Application of Semantic Segmentation for Enhancing Performance of Multi-View CNN-based BIM Element Classification

Youngsu Yu, youngsu@seoultech.ac.kr
Seoul National University of Science and Technology, Seoul, Republic of Korea

Koeun Lee, leekoeun@seoultech.ac.kr
Seoul National University of Science and Technology, Seoul, Republic of Korea

Daemok Ha, daemok@seoultech.ac.kr
Seoul National University of Science and Technology, Seoul, Republic of Korea

Bonsang Koo, bonsang@seoultech.ac.kr
Seoul National University of Science and Technology, Seoul, Republic of Korea

Abstract
Building Information Modeling (BIM), which is increasingly used in the construction industry, is not insusceptible to potential errors and omissions in their information. The lack of semantic integrity in BIM models hampers project stakeholders in leveraging BIM for their respective needs. To solve this problem, the authors have made efforts to rectify incomplete BIM models by evaluating the use of geometric deep learning models in automatically checking the classification of individual elements. However, the previous study using Multi-View CNN, the best performing model, had issues of misclassification when image resolutions were low. Conditional Random Fields as Recurrent Neural Networks (CRF-RNN), a deep learning algorithm for semantic segmentation, was deployed to enhance the quality of individual input images. Results of deploying the segmentation model were shown to improve MVCNN's performance by 5.97% and achieving a performance of 91.79%.

Keywords: BIM, IFC, Semantic integrity, MVCNN, CRF-RNN

1 Introduction
Building Information Modeling (BIM) is increasingly being employed as a virtual model for checking the integrity and constructability of building and infrastructure designs prior to actual construction. BIM models provide a digital repository to store and share information amongst multiple project participants (Shin et al., 2015). However, the models are not insusceptible to potential errors and omissions in their information. This may be simply due to erroneous human input, or because the information is not available at a particular project phase. Nevertheless, the lack of semantic integrity in BIM models hampers project stakeholders in leveraging BIM for their respective needs (Eastman et al., 2009; Koo et al., 2018).

Existing studies have made efforts to rectify incomplete BIM models using predefined inference rules (Belsky et al., 2016; Cursi et al., 2017; Ma et al., 2017), or more recently artificial intelligence approaches (Bloch & Sacks, 2018; Lomio et al., 2018; Kim et al., 2019; Wu & Zheng 2019). The authors had also contributed by evaluating the use of geometric deep learning models in automatically checking the classification of individual elements (Jung et al., 2019; Koo et al., 2021; Koo et al., 2021). In particular, the authors achieved high BIM elements classification accuracy by using the Multi-View CNN (MVCNN) algorithm, which uses multi-angle images for classifying 3D artifacts. However, MVCNN had issues of misclassification when image boundaries were poorly defined. In this study, Conditional Random Fields as Recurrent Neural Networks (CRF-RNN), a deep learning algorithm for semantic segmentation, was applied to the MVCNN training process to solve the low-definition problem. In detail, the semantic segmentation for improving classification accuracy in the training process was quantitatively verified by comparing the baseline MVCNN model and the MVCNN model applying the CRF-RNN algorithm to the learning process.

2 Research background

2.1 Multi-view CNN

Convolutional neural network (CNN) is a supervised deep learning model that have provided superior performance in image recognition and classification. In recent years, there is a growing trend of research that applies CNN to 3D element classification and segmentation. However, when applying CNN to classify or segment 3D objects, the objects must first be converted into a voxel or polygon mesh form for model training, which has the disadvantage of losing detailed geometric features (Maturana & Scherer, 2015; Wu et al., 2015).

Multi-View CNN (MVCNN), designed to solve this problem, classifies the shape by learning from multi-views, i.e., two-dimensional multi-images generated by photographing a three-dimensional shape from multiple angles. Specifically, MVCNN uses a total of 12 images - 10 images rotated by 30 degrees in the horizontal direction and 2 images taken vertically up and down as training data. Through this, even if there is no information about the depth of the subject compared to the representation of the 3D model, higher resolution data can be obtained with the same input size, thus exhibiting high classification performance. This approach has shown to outperform other geometric deep learning models (Su et al. 2015).

