International Journal of Geo-Information

MDPI

*Article*

# Automatic Extraction of Indoor Spatial Information from Floor Plan Image: A Patch-Based Deep Learning Methodology Application on Large-Scale Complex Buildings

**Hyunjung Kim [1], Seongyong Kim [2] and Kiyun Yu [3,*]**

[1] Institute of Construction and Environmental Engineering, Seoul National University, Seoul 08826, Korea; urbanistar@snu.ac.kr
[2] Georgia Institute of Technology, School of Civil and Environmental Engineering, Atlanta, GA 30332, USA; skim3310@gatech.edu
[3] Department of Civil and Environmental Engineering, Seoul National University, Seoul 08826, Korea
* Correspondence: kiyun@snu.ac.kr

**Abstract:** Automatic floor plan analysis has gained increased attention in recent research. However, numerous studies related to this area are mainly experiments conducted with a simplified floor plan dataset with low resolution and a small housing scale due to the suitability for a data-driven model. For practical use, it is necessary to focus more on large-scale complex buildings to utilize indoor structures, such as reconstructing multi-use buildings for indoor navigation. This study aimed to build a framework using CNN (Convolution Neural Networks) for analyzing a floor plan with various scales of complex buildings. By dividing a floor plan into a set of normalized patches, the framework enables the proposed CNN model to process varied scale or high-resolution inputs, which is a barrier for existing methods. The model detected building objects per patch and assembled them into one result by multiplying the corresponding translation matrix. Finally, the detected building objects were vectorized, considering their compatibility in 3D modeling. As a result, our framework exhibited similar performance in detection rate (87.77%) and recognition accuracy (85.53%) to that of existing studies, despite the complexity of the data used. Through our study, the practical aspects of automatic floor plan analysis can be expanded.

**Keywords:** automatic floor plan analysis; indoor spatial information; IndoorGML; convolutional neural networks

check for updates

**Citation:** Kim, H.; Kim, S.; Yu, K. Automatic Extraction of Indoor Spatial Information from Floor Plan Image: A Patch-Based Deep Learning Methodology Application on Large-Scale Complex Buildings. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 828. https://doi.org/10.3390/ijgi10120828

Academic Editor: Wolfgang Kainz

Received: 6 October 2021
Accepted: 9 December 2021
Published: 10 December 2021

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Copyright:** © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

## 1. Introduction

With the recent developments in technology, including Internet of Things, location tracking and location-based services such as networks and navigation are expanding indoors. To meet the high demand for vectorized indoor spatial information, studies on automatic floor plan analysis (i.e., automatically extracting indoor spatial information from floor plan images) have been recently proposed. Floor plans are a good source of indoor spatial information because they are easy to acquire and the automatic techniques based on floor plans are relatively affordable compared to other methods such as light detection and ranging (LiDAR) or manual digitalization [1–3]. In fact, a recent study by Kim [3] demonstrated that automatic floor plan analysis technology is more effective in terms of substitutability, completeness, supply, and demand than manual digitalization. Furthermore, there is a great demand for digitalized indoor information to obtain digital twins; however, there is a considerable portion of missing digitalized blueprints (e.g., CAD, BIM) in old buildings. Therefore, automatic floorplan analysis is attracting increased attention [3].

Most existing studies on automatic floor plan analysis have focused on a simple format, which is appropriate for deep learning [4]. The floor plans mostly used in previous studies are CVC (Computer Vision Center), Rakuten, and Rent 3D, whose floor plans are

in a simplified format favorable for deep learning [3,4]. Some studies have investigated complicated floor plans, containing diverse fuzzy architectural drawings using EAIS (Electronic Architectural Information System) floor plans [2,4]; however, existing studies have mainly been conducted on house-scale buildings. Nevertheless, there is a high demand and expectation for indoor spatial information on large buildings, with a high probability and need for indoor navigation. For example, digital twins are required for smarter energy management across large-scale buildings [5]. In addition, floor plan images in practice are not usually simplified for deep learning; they are complicated architectural drawings with various features. Therefore, considering the primary and practical purposes, it is necessary to consider large-scale and complicated floor plan images to extract indoor spatial information. This study aims to develop a deep learning-based automatic floor plan analysis framework suitable for large-scale and complex floor plans.

To be specific, this study utilizes a Convolution Neural Network (CNN). A CNN model has been mostly used in recent automatic floor plan analysis because it is a data-driven model having been shown to outperform other traditional models in image segmentation tasks. Instead, a CNN model requires uniform size of images as input data, causing a difficulty for varied size or high-resolution inputs, which are common in large-scale building floorplans. As our goal was to interpret floor plans of large-scale buildings, we needed to make the input image uniform, regardless of each floor plan scale. Therefore, we proposed dividing floor plans into several patches of regular size so that the framework could use the strength of CNN models while minimizing the weakness.

## 2. Related Work

### *2.1. Methodology in Existing Studies on Floor Plan Analysis*

Floor plan analysis is a field of research that better represents indoor space by reproducing geometric, topological, and semantic information that is not present in raster drawing images [1]. The general process of floor plan analysis involves pre-processing by removing unnecessary information in raster drawings, pattern recognition of object candidate extraction based on geometric properties, and structural recognition of object candidate and reconstruction of the indoor structure [6]. Existing studies on floor plan analysis can be summarized into two approaches: (1) a rule-based process [6–8] and (2) a learning-based process [2,4,9–12].

The rule-based approach improves the performance of the floor plan analysis process by leveraging the characteristics of the drawing format. At each stage of the process, a specific drawing format-dependent rule was determined, and the information was refined based on it. Macé et al. [7] performed structural detection of a room on the BlackSet of the CVC floor plan dataset. After separating the floor plan image into character and graphic elements based on the thickness of the line, they combined the Hough transformation and image vectorization to detect the preceding objects present in the floor plan. However, their methodology is difficult to apply in other formats because the walls in BlackSet are represented by black-filled forms, which might not be the way other formats represent walls; moreover, they utilized the distinctive properties of objects such as text and furniture. Ahmed et al. [8] also used CVC BlackSet data for room detection and reconstruction. After separating the floor plan images into text and graphic elements, they reclassified the graphic elements into thick, medium, and thin lines. Thick and medium thick lines were extracted as walls, whereas thin lines were used to detect openings. Their detection rate and recognition accuracy showed better performance than the framework proposed by Macé et al. [7]; however, there are limitations in that more data-dependent methods were used on the CVC BlackSet format. Gimenez et al. [6] conducted 3D modeling of buildings using floor plan images. Their framework is also difficult to apply to other floor plans because they also apply CVC BlackSet-dependent rule-based code. In summary, a rule-based process is based on the specific characteristics of a certain data format; it is difficult to apply to other formats and cannot respond to changes in format.

