Automation in Construction 166 (2024) 105649

Contents lists available at ScienceDirect

Automation in Construction

journal homepage: www.elsevier.com/locate/autcon

Semantic floorplan segmentation using self-constructing graph networks

Julius Knechtel [a,*], Peter Rottmann [a], Jan-Henrik Haunert [a], Youness Dehbi [b]

[a] University of Bonn, Institute for Geodesy and Geoinformation, Bonn, Germany
[b] HafenCity University Hamburg, Computational Methods Lab, Hamburg, Germany

ARTICLE INFO

Keywords:
Floorplan
Semantic segmentation
Graph Convolutional Network
Convolutional neural network
Self-constructing graph

ABSTRACT

This article presents an approach for the automatic semantic segmentation of floorplan images, predicting room boundaries (walls, doors, windows) and semantic labels of room types. A multi-task network was designed to represent and learn inherent dependencies by combining a Convolutional Neural Network to generate suitable features with a Graph Convolutional Network (GCN) to capture long-range dependencies. In particular, a Self-Constructing Graph module is applied to automatically induce an input graph for the GCN. Experiments on different datasets demonstrate the superiority and effectiveness of the multi-task network compared to state-of-the-art methods. The accurate results not only allow for subsequent vectorization of the existing floorplans but also for automatic inference of layout graphs including connectivity and adjacency relations. The latter could serve as basis to automatically sample layout graphs for architectural planning and design, predict missing links for unobserved parts for as-built building models and learn important latent topological and architectonic patterns.

1. Introduction

Information about the layout of existing apartments is required for various applications, e.g. (1) for the generation of as-built Building Information Models (BIMs) or (2) for the computer-aided generation of new residential building layouts in the architectural design process. The necessary floorplans exist for most buildings and contain valuable information for these areas of application. Nevertheless, especially since a great share of existing buildings stems from the pre-digital era, the according floorplans are often only available in raster format. Hence, a vectorization is mandatory to obtain the underlying information in a more usable machine-readable format. As a result, it is possible to subsequently generate an as-built BIM, which contains valuable information about the general layout including shape and location parameters of both walls and windows of the building. This represents a convenient basis for augmenting a model with existing infrastructures, such as installed electric networks, as, for instance, presented by Dehbi et al. [1]. This workflow is illustrated in Fig. 1. Given a raster image, an automatic semantic segmentation of the floorplan as well as an automatic induction of graphs representing the underlying dependencies is performed (red arrows). The corresponding results represent a basis for subsequent applications, e.g. the architectural design process or the generation of as-built BIMs (teal arrows). This approach allows to avoid the overhead of indoor surveying using traditional methods such as laser scanning. A similar pipeline to convert a floorplan to a 3D scene has been presented by Vidanapathirana et al. [2], however, they only focus on the second part of the pipeline, i.e. vectorizing the input layout building, and, hence, used manually labeled ground truth data as input. Nevertheless, for a complete application a semantic segmentation of the floorplan image is necessary, and the quality of the segmentation highly influences the quality of the subsequent steps. The main contribution of our article lies in establishing this missing link and filling the gap towards a complete vectorization pipeline based on an accurately segmented scene.

The automatic interpretation of such plans is an intense field of research. Initial approaches addressing the semantic segmentation of floorplan images were primarily based on handcrafted features and heuristics (e.g. [3–5]). Due to lacking stability and generality under different and various circumstances, deep learning approaches are increasingly used to handle the diversity and complexity of floorplans (e.g. [6,7]). Nevertheless, these approaches show deficiencies in detecting specific room classes, e.g. balconies, which are particularly challenging due to their high variety in shape and geometry. This is attributed to their different styles often characterized by very thin boundaries on the one hand or to the challenging environments with different wall thicknesses and irregular and round wall shapes on the other hand. Additionally, further disturbing symbols, e.g. a compass, are falsely classified as walls which has been identified as one main

[*] Corresponding author.
E-mail addresses: knechtel@igg.uni-bonn.de (J. Knechtel), rottmann@igg.uni-bonn.de (P. Rottmann), haunert@igg.uni-bonn.de (J.-H. Haunert), youness.dehbi@hcu-hamburg.de (Y. Dehbi).

https://doi.org/10.1016/j.autcon.2024.105649
Received 14 October 2022; Received in revised form 22 July 2024; Accepted 22 July 2024
Available online 9 August 2024