The figure presented below shows the architecture of MVCNN, which consists of a convolutional layer to extract features from individual images, a view-pooling layer to incorporate extracted features, and a full-connected layer to classify through features from integrated images, each CNN has a VGG-M network structure.

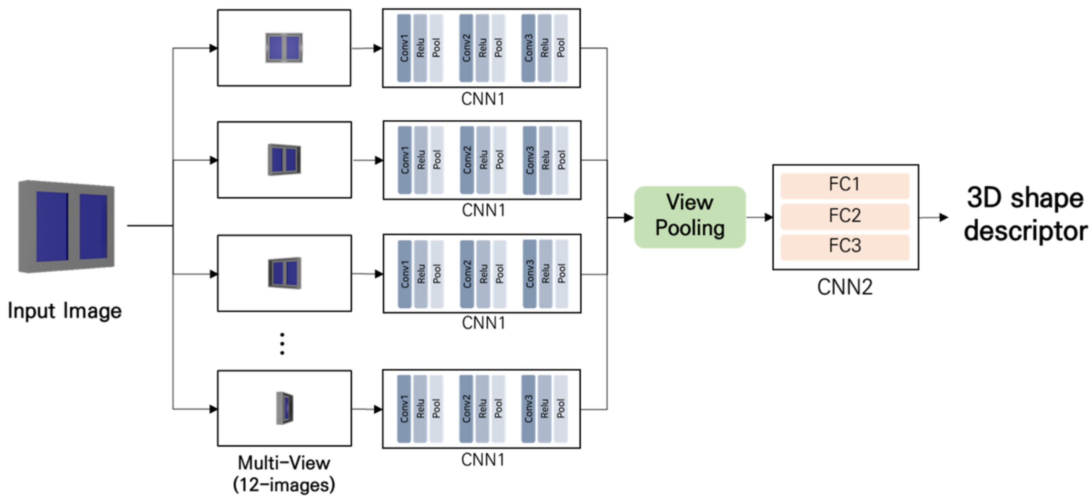


Figure 1. MVCNN structure for 3d shape recognition (Su et al 2015)

2.2 Conditional random field as recurrent neural networks

Conditional random field as recurrent neural networks (CRF-RNN) allows computers to recognize objects in images recovering the 2D outline of objects (Zheng et al. 2015).

Conditional Random Field (CRF) refers to an undirected probability graph model used for pattern recognition and structural prediction by labeling and segmenting consecutive pixels in an image. However, the conventional CRF is composed of a grid in which nodes of adjacent pixels are connected to the edge, and there is a disadvantage that detailed semantic segmentation becomes difficult as the boundary of the grid becomes smooth. CRF-RNN, designed to solve this problem, uses the weights output through the CRF as a parameter of CNN training. In this process, it utilizes a recurrent neural network (RNN) to convert the two models into one framework.

Original image (hover to highlight segmented parts) **Semantic segmentation**

Figure 2. Semantic segmentation by CRF-RNN (Zheng et al., 2015)

CRF-RNN not only showed excellent performance in the semantic segmentation field, but has also been used to improve the performance of the CNN learning model. For example, Xu et al. (2018) applied the CRF-RNN algorithm to the CNN learning model to segment the bladder boundaries of different sizes and shapes for each patient from voxel data taken by CT. As a result, a model with DSC (dice similarity coefficient) improved by 8.12% compared to V-net (baseline) was constructed.

In Figure 3 below, the MVCNN algorithm combined with the CRF-RNN constructed in this research is presented. That is, as mentioned in section 2.1, semantic segmentation was performed by applying the CRF-RNN algorithm to the individual images, which are the input data of the MVCNN algorithm. The goal was to improve the classification accuracy of the MVCNN algorithm by solving the low definition of image boundaries. Subsequently, the algorithm was built by inserting the segmented image into the convolutional neural networks of MVCNN.