To overcome the limitations of the format-dependent rule-based process, recent floor plan analysis has implemented a learning-based process. De las Heras et al. [9] conducted a learning-based process, which was shown to work through each learning in four different formats of the CVC dataset. Although, they exhibited general availability within different formats of CVC datasets, there is a limitation that many pre-processing and post-processing methods are required. Dodge et al. [10] utilized a deep learning network using the Rakuten floor plan dataset, and reconstructed walls and openings into a three-dimensional (3D) form. Liu et al. [11] also used the Rakuten dataset for automatic floor plan analysis using a deep learning network. They utilized deep learning for the extraction of intermediates for the vectorization of objects, whereas Dodge et al. utilized deep learning for the segmentation of objects. In addition, they extracted corners to reconstruct vectorized indoor models. However, this approach is applicable only to simple indoor structures with a standardized wall thickness. Zeng et al. [12] designed a deep multi-task neural network and extracted diverse floor plan elements, such as doors, windows, and different types of rooms, using the Rakuten dataset. They showed good performance in recognition; however, there are still limitations in that their dataset consists of organized and labeled data in housing scale. Various computer vision techniques have been applied based on the researcher's individual purpose to extract informative objects from a floor plan in a 2D image. Above all, the dominant approach in floor plan analysis or floor plan recognition is known as Convolutional Neural Networks [13]. A CNN model shows an outweighing performance in image segmentation tasks, enabling end-to-end implementation, and securing robust results given sufficient training data sets. Generally, in the overall framework, by directly segmenting the object of interest or extracting the feature maps, CNNs detect target building objects from a floor plan image. Then, the detected outputs will process concrete objects with semantic information to be plugged into the final indoor model.

Although a learning-based process has been introduced in automatic floor plan analysis to enhance the application to other formats, there is still a limitation in that floor plans in a learning-based approach use a simplified and standardized format favorable for learning. In this situation, in order to expand the practical aspects of automatic floor plan analysis, Jang et al. [2] and Kim et al. [4] used a complicated floor plan EAIS with fuzzy drawing and various formats using a deep learning process. However, even these studies used housing-scale floor plans. It is necessary to expand the application to larger scale and diversified formats to ensure its practical use, such as indoor navigation or building energy modeling. Therefore, in this study, a framework for automatic floor plan analysis of complex and large-scale floor plan images is developed using a patch-based deep learning framework.

### *2.2. Dataset Used in Existing Studies on Floor Plan Analysis*

The CVC floor plan dataset is an architectural drawing classified into four types according to the representation of walls [9]. BlackSet comprises 90 datasets of high-resolution drawing images of pixel size 2480 × 350. TexturedSet consists of 10 datasets with pixel sizes ranging from 1098 × 905 to 2218 × 227. Textured2Set has 18 high-resolution architectural drawings with a pixel size of 7383 × 5671 that represent walls similar to TexturedSet; however, the thickness of the lines differs according to whether they represent the inner or outer walls. Finally, ParrelSet is a house drawing with a total size of four datasets of 2500 × 3300 pixels, which represent walls with two parallel lines. The CVC datasets have both low- and high-resolution images, and the pixel area of the objects is large compared to the total pixel area in the floor plan image. There are not many different types of objects in the drawing; moreover, the number of objects in the drawing is also small. There are more types of objects in Textured2Set; however, there are only 18 datasets that are not sufficient for training for learning-based processes.

The Rakuten dataset is provided by the Rakuten Institute of Technology, a Japanese research institute; it consists of 500 real estate floor plans [14]. The floor plans are low-resolution images of approximately 1000 × 1000 pixels and are composed of objects and spaces of various colors and shades, which have the advantage of being informative. In

addition, the walls contain labeled data for deep learning network training, which many researchers have used in learning-based floor plan analysis. However, the structure of the walls is simple, and the number of rooms and objects contained is small. In addition, similar to CVC datasets, objects occupy a large proportion of the area relative to the entire space of the floor plans.

The EAIS is a floor plan dataset provided by the architectural administrative system operated by the Ministry of Land, Infrastructure, and Transport of Korea [15]. It comprises 400 low-resolution drawings mainly containing drawings of small buildings, including houses, apartments, and stores. The EAIS floor plan represents walls in various marking methods; there are several types of walls, including columns and insulation. In addition, it contains furniture information, facilities, and construction-related information in the drawing, comprising various formats of architectural drawings. However, diagonal walls are not represented in the drawing, and, like the Rakuten dataset, many of the small residential buildings have a large proportion of objects relative to the drawing area and the number of objects that comprise the indoor space.

## 3. Materials and Methods

### *3.1. Materials: Data*

The data used in this study are scanned floor plan images of Seoul National University (SNU) buildings. The dataset was acquired from the Division of Facilities Management, SNU [16]. The floor plans are diverse, complicated, and large. They have been drawn by various architectural designers/offices; the buildings were built over a long period, from the 1970s to the 2010s. In addition, the university complex contains various types of buildings; buildings with diagonal lines, curves, rectangular shapes, and pinwheels. The drawing style and symbols are varied, based on each architectural design office, and the floor plan contains numerical lines and symbols. In addition, it contains large public facilities, such as auditoriums, dormitories, libraries, and cafeteria buildings; at the same time, it includes drawings of large buildings and complex interior spaces compared to those used in previous research. In addition, it contains curved and diagonal walls. As the floor plan contains each floor of the building, there are many objects such as elevators and stairs that can represent interlayer connections. The images have a high resolution of more than 3000 pixels, and the proportion of the objects' area relative to the total area in the drawing is expressed as much smaller than the existing drawings. In addition, as of April 2020, the number of SNU members is 41,426 [17]. Library buildings have many users, including noncollege passengers. On average, four million people use the SNU library buildings in a year [18]. The buildings covered in this study are more diverse and larger in scale than those used in previous studies, i.e., housing scale buildings. The datasets used for existing floor plan analysis and our SNU dataset can be seen in Figure 1.

In this study, 230 SNU building floor plan image data were obtained, 200 of which were used for learning and 30 for testing. Additionally, to validate the capacity of the developed methodology for generalization to other datasets, we applied the University of Seoul (UOS) floor plan dataset to develop a patch-based deep learning framework. The indoor spatial information that is automatically extracted from the floor plan in this study has five classes: wall, door, window, stair room, and elevator. Walls and openings (doors and windows) are the fundamental indoor elements that can reconstruct indoor structures; hence, many previous studies used extracted walls and openings. In this study, we expanded the indoor elements to those that can connect floors, such as stair rooms and elevators, considering the shape of the completed building, rather than just a single floor. We manually annotated the floor plan dataset using LabelMe, which is a web-based annotation tool [19]. All five classes are labeled as polygon-shaped, precisely portraying each building element, even if it contains a curved boundary.

