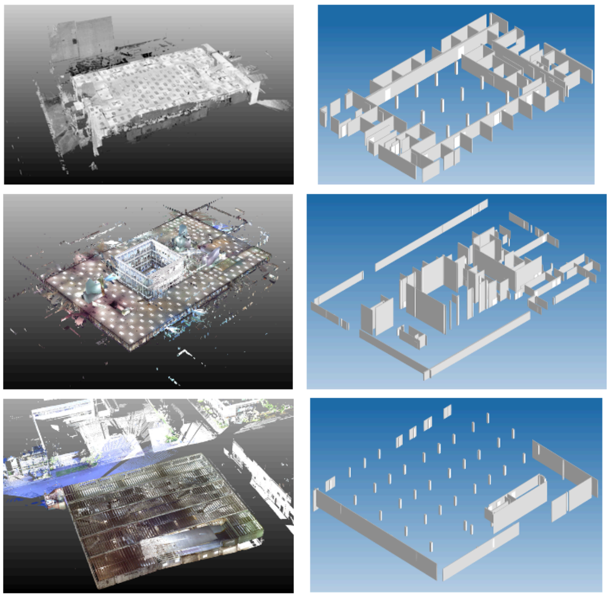

Figure 10: Examples of 3D reconstruction by FLKPP.

FLKPP presents a prototype taking the above three aspects into account. The preprocessing removes many of the unwanted clutters of the point clouds to simplify the input of DNN-based 3D semantic segmentation. Meanwhile, the preprocessing also separates the cleaned point cloud into floor layers, which leads to shorter point pillars for the learning of 2D reconstruction. In the learning stage, to improve the efficiency of learning, FLKPP converts the 3D scans into the point pillars of a 2D grid and frames the 2D reconstruction as the line prediction based on DNN, leading to the end-to-end learning for Scan-to-Floorplan. But refinement to fill the gaps between segments predicted by the DNN is still needed in our implementation. For 3D reconstruction, FLKPP uses a SOTA DNN of 3D semantic segmentation directly with the postprocessing to further segment and estimate the bounding boxes of the wall, column, and door instances.

Limitations of the current FLKPP

Results reported on the Challenge in Table 1 and 2 reveal that there is much space for the full automation of Scan-to-BIM to improve its accuracy. The accuracy bottlenecks partly originated from the DNNs themselves and also partly from the integration for Scan-to-BIM. For FLKPP, the limitations and further improvements are listed below.

1. Closeness and topology. Both 2D and 3D reconstruction of FLKPP lack explicit mechanisms to guarantee the closeness of the split space in a building story and the topological consistency around adjacent architectural structures. This issue results in the missing connections between adjacent walls and thus a lower IoU in the 2D track. Inspired by the presentations of other entries especially the winner, Seg2BIM or Seg2Plan, an initial prediction can be extended to intersect those other non-parallel predictions, and a selection based on integer optimization can be conducted to repair the disconnection.

2. Clutter removal. Although the preprocessing largely removes the unwanted clutter for the following reconstruction, the heuristic rules used could be problematic if the ceiling scans do not form large planes, resulting in unstable predictions of the 2D learning as shown in the bottom row in Fig. 9. More robust preprocessing should be designed to handle such cases.

3. Endpoint accuracy. As shown in Table 1 and 2, the precision, recall, and F1-measure of endpoints in both tracks are far from satisfactory, especially in the 2D reconstruction. Except for improving the postprocessing in the future, this issue can be traced back to the deviation between the 2D ground truth, i.e., existing CAD drawings of the scanned buildings, and the actual situations in 3D scans. Meanwhile, there could be no physical correspondences of the segment endpoints in CAD drawings, which confuses the DNN training for 2D reconstruction. Specific mechanisms to handle the issues of using existing CAD drawings for supervision should be designed.

Conclusion

The efficiency and automation of creating as-is/as-built BIMs are highly desired for BIM popularization in AEC. Despite the rapid advance of 3D scanning and deep neural networks, fully automatic Scan-to-BIM still encounters many challenges. To adopt DNNs in this automation, careful designs are needed to handle the issues caused by data quality in real scenarios in the industry. Meanwhile, a robust pipeline of Scan-to-BIM based on DNNs also relies on the appropriate selection or adjustment of network architectures as well as the linking from the prediction of DNNs to the target BIM reconstruction.

This paper presents a prototype, FLKPP, successfully integrating DNNs with architectonic grammar and aiming at the full automation of Scan-to-BIM. FLKPP consists of three stages, i.e., preprocessing, learning, and postprocessing, and performs both floor plan (2D) and 3D BIM reconstruction. The preprocessing cleans and tiers input point clouds to ease and improve the prediction quality of the following learning. Next, FLKPP explores two different network architectures for 2D and 3D reconstruction. We directly adopt SOTA DNN-based 3D semantic segmentation for 3D reconstruction, while the 2D counterpart converts 3D scans into point pillars of 2D girds and frames floor plan reconstruction as DNN-based line prediction. The postprocessing then turns the 3D segmentation into parametric walls, doors, and columns.