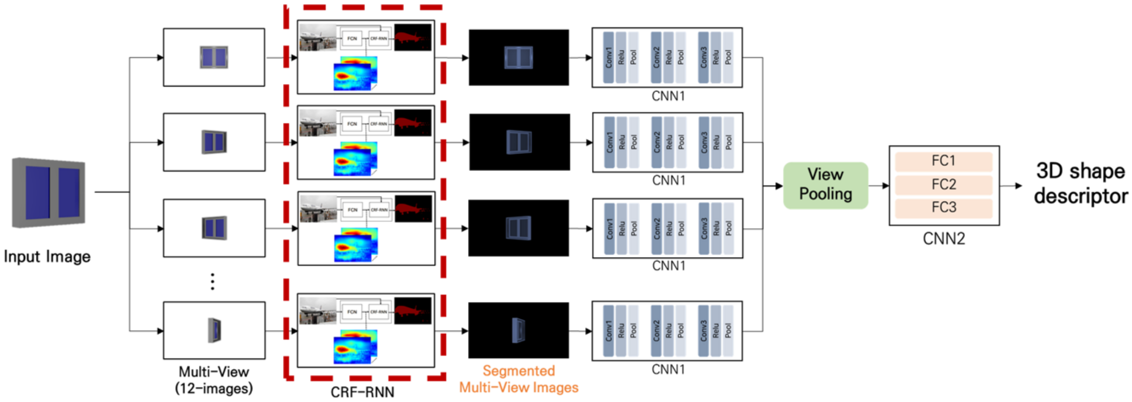


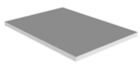
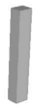
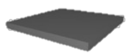
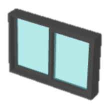
Figure 3. CRF-RNN + MVCNN structure

3 BIM elements classification based on deep learning models

3.1 Data overview

The deep learning model developed in this study employed architectural BIM elements contained in the KBIMS Library[1] provided by the Korean OpenBIM research group. There are 1,207 data composed of 13 IFC classes in the library, but classes that either did not contain enough element samples or composite elements which did not allow the capturing of individual element images were excluded. The final data set consisted of 889 sample elements of 8 IFC classes. The sample images for each IFC class is presented in Table 1.

Table 1. Sample images of BIM elements per IFC class

IFC Classes	*IfcBeam*	*IfcCeiling*	*IfcColumn*	*IfcDoor*
Image				
IFC Classes	*IfcFoundation*	*IfcSlab*	*IfcWall*	*IfcWindow*
Image				

3.2 Data preprocessing

Data preprocessing was performed for use in the MVCNN model training process. For model training, training and test data for each element were divided by a ratio of 7:3, resulting in a total of 622 data sets for training and 267 data sets for test were constructed. Subsequently, the final dataset for model training was constructed by using Blender, an open-source software, to build images taken in 12 directions for each element required for MVCNN training (Figure Figure 4).

[1] Korean Building Information Modeling Standards (KBIMS), http://www.kbims.or.kr/

Table 2. Element breakdown for deep learning model training

IFC Classes	Train set	Test set	Sum
IfcBeam	94	40	134
IfcCeiling	64	27	91
IfcColumn	72	31	103
IfcDoor	51	22	73
IfcFoundation	15	6	21
IfcSlab	155	67	222
IfcWall	130	56	186
IfcWindow	41	18	59
Total	**622**	**267**	**889**