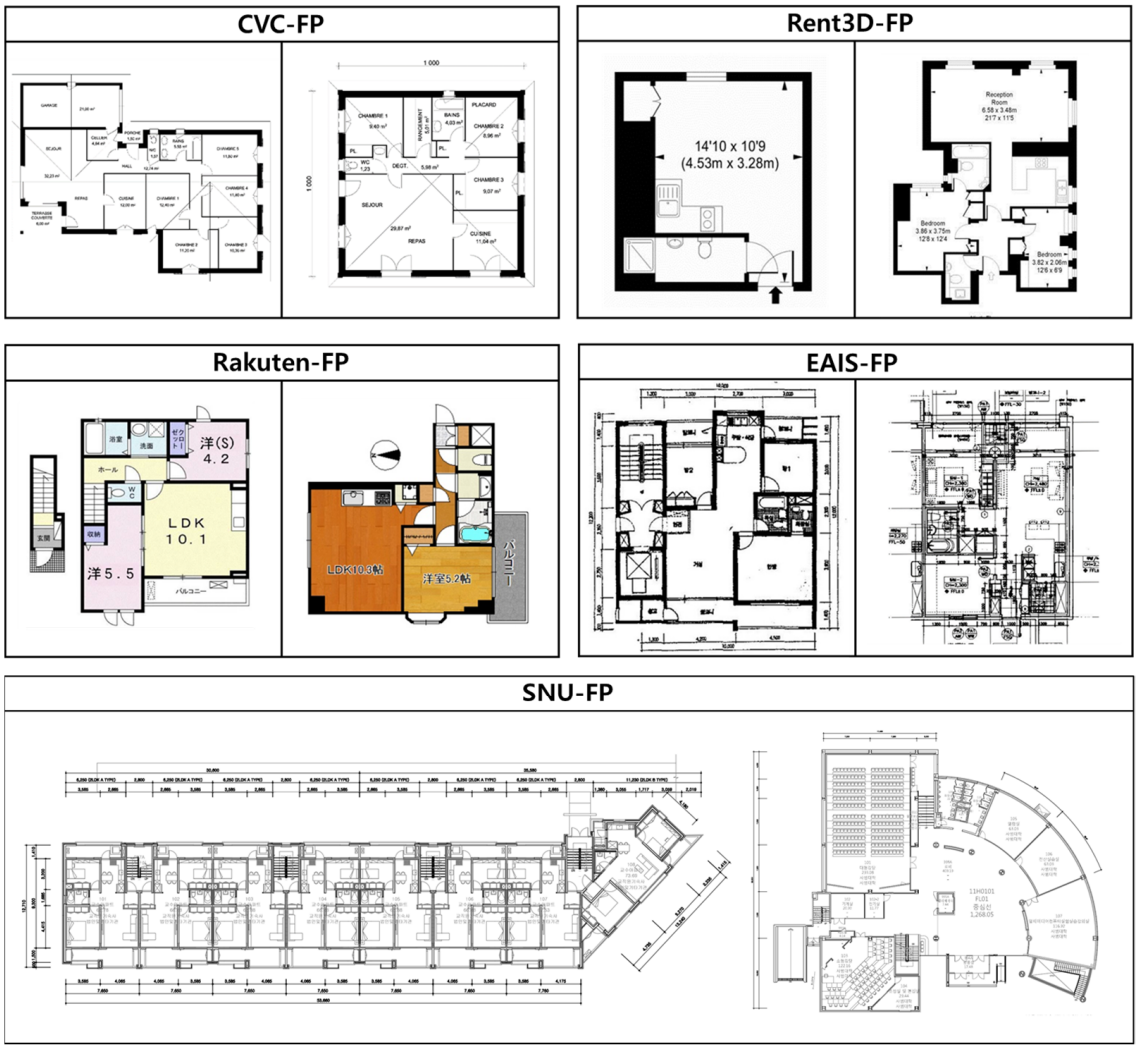



**Figure 1.** Datasets used for floor plan analysis in existing studies and ours (adapted from [3], modified by the authors).

### *3.2. Methodology*

The overall framework can be divided into three phases: (1) normalized patch extraction, (2) patch-based floor plan recognition via deep networks, and (3) generation of indoor models. Conventional studies on automatic floor plan analysis rely on handcrafted features based on geometry. This methodology is useful for the extraction of simple objects, such as walls of a certain thickness with straight lines, and does not overlap with other graphics. However, this approach is limited in that it tends to specify a particular format of floor plan that cannot be generalized to other formats. In particular, when applied to large-scale complicated floor plans, this limitation is emphasized because the divergence in drawing scale is greater for large buildings than houses.

Theoretically, data-driven models with sufficient capacity to analyze complex geometric features can overcome the aforementioned limitations. However, training a model using a floor plan dataset to automatically extract indoor spatial information (e.g., walls and openings) is fundamentally challenging because the floor plan itself makes the learning process difficult. The majority of the space of floor plans is empty, but the gathering of sparse and simple geometric features such as lines and curves involves high-level information. This is challenging because of the principles of deep-learning methodology that extract and interpret images sequentially from low-level to high-level features. In particular, the scale range of floor plans for large spaces is varied, and it is computationally burdensome because the resolution is too high to apply commonly used algorithms in image processing.

Furthermore, owing to the high complexity of the format, training and learning become more difficult. In particular, existing convolutional neural network (CNN) models are applicable to fixed square-sized images and have limitations in that they are unable to utilize high-resolution images owing to restrictions in computation. Because significant information is lost in the process of resizing a floor plan to a uniform size and an image of low resolution, deep learning models have been primarily developed for a particular type of drawing dataset with low resolution and a similar aspect ratio.

To address this limitation, we proposed a CNN-based framework designed for large-scale complex floor plans through patch-based learning, which uses varying scale floor plans divided into normalized patches (Figure 2). By applying the normalized patch-to-floor plan analysis, it is possible to manage computational burden, diverse scale issues, and poor convergence on training deep networks.

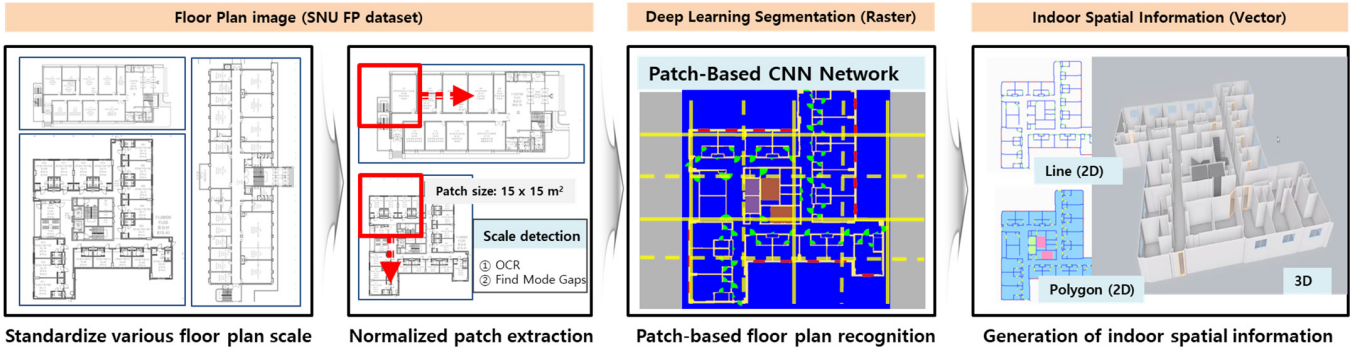



**Figure 2.** Analytical framework.

3.2.1. Normalized Patch Extraction

To efficiently utilize the CNN model, high-quality learning data (i.e., data that are repeated in uniform patterns) are required. The goal of normalized patch extraction is to unify the actual distance expressed per pixel by unifying the floor plan scale. This enables consistent input delivery to the learning-based model, even with datasets containing diverse floor plan image sizes and diverse building scales. This process has a similar effect to that of normalizing data for training.

The total number of the patches can be calculated as in Equation (1). It is affected by a patch overlapping factor and the scale of the drawings as well as the total building area and the patch size.