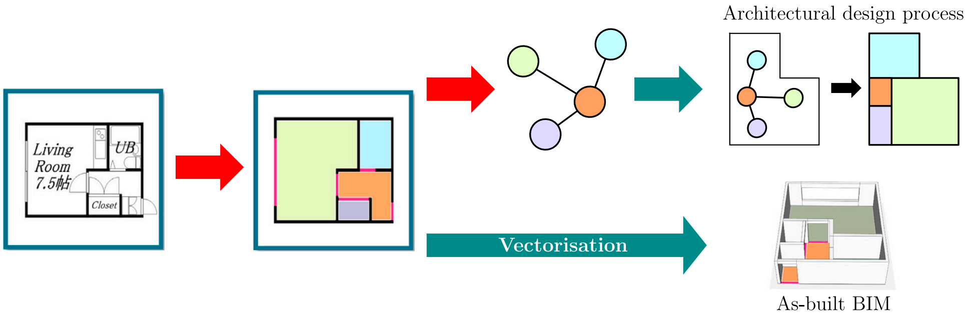


Fig. 1. Semantic floorplan segmentation as prior for both (a) architectural design and (b) generation of as-built BIMs.

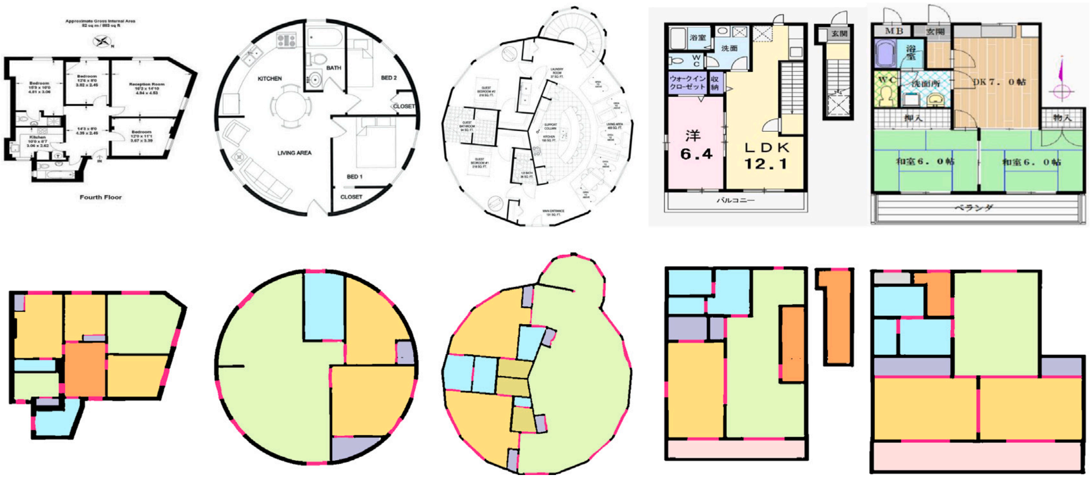


Fig. 2. Different floorplans (top) and the corresponding segmentation results from our network (bottom).

problem by Zeng et al. [6]. Although different types of networks were previously investigated, the efficient incorporation of long-range relations beyond the local dependencies, i.e. neighboring pixels, to extend the receptive field is not yet addressed in a satisfactory manner. This is, however, of high interest in the context of the built environment characterized by not only locally repetitive structures but also globally propagated patterns. Hence, incorporating these information in a convenient way is a promising approach to enhance the result of the semantic segmentation of floorplan images. To this aim, this article presents an approach that overcomes the deficiencies encountered in previous approaches by designing a multi-task network combining a Convolutional Neural Network (CNN) with a Graph Convolutional Network (GCN). The latter can succinctly represent latent relations between not necessarily neighboring objects and, hence, encode long-range dependencies. Instead of relying on manually engineered and built graphs for the GCN module, we establish the missing link to a pre-attached CNN module by automatically inducing the required input graph for the GCN. This has been performed following the spirit of Liu et al. [8], who introduced a Self-Constructing Graph (SCG) Convolutional Network for the semantic labeling of remotely sensed urban scenes.

In our approach, the incorporation of the SCG module for tackling one of the segmentation tasks turns out to be a key step leading to a successful semantic interpretation of floorplans superior to the state-of-the-art methods. Some exemplary floorplans interpreted by our method are shown in Fig. 2. The semantic segmentation of walls and windows overcomes deficiencies from preceding approaches even in challenging arrangements. In particular, the network is capable of handling round and irregular shapes, walls of different thickness, floorplans with multiple stories, balconies and other symbols like compasses.