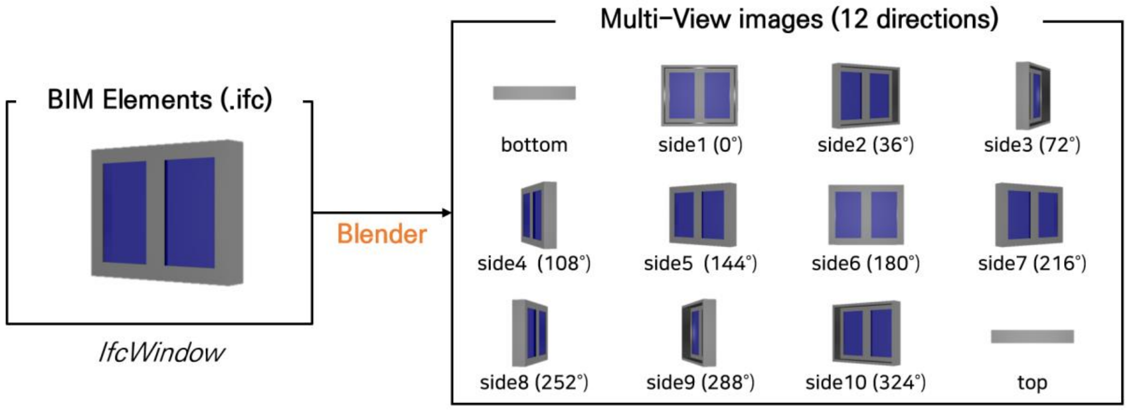


Figure 4. Sample of Multi-View Images for Model Training

3.3 MVCNN implementation

Based on the previously constructed data, the accuracy of the learning model using the MVCNN algorithm was 0.86 and the F_1 score was 0.84. Table 3 presents the classification performance for each element and Figure 5 shows the precision-recall curve for observation of changes in precision and recall values according to threshold values and changes in model performance per element. The precision-recall curve is drawn using the precision and recall values for each threshold, and curves were the Area Under the Curve (AUC) is larger depicts higher classification performance.

The overall classification performance was high, with the exception of *IfcDoor* and *IfcCeiling* classes. In particular, as a result of confirming the confusion matrix indicating the classification prediction result for each element, it was confirmed that all cases that misclassified both elements were predicted erroneously as *IfcSlab*.

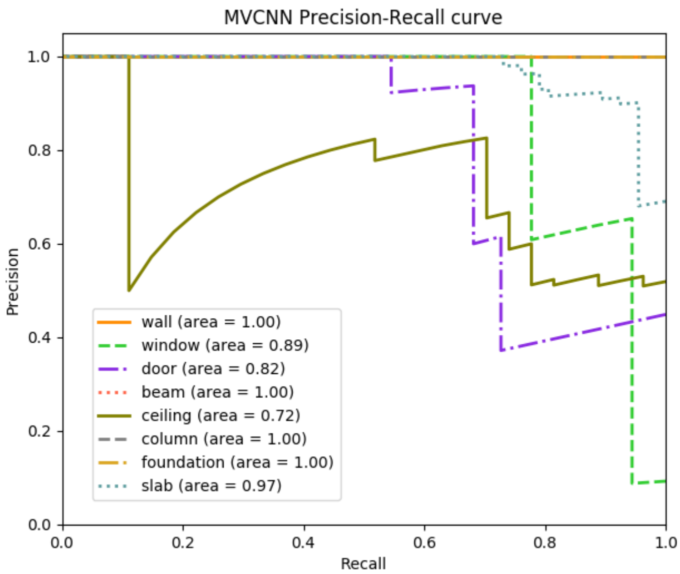


Figure 5. Precision-recall curve for MVCNN

Table 3. Accuracy, precision, recall and F_1 score per IFC class for MVCNN

	Accuracy	*Precision*	*Recall*	F_1 *score*
IfcBeam	1.00	1.00	1.00	1.00
IfcDoor	0.68	0.71	0.68	0.70
IfcCeiling	0.19	0.63	0.19	0.29
IfcColumn	0.94	1.00	0.94	0.97
IfcFoundation	1.00	1.00	1.00	1.00
IfcSlab	0.96	0.70	0.96	0.81
IfcWall	1.00	1.00	1.00	1.00
IfcWindow	0.78	0.87	0.82	0.84
Total	**0.86**	**0.87**	**0.82**	**0.84**

3.4 CRF-RNN + MVCNN implementation

As mentioned in Section 2.2, after semantic segmentation of 12 images with CRF-RNN was applied to the MVCNN algorithm, the accuracy was 0.92 and the F_1 score was 0.91. Table 4 presents the classification performance by element and Figure 6 shows the precision-recall curve for each threshold.

It was confirmed that the classification performance of individual elements was better than the previous MVCNN algorithm. In particular, it was confirmed that the classification accuracy of *IfcDoor*, which was observed to have low classification performance in the MVCNN algorithm, was significantly improved to 1.00. However, in the case of *IfcCeiling*, the classification performance was partially improved, but as a result of confirming the confusion matrix, it was verified that many elements were still misclassified by *IfcSlab*, and the classification performance was still low.