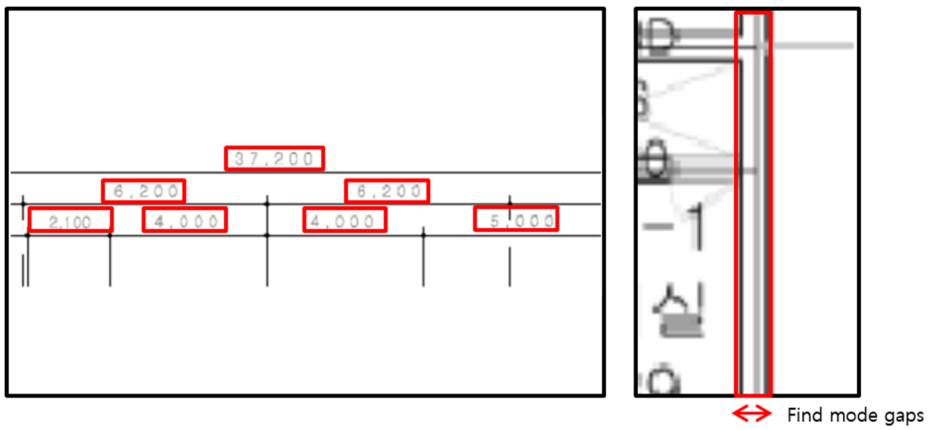
$$|P| = \left[ \frac{BuildingAres(1+\alpha)}{(\kappa \cdot Ps)^2} \right] \quad (1)$$

*where P is a set of patches, Ps is a patch size in the real world, α is an overlapping factor ranging from 0 to 1, κ is the scale of drawing.*

The first step of normalized patch extraction is to resize the image to match the scale of the floor plan and then to split it into appropriately sized patches. The appropriate size is determined to be as large as possible within the interpretable capacity, with sufficient performance for each patch with the model used for floor plan recognition. The patch size can be decided in a way of optimizing the segmentation performance. In general, a patch size shows a trade-off relationship with the segmentation performance; for example, a small patch size increases input image resolution, which is good, but decreases the field of view in the deep learning models, which results in a loss of capacity to look over neighboring objects. Considering this, we determined the appropriate patch size to the extent that the server's memory for learning permits. In this study, a patch was set to be about the size of a square office (i.e., approximately 15 × 15 $m^2$) which shows the best performance on SNU datasets and validates that this size works well in other datasets, such as UOS. Within our CPU (intel i5-4690, 8GB memory) and GPU (gtx 1080) specs, our proposed framework

takes under 5 min even for the largest building in our dataset, which requires more than 225 patches.

Regarding auto detection of scale, we used two different methods: (1) utilizing information from dimension lines in floor plans, and (2) extracting and utilizing the pixel size of the thinnest wall if there is no dimension line (Figure 3). If rich information exists in the floor plan itself, it is reasonable to use it. The floor plan is drawn proportional to the actual building, and its magnification is plotted using dimension lines. Through optical character recognition (OCR), the number of labels and corresponding pixel distance can be extracted, and, as a result, the distance per pixel (i.e., scale) can be calculated. The accuracy was improved by utilizing the median value of the scales measured in several numerical lines existing within a one-floor plan. Many floor plans have missing dimension lines for various reasons, for example, when the floor plan is not an architectural drawing (e.g., real estate floor plans and evacuation guide maps). It is important to be able to respond in this case (i.e., missing dimension lines) by generalizing our methodology. Buildings are generally geometrically structured; hence, it is not necessary to specify the thinnest walls in the floor plan. Histograms with data that count the consecutive pixel distances for each x- and y-axis were drawn; then, the pixel distance to be used as a scale criterion was specified by calculating the mode. To achieve robust results, non-maximum suppression was applied to the histogram prior to mode calculation. This process allows matching the scale on the input image of various floor plan sizes, even without dimension lines.




(**a**) (**b**)

**Figure 3.** Scale detection algorithm (**a**) utilizing dimension line and (**b**) utilizing and extracting the pixel size.

#### 3.2.2. Patch-Based Floor Plan Recognition via Deep Networks

The normalized patch derived from Section 3.2.1 is used as the input in the patch-based training of the data-driven model. Each floor plan handles a different number of patches for learning because the image pixel size of the matching floor plan is disparate. In general, the floor plan has an imbalanced distribution between object classes; a white background occupies most of the space, which adversely affects the training of the CNN model. The focal loss is used to reflect a larger value when updating the weight of the model inversely proportional to the area of each class included in the training dataset. For example, in the case of elevators which are fewer in number compared to other objects, the weight of the model changes significantly once it compensates for the small proportion. In addition, because walls directly affect the closure of indoor spaces if lost in the deep network model's output, we gave more weight to the wall than to other object classes. We set the weighted loss for the wall class to increase the accuracy of the wall, which has the greatest influence on the intact indoor model generation among various objects. In addition, to prevent redundant learning and bias in the model, certain patches that account for more than 80% of the background are excluded from learning. In the inference phase,

the normalized patches are generated from the test floor plan in the same manner, and the trained model yields the segmented patches, where each class is detected based on the pixel unit.

The segmented result of the floor plan is generated by stitching the results onto each patch, while reflecting the ensembled outputs in the overlapped areas. In Equation (2), a segmentation prediction of each patch is multiplied to a translation matrix that moves images with a relative coordinate of the corresponding patch. As a result, each predicted patch makes up the entire prediction of the floor plan. Each segmented patch from the learning is combined into one, and floor plan recognition can ultimately be completed. To mitigate discrepancies and adjust the boundary of the patches, a normalized patch is generated with more than 70% overlap. The inference outputs are ensembled and averaged when combining all the segmented patches.

$$SegPred^{*}_{fp} = argmax_{class} \sum_{i} T_{P_i} \cdot SegPred_{P_i} \quad (2)$$

*where* $T_{p_i}$ *is a tranlation matrix for ith patch.*

Deep learning is used as a data-driven model for floor plan recognition. ResNET-50 [20] was used as the skeleton network. It was modified to improve the performance of segmentation, increase the resolution of outputs, and sharpen boundaries between objects. A 512 × 512 input pixel size was also utilized instead of 224 × 224 in a direction that increases the resolution of the output. Compared to the previous model [20], the stride was lowered, and the L1 error was used instead of the L2 error to obtain a high-resolution output with clear boundaries. The architecture of the generator and its modifications from the original ResNet-50 are shown in Figure 4.

| | Type | Filters | Sizes | Output |
|---|---|---|---|---|
| | Convolutional | 64 | 7 x 7 / 2 | 256 x 256 |
| | Pooling | | 3 x 3 / 2 | 128 x 128 |
| 3 x | Convolutional | 64 | 1 x 1 | |
| | Convolutional | 64 | 3 x 3 | |
| | Convolutional | 256 | 1 x 1 | |
| | Residual | | | 128 x 128 |
| 4 x | Convolutional | 128 | 1 x 1 | |
| | Convolutional | 128 | 3 x 3 | |
| | Convolutional | 512 | 1 x 1 | |
| | Residual | | | 128 x 128 |
| 6 x | Convolutional | 256 | 1 x 1 | |
| | Convolutional | 256 | 3 x 3 | |
| | Convolutional | 1024 | 1 x 1 | |
| | Residual | | | 128 x 128 |
| 3 x | Convolutional | 512 | 1 x 1 | |
| | Convolutional | 512 | 3 x 3 | |
| | Convolutional | 2048 | 1 x 1 | |
| | Residual | | | 128 x 128 |
| | Feature maps: | | | 2048 x 128 x 128 |

**Figure 4.** CNN architecture; note that the shaded parts are modifications from the original ResNet-50 [20] to improve the resolution of the output.