To successfully handle the geometry as well as the semantics of an underlying floorplan we designed a multi-task network solving the problem at hand in a two-branched fashion as shown in Fig. 3: Two parallel tracks are followed to infer both the boundaries and interior structures. The latter are semantically interpreted as room types, e.g. living room or bedroom. The room boundaries are further refined differentiating between openings (doors and windows) and the enclosing walls.

The second aforementioned possible application, i.e. the generation of building layouts, requires additional processing of the semantically segmented images. It often relies on a bubble diagram, which builds upon a graph structure: The nodes represent rooms and the edges the connections between them. Based on such diagrams and in combination with a probabilistic modeling utilizing a Bayesian Network, it is possible to perform a computer-generated design process [9]. Other approaches use Mixed Integer Quadratic Programming (MIQP, Wu et al. [10]) or Deep Learning [11] for the generation of building layouts. Learning-based approaches benefit from a larger database and an exhaustive background knowledge, thus, an automatic extraction of information from raster images is desirable to support the learning process. Hence, beyond the geometric and semantic aspects, we address the topological relations between the underlying interpreted structures by automatically retrieving such layout graphs.

The remainder of this paper is structured as follows: Section 2 presents an overview of the related work. Section 3 gives insights into our proposed method where the designed network architecture is introduced as well as the loss function and the training process are described. Section 4 discusses the performed experiments and the achieved results, in addition the results of ablation studies are presented. In Section 5 we introduce one of the possible applications of the segmented floorplans, namely building a knowledge base for the automatic design of new buildings by inducing connectivity and neighborhood graphs. Additionally, a deeper discussion of the results,

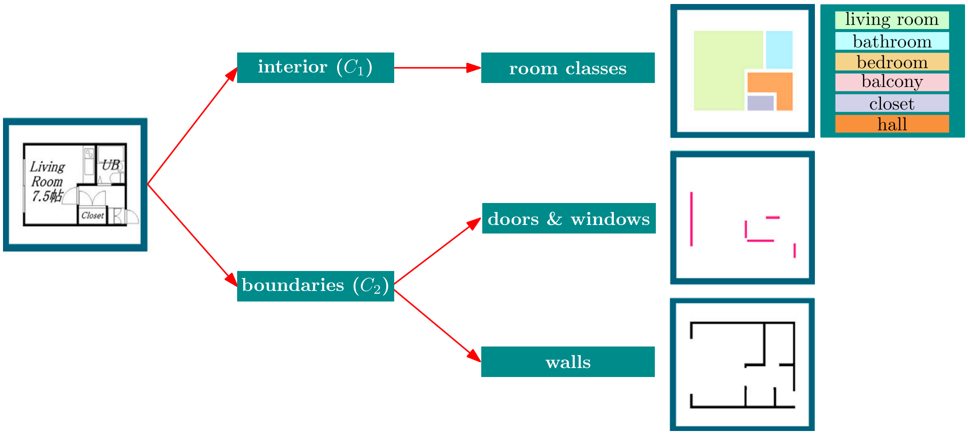


Fig. 3. Overview on our two-branched approach for the semantic segmentation of floorplans.

limitations and the influence of the quality of the annotated ground truth data is provided. We summarize the paper and give an outlook to future research in Section 6.

2. Related work

Architectural floorplans play a prominent role in several fields such as real estate or furniture and interior space design. The semantic interpretation of such drawings recognized a rapid and important evolution. The early methods tried to perform a bottom-up semantic segmentation of walls and openings by detecting low-level features and geometric primitives such as lines in the underlying floorplans [3,12–14]. Another method by Ryall et al. [15], for instance, aimed for identifying partially or fully bounded regions based on their centers and an according pixel assignment using a proximity metric in a semi-automatic fashion. Most of the mentioned approaches are heuristically guided for the identification of the underlying low-level features and primitives, which often leads to inaccurate results. Machine learning opened up new opportunities to renounce heuristics and automatically learn biased geometries [4,16,17].

In order to tackle the high diversity and variety of floorplans, the semantic segmentation took profit from the increasingly emerging deep representations. In this context, several deep learning methods, e.g. Convolutional Neural Networks (CNNs, e.g. [6,18–21]), Generative Adversarial Networks (GANs, e.g. [7,22,23]) or Fully Convolutional Networks (FCNs, e.g. [24–26]) have been applied to avoid heuristic-based assumptions. Recently, networks utilizing attention modules played an increasingly prominent role [27–29]. In this context, Yang et al. [30] applied attention in combination with Graph Neural Networks for floorplan segmentation. Wen et al. [31] proposed a multimodal segmentation network (OCR) to additionally extract texts and added a subsequent vectorization step. Some approaches focus on using deep learning methods for specific tasks in the context of floorplans, e.g. to detect and recognize text [32] or to semantically segment the walls in historical floorplans of Versailles [33]. For more deep learning methods in the field of automatic floorplan analysis, the interested reader is referred to Kim [34] or Pizarro et al. [35], who provide an overview and evaluation of existing approaches.