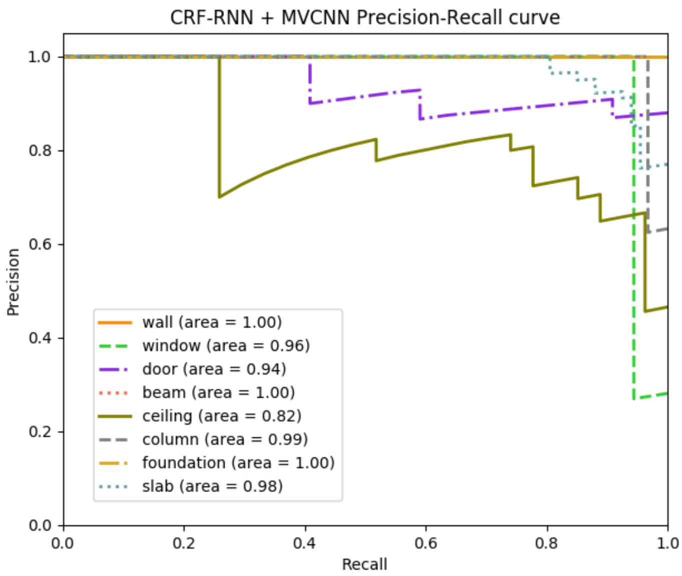


Figure 6. Precision-recall curve for CRF-RNN + MVCNN

Table 4. Accuracy, precision, recall and F_1 score per IFC class for CRF-RNN + MVCNN

	Accuracy	*Precision*	*Recall*	F_1 *score*
IfcBeam	1.00	1.00	1.00	1.00
IfcDoor	1.00	0.88	1.00	0.94
IfcCeiling	0.37	0.77	0.37	0.50
IfcColumn	0.97	1.00	0.97	0.98
IfcFoundation	1.00	1.00	1.00	1.00
IfcSlab	0.96	0.80	0.96	0.87
IfcWall	1.00	1.00	1.00	1.00
IfcWindow	0.94	1.00	0.94	0.97
Total	**0.92**	**0.93**	**0.89**	**0.91**

3.5 Results

The accuracy of the MVCNN model presented in Sections 3.3 was 0.86 and the F_1 score was 0.84, and the accuracy of the CRF-RNN + MVCNN model in Section 3.4 was 0.92 and the F_1 score was 0.91. Accordingly, it was confirmed that semantic segmentation through CRF-RNN in the MVCNN training process improved accuracy by 0.06 and F_1 score by 0.07. In particular, in Table 6, which presented the degree of improvement for each element, it was confirmed that the classification performance was improved centering on the elements of *IfcDoor*, *IfcCeiling*, and *IfcWindow*.

However, as mentioned in Section 3.4, it was observed that *IfcCeiling* was still not properly distinguished from *IfcSlab*. This is likely because the shapes of the *IfcCeiling* and *IfcSlab* in the KBIMS library are very similar, as can be seen in the image for each element presented in Table 1. In other words, it can be seen that the CRF-RNN can improve the classification performance of elements having clear geometric characteristics, but there is a limitation in improving the performance of elements that do not.

Table 5. ACC, precision, recall and F_1 score for MVCNN and CRF-RNN + MVCNN

	Accuracy	Precision	Recall	F_1 score
MVCNN	0.86	0.87	0.82	0.84
CRF-RNN + MVCNN	0.92	0.93	0.89	0.91

Table 6. Delta values between MVCNN and CRF-RNN + MVCNN

	Accuracy	Precision	Recall	F_1 score
IfcBeam	0.00	0.00	0.00	0.00
IfcCeiling	0.19	0.14	0.19	0.21
IfcColumn	0.03	0.00	0.03	0.02
IfcDoor	0.32	0.17	0.32	0.24
IfcFoundation	0.00	0.00	0.00	0.00
IfcSlab	0.00	0.10	0.00	0.07
IfcWall	0.00	0.00	0.00	0.00
IfcWindow	0.17	0.07	0.17	0.12
Total	**0.06**	**0.06**	**0.07**	**0.07**

4 Conclusion

This research proposed a deep learning-based approach to solving the semantic integrity problem that hinders the use of BIM in construction projects. CRF-RNN-based semantic segmentation was applied in the model training process to improve the 3D shape classification performance based on the MVCNN algorithm. As a result, it was confirmed that when the CRF-RNN algorithm was applied compared to the existing MVCNN, classification performance was improved by 0.06 accuracy and 0.07 F_1 score.