### 3.2.3. Generation of Indoor Models

The result of floorplan recognition is a raster data type with pixel-wise classification. It contains only approximate geometric information, which does not contain topology and semantic information as indoor spatial information. Therefore, processing is required to convert the floorplan recognition results into indoor models. As the first step in the conversion process, we converted the extracted raster data into vectorized objects represented by relative coordinates. Raster-to-vector conversion did not require complex algorithms, because complicated formats in the floor plan had already been analyzed/simplified through the deep learning model, and only simple forms of the geometry features remained. A sim-

ple algorithm based on the Hough transformation [21] enabled us to generate vectorized objects with curves and straight lines. Objects that could be used as boundaries of rooms (e.g., walls, doors, windows) were represented by one-layer polylines, whereas objects that needed volume (e.g., stair room, elevators) were represented by polygons. For non-curved linear objects, post-processing was performed to comply with the Manhattan-rule, which averages the coordinates of the connected objects. Consequently, topology information for the indoor models was generated. In addition, a model that detects closed space (i.e., walls, windows, and doors) and creates space to separate buildings by space was developed. Stair rooms and elevators were stored as linked objects in each space, in addition to geometric information. To prevent the deep learning models from losing topology information due to missing objects, postprocessing, which connects the close gaps between walls and objects and adds virtual wall objects, was performed; the room space was subsequently divided. There was a possibility that the door may have not been detected or combined (e.g., when the door does not exist); however, the topology information retains minimum information loss because the space is divided into open boundaries (penetrating space) by virtual walls.

## 4. Results

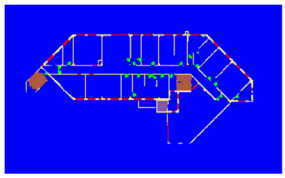
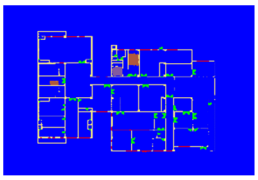
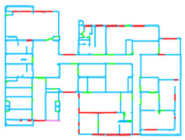
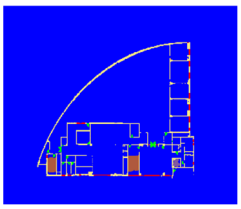
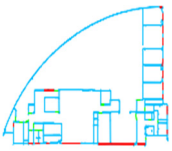
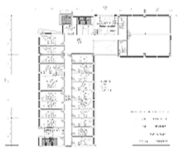
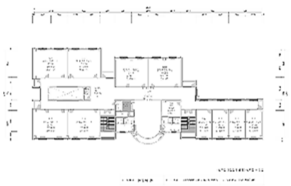
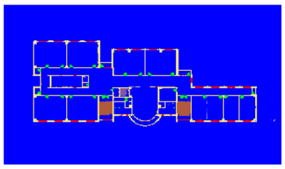
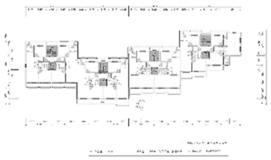
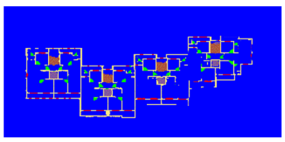
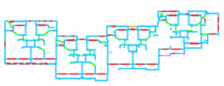
The results of the deep learning segmentation and the generated indoor model are shown in Table 1. As can be seen in the table, the input image of the SNU floor plan has diverse features. It has various scales and shapes, and some of the floor plans contain diagonal and curved walls that were not usually explored in previous studies. The result of patch-based deep learning segmentation shows clear recognition, which is sufficient for generating indoor models. Indoor models are vectorized indoor spatial information that can be used to reconstruct 2D or 3D indoor structures. The results of the indoor model (in Table 1, line vector) show that the floor plan's pattern is sufficiently implemented and completed as a supplement to the deep learning segmentation result.

The test images were newly selected in this study because no existing studies used test sets for SNU data. To train the 230 images in the datasets, we split the SNU data into 200 images for training and 30 images for testing while maintaining diverse building types. The precision and recall of the deep learning results were approximately 89% and 86%, respectively, showing a similar performance to recent related studies (see Table 2). Notably, the performance for the window was relatively low because wall and window objects in the SNU dataset are hardly distinguishable and missing windows are usually detected as a wall, which means it does not significantly affect the room division when generating the indoor model. However, stairs show an irregular and unrepeating pattern, thus yielding a low recall value in pixel-based evaluation. For automatic floor plan analysis, good performance in segmentation tasks cannot guarantee good performance on vector (indoor model) results [4]. The deep learning segmentation result is a mediated product for indoor models [4]; therefore, it is necessary to evaluate the indoor model itself.

Generally, the evaluation of indoor models for automatic floor plan analysis is conducted through either a wall segmentation task or a room-detection task [22]. Because our goal was to reconstruct indoor structures, we selected the room-detection task as an evaluation process. The evaluation protocol for the room-detection task resulted in a detection rate (DR) and recognition accuracy (RA) based on the match score table [23], which was characterized by reflecting the exact one-to-one matches as well as the partial one-to-many and many-to-one matches based on vector evaluation. In particular, we utilized the same metric for evaluating the floor plan analysis from previous studies [4,9]. When calculating the match-scores of the predicted room and ground truth, the acceptance and rejection thresholds used to determine whether each pair matched were 0.5 and 0.05, respectively. In addition, an evaluation of the walls was replaced with an evaluation of the closed space implemented through the walls (i.e., room) because the evaluation of a single wall is less relevant to the reproduction performance of the entire structure. Table 3 shows an evaluation of the room-detection task for our proposed method using the SNU_FP data. The proposed model resulted in an 89% detection rate and 86% recognition accuracy, and overall, it exhibited a slightly higher performance except for the door object compared to

the pixel-based assessment shown in Table 2. This is because our framework complements the deep learning results while generating the final indoor model. For example, the stair room shows a relatively low score on the pixel-based evaluation, but a higher score on the vector evaluation. This is due to the model successfully identifying most of the stairs, despite failing to detect a precise boundary.

**Table 1.** Results of the proposed model on SNU dataset. From left to right, (**a**) input floor plan, (**b**) segmented results, and (**c**) generated indoor model in a vector format.

| (a) SNU Floor Plan | (b) Floor Plan Segmentation | (c) Indoor Model |
|---|---|---|
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

The main purpose of extracting indoor spatial information in vector form from floor plan images is to reconstruct indoor building structures. To show the application of our result in a compatible way, we converted our indoor models into Java OpenStreetMap Editor plugins [24], and IndoorGML (OGC standard for an open data model and XML schema for indoor spatial information) [25], which have the potential to be compatible with 3D modeling programs such as Sketchup, BIM, and Open Street Map (see Figure 5). In addition, it can be integrated adaptively with other standards such as CityGML [26], making the 3D indoor models compatibly expanding to the surrounded outdoor city models, beyond the building level.