FLKPP won the 2nd and 3rd places on the 2nd International Scan-to-BIM Challenge, which validated the prototype on a large-scale dataset close to real situations in AEC. However, the reported metrics of all entries in the Challenge are still far from satisfactory. The precision and recall of the endpoints in floor plans could be lower than 0.1. Significant efforts towards the accurate automation of Scan-to-BIM are required in the future to improve both the direct predictions of DNNs and the integration of learning and architectonic grammar.

Acknowledgments

The work described in this paper was supported by Hong Kong Research Grants Council (No. 27200520) and in part by Hong Kong Innovation and Technology Commission (No. ITP/004/23LP) and Natural Science Fund of Guangdong (2023A1515010757).

References

Aoki, Y., Goforth, H., Srivatsan, R. A., and Lucey, S. (2019). Pointnetlk: Robust & efficient point cloud registration using pointnet. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 7163–7172.

Armeni, I., Sener, O., Zamir, A. R., Jiang, H., Brilakis, I., Fischer, M., and Savarese, S. (2016). 3d semantic parsing of large-scale indoor spaces. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1534–1543.

Batty, M. (2013). The New Science of Cities. MIT press.

Berger, M., Tagliasacchi, A., Seversky, L. M., Alliez, P., Guennebaud, G., Levine, J. A., Sharf, A., and Silva, C. T. (2017). A survey of surface reconstruction from point clouds. In Computer graphics forum, volume 36, pages 301–329. Wiley Online Library.

Bosché, F., Ahmed, M., Turkan, Y., Haas, C. T., and Haas, R. (2015). The value of integrating Scan-to-BIM and Scan-vs-BIM techniques for construction monitoring using laser scanning and BIM: The case of cylindrical MEP components. Automation in Construction, 49:201–213.

Chen, J., Liu, C., Wu, J., and Furukawa, Y. (2019). Floor-sp: Inverse cad for floorplans by sequential room-wise shortest path. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 2661–2670.

Choy, C., Dong, W., and Koltun, V. (2020). Deep global registration. In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pages 2514–2523.

Dai, A., Chang, A. X., Savva, M., Halber, M., Funkhouser, T., and Nießner, M. (2017). Scannet: Richly-annotated 3d reconstructions of indoor scenes. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 5828–5839.

Esfahani, M. E., Rausch, C., Sharif, M. M., Chen, Q., Haas, C., and Adey, B. T. (2021). Quantitative investigation on the accuracy and precision of scan-to-bim under different modelling scenarios. Automation in Construction, 126:103686.

Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In kdd, volume 96, pages 226–231.

Jain, V., Bollmann, B., Richardson, M., Berger, D. R., Helmstaedter, M. N., Briggman, K. L., Denk, W., Bowden, J. B., Mendenhall, J. M., Abraham, W. C., et al. (2010). Boundary learning by optimization with topological constraints. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pages 2488–2495. IEEE.

Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J., and Beijbom, O. (2019). Pointpillars: Fast encoders for object detection from point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12697–12705.

Li, M., Xue, F., Wu, Y., and Yeh, A. G. O. (2022). A room with a view: Automatic assessment of window views for high-rise high-density areas using City Information Models and deep transfer learning. Landscape and Urban Planning, 226:104505.

Liu, C., Wu, J., and Furukawa, Y. (2018). Floornet: A unified framework for floorplan reconstruction from 3d scans. In Proceedings of the European Conference on Computer Vision (ECCV), pages 201–217.

NBIMS-US (2022). National BIM Standard-United States V4. `https://www.nationalbimstandard.org/`. Accessed: 2022-12-29.

Qi, C. R., Liu, W., Wu, C., Su, H., and Guibas, L. J. (2018). Frustum pointnets for 3d object detection from rgb-d data. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 918–927.

Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. Advances in neural information processing systems, 30.

Thomas, H., Qi, C. R., Deschaud, J.-E., Marcotegui, B., Goulette, F., and Guibas, L. J. (2019). Kpconv: Flexible and deformable convolution for point clouds. In Proceedings of the IEEE/CVF international conference on computer vision, pages 6411–6420.

Wang, Q., Guo, J., and Kim, M.-K. (2019). An application oriented scan-to-bim framework. Remote sensing, 11(3):365.

Wu, Y., Shang, J., and Xue, F. (2021). Regard: Symmetry-based coarse registration of smartphone's colorful point clouds with cad drawings for low-cost digital twin buildings. Remote Sensing, 13(10):1882.

Xue, F., Lu, W., Chen, Z., and Webster, C. J. (2020). From lidar point cloud towards digital twin city: Clustering city objects based on gestalt principles. ISPRS Journal of Photogrammetry and Remote Sensing, 167:418–431.

Xue, F., Wu, L., and Lu, W. (2021). Semantic enrichment of building and city information models: A ten-year review. Advanced Engineering Informatics, 47:101245.

Zhao, K., Han, Q., Zhang, C.-B., Xu, J., and Cheng, M.-M. (2021). Deep hough transform for semantic line detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 44(9):4793–4806.

Zhou, Y., Qi, H., and Ma, Y. (2019). End-to-end wireframe parsing. In Proceedings of the IEEE/CVF International Conference on Computer Vision, pages 962–971.