The classification accuracy of the *IfcDoor* and *IfcWindow* with distinct geometric shapes was greatly improved, verifying the effectiveness of semantic segmentation in the model training process. However, from the failure to solve the problem of misclassification between *IfcCeiling* and *IfcSlab*, it also demonstrated that CRF-RNN based approach has limitations when geometric shapes are similar. Therefore, in the future, the authors plan to devise a method to add relationship information between elements in the model training process to solve the problem of misclassification among elements with similar geometric shapes.

Acknowledgements

This work is supported by the Korea Agency for Infrastructure Technology Advancement (KAIA) grant funded by the Ministry of Land, Infrastructure and Transport (Grant 21AATD-C163269-01).

References

Belsky, M., Sacks, R. & Brilakis, I. (2016). Semantic enrichment for building information modeling, Computer-Aided Civil and Infrastructure Engineering, 31(4), pp. 261-274.

Bloch, T. & Sacks, R. (2018). Comparing machine learning and rule-based inferencing for semantic enrichment of BIM models. Automation in Construction. 91(22). pp. 256-272.

Cursi, S., Simeone, D. & Coraglia, U. M. (2017). An ontology-based platform for BIM semantic enrichment, Educational and research in Computer Aided Architectural Design in Europe 35, 2, pp. 649-656.

Eastman, C., Lee, J. M., Jeong, Y. S. & Lee, J. K. (2009). Automatic rule-based checking of building designs. Automation in construction. 18(8). pp. 1011-1033.

Jung, R. K., Koo, B. S. & Yu, Y. S. (2019). Using Deep Learning for automated classification of wall subtypes for semantic integrity checking of Building Information Models. Korean Institute of Building Information Modeling. 9(4), pp. 31-40.

Kim, J., Song, J. & Lee, J. K. (2019). Recognizing and Classifying Unknown Object in BIM Using 2D CNN. International Conference on Computer-Aided Architectural Design Futures. pp. 47-57.

Koo, B., Jung, R., Yu, Y. & Kim, I. (2021). A geometric deep learning approach for checking element-to-entity mappings in infrastructure building information models. Journal of Computational Design and Engineering. 8(1). pp. 239-250.

Koo, B., Jung, R. & Yu, Y. (2021). Automatic classification of wall and door BIM element subtypes using 3D geometric deep neural networks. Advanced Engineering Informatics. 47. 101200.

Lomio, F., Farinha, R., Laasonen, M. & Huttunen, H. (2018). Classification of Building Information Model (BIM) Structures with Deep Learning. 2018 7th European Workshop on Visual Information Processing (EUVIP). pp. 1-6. IEEE.

Ma, L., Sacks, R. & Kattell, U. (2017). Building model object classification for semantic enrichment using geometric features and pairwise spatial relations. Proceedings of the Joint Conference on Computing in Construction. 1. pp. 373-380.

Maturana, D. & Scherer, S. (2015, September). Voxnet: A 3d convolutional neural network for real-time object recognition. In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS). pp. 922-928. IEEE.

Shin, J., Kwon, S., Lee, K., Choi, S. & Kim, J. (2015). A Study of the Establishment of Framework for Information Exchange based on IFC Model in Domestic Collaborative Design Environment. Korean Journal of Construction Engineering and Management. 16(1). pp. 24-34.

Su, H., Maji, S., Kalogerakis, E. & Learned-Miller, E. (2015). Multi-view convolutional neural networks for 3d shape recognition. In Proceedings of the IEEE international conference on computer vision. pp. 945-953.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X. & Xiao, J. (2015). 3d shapenets: A deep representation for volumetric shapes. In Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 1912-1920.

Zheng, S., Jayasumana, S., Romera-Paredes, B., Vineet, V., Su, Z., Du, D. & Torr, P. H. (2015). Conditional random fields as recurrent neural networks. In Proceedings of the IEEE international conference on computer vision. pp. 1529-1537.