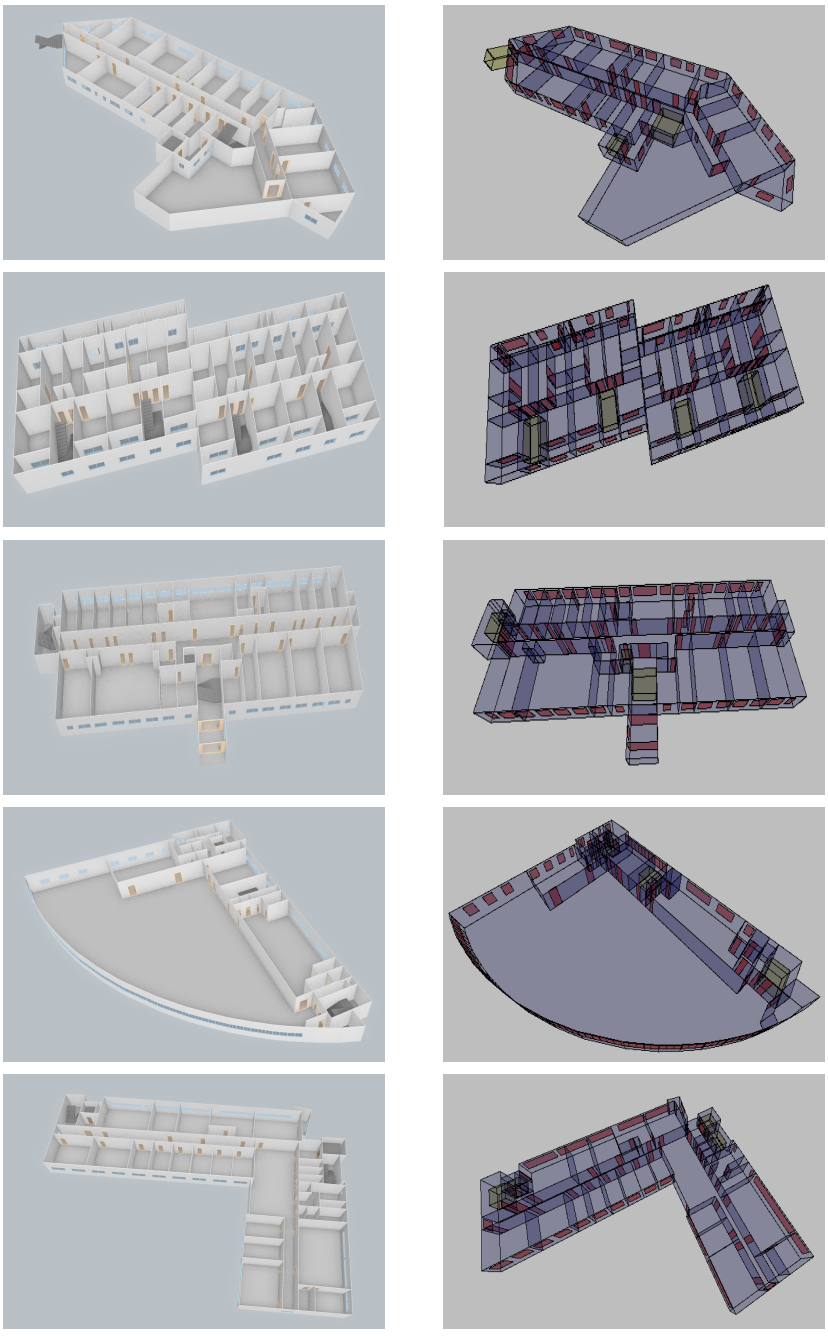
**Table 2.** Deep learning result.

| Metrics | Wall | Window | Door | Stair Room | Elevator | Mean |
|---|---|---|---|---|---|---|
| Precision (%) | 82.38 | 77.46 | 89.47 | 94.19 | 88.01 | **88.51** |
| Recall (%) | 87.19 | 73.01 | 87.95 | 78.23 | 87.51 | **85.56** |

**Table 3.** Indoor model result.

| Metrics | Room [1] | Window | Door | Stair Room | Elevator | Mean |
|---|---|---|---|---|---|---|
| DR (%) | 81.80 | 83.63 | 75.73 | 97.67 | 97.67 | **87.77** |
| RA (%) | 85.20 | 79.90 | 79.77 | 91.20 | 91.20 | **85.52** |

[1] The evaluation for the wall was replaced with an evaluation of the room.



(**a**) (**b**)

**Figure 5.** Example of 3D indoor models based on (**a**) JOSM, and (**b**) IndoorGML.

The 3D indoor models can be used for a variety of purposes, such as the creation of digital twins for smart cities or the development of real estate information services. Our findings can also be combined with existing researches regarding the indoor navigation supporting system in the land administration domain model [27] or the localization and positioning aspect considering partitioning in indoor spaces [28]. In particular, since our framework enabled automatic extract of indoor spatial information in vector format on a large-scale complex building, the application aspect of indoor navigation would be more emphasized in cooperation with such researches such as human navigation usage in 3D indoor construction [29,30], the indoor navigation system for emergency evacuation [31–33], and so on.

## 5. Discussion

### *5.1. Discussion on the Performance with Existing Studies*

This paper proposes a framework that overcomes the limitations of existing studies on automatic floor plan analysis, which considers relatively simple drawings favorable to deep learning. Digitalization of indoor spaces is required in larger buildings that many people navigate, and the actual floor plans in the real world are not always organized or simplified for learning. As the purpose of this study was to reconstruct indoor spatial information from a floorplan of the large-scale buildings, thereby expanding the previous approach mostly focusing on a unit of the house in complex space or buildings with high utilization of indoor spatial information. To achieve this purpose, compared to the previous work using representative public datasets, such as CVC [6–9], Rakuten [10–12], and EAIS [2,4], our target floor plan was aligned with large-scale buildings and the proposed method was specified to the floor plan images which are difficult to process at once due to their huge size. In addition, to extract essential topology information for use as indoor spatial information, our target of interests expanded to topology objects such as a stair, an elevator, beyond opening information of the room.

Since we explored the SNU floor plan, which covers various types of complex drawing styles as well as large-scale buildings that have never been attempted in existing studies, a direct performance comparison with existing studies cannot be obtained. The existing methods based on learning-based models, whether they distributed their trained models or not, are not able to be applied to a large-scale floor plan as they do not have enough capacity to handle high-resolution and large-scale drawings or are too specific to certain formats of the floor plan dataset. In addition, there is no shared annotation for large-scale complex floor plan datasets, which are fundamental barriers for the comparison of learning-based approaches. In this situation, capacity for a large-scale drawing is our main contribution, but at the same time, forces us to evaluate only our own dataset and to perform an indirect comparison. Nevertheless, to verify our performance with existing studies, we refer to the vectorized indoor model on EAIS and used it as baseline performance because this dataset is the most complicated format in existing studies. Evaluation of the EAIS dataset is based on a room-detection task, including openings (window, door); moreover, the model does not extract the stair room and elevator. On comparison with the vectorized indoor model on EAIS data [4] in Table 4, our model shows similar performance on DR (Ours: 87.77, EAIS: 87.87), and slightly lower performance on RA (Ours: 85.52, EAIS: 89.96). Although our dataset includes various and larger floor plan image sizes, the results show that our framework has a similar performance to existing studies.

### *5.2. Discussion on Generalization Capability with Other Datasets*

To verify the applicability of our proposed framework to other formats, we tested another complex and large-scale dataset. We applied the UOS dataset to our trained network, trained solely using the SNU dataset, without training it with the UOS dataset. The UOS dataset is a large and complex drawing showing the exterior of the building, including curved walls, and various symbols that are not present in the SNU dataset. Figure 6 shows the results of applying our framework to the UOS dataset. As can be seen

in Figure 6, our framework worked well with the UOS dataset, and could generate both 2D line and polygon indoor models as well as the 3D JOSM indoor model and indoor GML. We manually added some virtual walls before converting our results into 3D models. Because of the format discrepancy between datasets, our trained model experienced difficulty in detecting openings and certain divisions of the rooms, but overall, it still resulted in intact indoor models despite some missing elements.

**Table 4.** Performance comparison of indoor models.

| Dataset | Metrics | Room | Window | Door | Stair Room | Elevator | Mean |
|---|---|---|---|---|---|---|---|
| SNU | DR (%) | 81.80 | 83.63 | 75.73 | 97.67 | 97.67 | **87.77** |
| | RA (%) | 85.20 | 79.90 | 79.77 | 91.20 | 91.20 | **85.52** |
| EAIS [4] | DR (%) | 87.87 | - | - | - | - | **87.87** |
| | RA (%) | 89.96 | - | - | - | - | **89.96** |

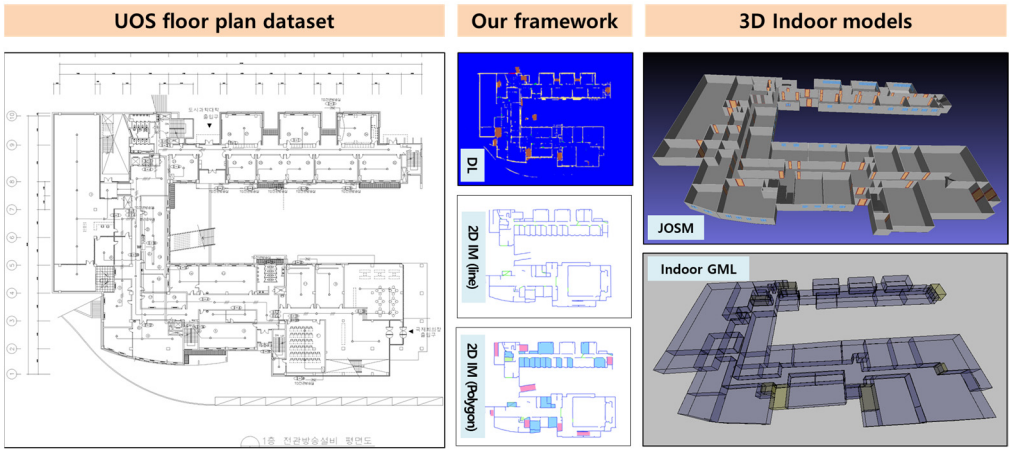



**Figure 6.** Application of our framework to the UOS dataset.

## 6. Conclusions

The demand for indoor spatial information is increasing, and automatic floor plan analysis is gaining more attention as an affordable means of acquiring indoor spatial information [2–4]. In this context, this study presents a patch-based deep learning network and a framework for reconstructing indoor space for more complex and large-scale buildings as compared to previous studies. We utilized a CNN to overcome its limitation on interpreting varied size or high-resolution inputs, commonly found in floor plans of large-scale buildings. As input data, SNU dataset (200 for learning, 30 for testing) was used, which contains various types of data drawn by various architectural offices from the 1970s to the 2010s, containing large-scale buildings with diagonal lines, curves, rectangular shapes, and pinwheels. The floor plan images of the SNU dataset have a high resolution of more than 3000 pixels. To unify the actual distance expressed per pixel by unifying the floor plan scale, we normalized the patch consistent input delivery to the learning-based model, even with datasets containing diverse sizes and scales. The segmented result of the floor plan was generated by stitching the results on each patch, while reflecting the ensembled outputs in the overlapped areas. After raster to vector conversion, the indoor model of walls, windows, rooms, stair rooms, and elevators was generated. The performance showed detection rate (87.77%) and recognition accuracy (85.53%), similar to that of existing studies that used a relatively unified and organized format with a regular scale.

The main implications of our work can be summarized into three aspects. First, this study enabled the automatic extraction of indoor elements from complicated and variously scaled floor plan images with high performance. The fundamental purpose of reconstructing indoor structures is to interpret or navigate large-scale complex buildings

rather than simple housing scales. Second, this study extracted indoor elements not only for reconstructing the interior space geometry, but also for connectivity with other floors. Unlike existing studies, this study extracted stair rooms and elevators to connect with other stories. This can facilitate automatic floor plan analysis to reconstruct the indoor space in a floor as well as a whole building. Third, this study enabled the reconstruction of indoor space and converted it to a standard format that can be utilized for other purposes. Based on the results of our generated indoor models, it is possible to construct 3D-based indoor models such as JOSM-based indoor models or IndoorGML. These are open-source-based indoor formats; hence, it is possible to utilize the results depending on the subjects.

This study has limitations in that it can only train the SNU dataset as a sample for complex and large-scale buildings due to the difficulty in securing data. However, if large-scale and navigable buildings such as multiplex shopping malls can be included, our proposed framework can be trained into a more reliable model. In future research, we plan to further extract and assemble the topology information of the buildings, given that our proposed framework includes an expandable indoor model. Nevertheless, this study is significant in that it expands the practical aspect of automatic floor plan analysis as it covers large-scale floor plans that have been excluded from previous research. This study enables the recognition of floor plan datasets for large-scale buildings in complex and diverse formats, which extends the application of automatic floor plan analysis technology.

**Author Contributions:** Conceptualization, Hyunjung Kim; methodology, Hyunjung Kim and Seongyong Kim; software, Hyunjung Kim and Seongyong Kim; validation, Hyunjung Kim, Seongyong Kim, and Kiyun Yu; formal analysis, Hyunjung Kim and Seongyong Kim; investigation, Hyunjung Kim and Seongyong Kim; resources, Kiyun Yu; data curation, Hyunjung Kim and Seongyong Kim; writing—original draft preparation, Hyunjung Kim; writing—review and editing, Hyunjung Kim, Seongyong Kim, and Kiyun Yu; visualization, Hyunjung Kim and Seongyong Kim; supervision, Kiyun Yu; project administration, Kiyun Yu; funding acquisition, Kiyun Yu. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The SNU floor plan can be acquired through an official request to the Division of Facilities Management, Bureau of Construction and Facilities Management at Seoul National University (https://en.snu.ac.kr/about/overview/organization/administrative_organizations, accessed on 18 August 2021). Some or all data: models, or codes that support the findings of this study are available from the corresponding author upon reasonable request (list items); (1) our trained models and results, (2) SNU floor plan dataset with our annotations.

**Acknowledgments:** This research was supported by a grant (21NSIP-B135746-05) from the National Spatial Information Research Program (NSIP) funded by the Ministry of Land, Infrastructure and Transport of the Korean government.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

1. Gimenez, L.; Hippolyte, J.L.; Robert, S.; Suard, F.; Zreik, K. Review: Reconstruction of 3D building information models from 2D scanned plans. *J. Build. Eng.* **2015**, *2*, 24–35. [CrossRef]
2. Jang, H.; Yu, K.; Yang, J.H. Indoor reconstruction from floorplan images with a deep learning approach. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 65. [CrossRef]
3. Kim, H. Evaluation of Deep Learning-Based Automatic Floor Plan Analysis Technology: An AHP-Based Assessment. *Appl. Sci.* **2021**, *11*, 4727. [CrossRef]
4. Kim, S.; Park, S.; Kim, H.; Yu, K. Deep Floor Plan Analysis for Complicated Drawings Based on Style Transfer. *J. Comput. Civ. Eng.* **2021**, *35*, 04020066. [CrossRef]
5. Francisco, A.; Mohammadi, N.; Taylor, J.E. Smart City Digital Twin–Enabled Energy Management: Toward Real-Time Urban Building Energy Benchmarking. *J. Manag. Eng.* **2020**, *36*, 4019045. [CrossRef]
6. Gimenez, L.; Robert, S.; Suard, F.; Zreik, K. Automatic reconstruction of 3D building models from scanned 2D floor plans. *Autom. Constr.* **2016**, *63*, 48–56. [CrossRef]
7. Macé, S.; Valveny, E.; Locteau, H.; Tabbone, S. A system to detect rooms in architectural floor plan images. *ACM Int. Conf. Proceeding Ser.* **2010**, 167–174. [CrossRef]

8. Ahmed, S.; Liwicki, M.; Weber, M.; Dengel, A. Improved automatic analysis of architectural floor plans. *Proc. Int. Conf. Doc. Anal. Recognit.* **2011**, 864–869. [CrossRef]
9. De las Heras, L.P.; Ahmed, S.; Liwicki, M.; Valveny, E.; Sánchez, G. Statistical segmentation and structural recognition for floor plan interpretation: Notation invariant structural element recognition. *Int. J. Doc. Anal. Recognit.* **2014**, *17*, 221–237. [CrossRef]
10. Dodge, S.; Xu, J.; Stenger, B. Parsing floor plan images. In Proceedings of the 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA), Nagoya, Japan, 8–12 May 2017; pp. 358–361. [CrossRef]
11. Liu, C.; Wu, J.; Kohli, P.; Furukawa, Y. Raster-to-Vector: Revisiting Floorplan Transformation. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2214–2222. [CrossRef]
12. Zeng, Z.; Li, X.; Yu, Y.K.; Fu, C.W. Deep floor plan recognition using a multi-task network with room-boundary-guided attention. *arXiv*. 2019. Available online: https://arxiv.org/abs/1908.11025 (accessed on 12 July 2021).
13. Guo, Y.; Liu, Y.; Georgiou, T.; Lew, M.S. A review of semantic segmentation using deep neural networks. *Int. J. Multimed. Inf. Retr.* **2018**, *7*, 87–93. [CrossRef]
14. Rakuten Institute of Technology. Available online: https://rit.rakuten.co.jp/data_release/ (accessed on 12 July 2021).
15. Open EAIS. Available online: https://open.eais.go.kr/main/main.do (accessed on 12 July 2021).
16. Division of Facilities Management, Bureau of Construction and Facilities Management in Seoul National University. Available online: https://en.snu.ac.kr/about/overview/organization/administrative_organizations (accessed on 12 July 2021).
17. Statistical Yearbook of Seoul National University. Available online: https://www.snu.ac.kr/about/downloads?md=v&bbsidx=130691 (accessed on 12 July 2021).
18. Bulletin of the Seoul National University Library no.142. 2020. Available online: https://hdl.handle.net/10371/164939 (accessed on 12 July 2021).
19. Russell, B.C.; Torralba, A.; Murphy, K.P.; Freeman, W.T. LabelMe: A database and web-based tool for image annotation. *Int. J. Comput. Vis.* **2008**, *77*, 157–173. [CrossRef]
20. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2016; pp. 770–778.
21. Duda, R.O.; Hart, P.E. Use of the Hough transformation to detect lines and curves in pictures. *Commun. ACM* **1972**, *15*, 11–15. [CrossRef]
22. De las Heras, L.P.; Terrades, O.R.; Robles, S.; Sánchez, G. CVC-FP and SGT: A new database for structural floor plan analysis and its groundtruthing tool. *Int. J. Doc. Anal. Recognit.* **2015**, *18*, 15–30. [CrossRef]
23. Phillips, I.T.; Chhabra, A.K. Empirical performance evaluation of graphics recognition systems. *IEEE Trans. Pattern Anal. Mach. Intell.* **1999**, *21*, 849–870. [CrossRef]
24. JOSM. Available online: https://josm.openstreetmap.de/ (accessed on 12 May 2021).
25. Indoor GML. Available online: http://www.indoorgml.net/ (accessed on 12 May 2021).
26. Kang, H.-K.; Li, K.-J. A Standard Indoor Spatial Data Model—OGC IndoorGML and Implementation Approaches. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 116. [CrossRef]
27. Alattas, A.; Zlatanova, S.; Van Oosterom, P.; Chatzinikolaou, E.; Lemmen, C.; Li, K.-J. Supporting Indoor Navigation Using Access Rights to Spaces Based on Combined Use of IndoorGML and LADM Models. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 384. [CrossRef]
28. Kim, S.-H.; Li, K.-J.; Cho, H.-G. A Flexible Framework for Covering and Partitioning Problems in Indoor Spaces. *ISPRS Int. J. Geo-Inf.* **2020**, *9*, 618. [CrossRef]
29. Keil, J.; Korte, A.; Ratmer, A.; Edler, D.; Dickmann, F. Augmented Reality (AR) and Spatial Cognition: Effects of Holographic Grids on Distance Estimation and Location Memory in a 3D Indoor Scenario. *PFG-J. Photogramm. Remote Sens. Geoinf. Sci.* **2020**, *88*, 165–172. [CrossRef]
30. Zhang, J.; Xia, X.; Liu, R.; Li, N. Enhancing human indoor cognitive map development and wayfinding performance with immersive augmented reality-based navigation systems. *Adv. Eng. Inform.* **2021**, *50*, 101432. [CrossRef]
31. Chen, L.-W.; Liu, J.-X. Time-efficient indoor navigation and evacuation with fastest path planning based on Internet of Things technologies. *IEEE Trans. Syst. Man, Cybern. Syst.* **2019**, *51*, 3125–3135. [CrossRef]
32. Fu, M.; Liu, R. Automatic Generation of Path Networks for Evacuation Using Building Information Modeling. In *Computing in Civil Engineering 2019: Visualization, Information Modeling, and Simulation*; American Society of Civil Engineers: Reston, VA, USA, 2019; pp. 320–327.
33. Niu, L.; Wang, Z.; Song, Y.; Li, Y. An Evaluation Model for Analyzing Robustness and Spatial Closeness of 3D Indoor Evacuation Networks. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 331. [CrossRef]