As yet, Deep FP published by Zeng et al. [6], DLAK-GAN introduced by Zhang et al. [7] and Offset-GA proposed by Wang and Sun [27] test their approaches extensively and represent the state-of-the-art methods in the field of semantic segmentation of floorplans. Hence, we compared the results of our experiments with these approaches. Zeng et al. [6] introduced in Deep FP a room-boundary guided attention module within a multi-task CNN to improve the performance of the semantic segmentation task by capturing and exploiting spatial information between the walls and the according room types. Zhang et al. [7] combined GANs and direction-aware kernels into DLAK-GAN to achieve better results. Wang and Sun [27] proposed a new Offset-Guided Attention mechanism aiming to improve the semantic consistency within the rooms. All of these approaches demonstrated their outcomes based on the Rent3d (R3D) [36] and the R2V [18] datasets. We used the same datasets to ensure comparability. The corresponding benchmark introduced by Liu et al. [18] demonstrated the feasibility of combining CNNs for the identification of floorplans' junctions and Integer Programming to vectorize the input drawings. This method is, however, not able to deal with non-rectilinear structures.

Our approach draws upon the ideas of self-constructing graphs (SCG) which have been introduced by Liu et al. [8] for the semantic labeling of urban scenes based on remote sensing data. This method has been expanded by the same authors leveraging multiple views to exploit rotational invariances in airborne images in an explicit way [37]. In the context of SCG, Zi et al. [38] incorporated an attention-mechanism to acquire possible correlations between different channels in remote sensing images (SGA-Net). However, to the best of our knowledge, this is the first demonstration of the impact of adapting these recent self-constructing graphs for the semantic interpretation of architectural floorplans. Our method differs from existing approaches as we propose a novel architecture for a multi-task network combining the strengths of a CNN on the one hand and graph networks using SCG and GCN on the other hand, which allows for outperforming the state-of-the-art methods.

The generation of floorplans can be performed with multiple different methods, e.g. applying Mixed-Quadratic Linear Programming (MIQP, Wu et al. [10]) or again deep learning [11]. Luo and Huang [39] use an adversarial generative framework to generate floorplans while utilizing a self-attention mechanism for explicitly capturing interrelations of rooms. The importance of layout graphs for the generation of building layouts is evident and the results of our semantic segmentation can be used for retrieving them. To generate layout graphs, Lu et al. [40] applied a combination of semantic neural networks with a post-processing algorithm for room segmentation to infer neighborhood information. In this context, Moradi et al. [41] propose an algorithmic approach to extract adjacency matrices from floorplans. CB-SAGE, a network proposed by Verma and Jadeja [42], performs node classification on layout graphs, significantly outperforming previous methods in this domain.

3. Methodology

3.1. Network architecture

Our designed multi-task network consists of two decoder modules, called WallNet and RoomNet, which share one encoder, as depicted in Fig. 4. To automatically extract features from an input

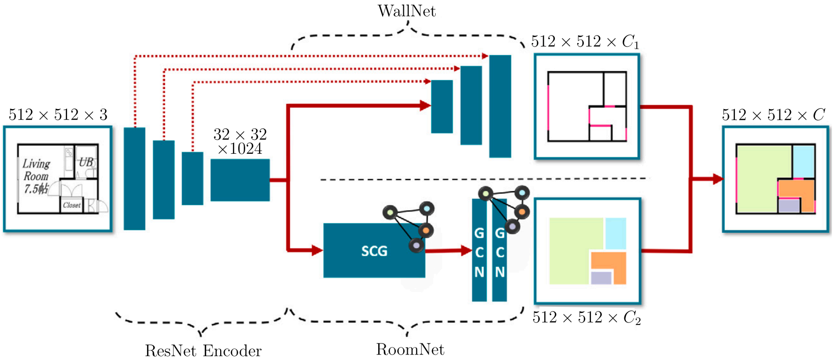


Fig. 4. Architecture of the presented network: First the input image is passed into a ResNet encoder. Subsequently, the WallNet produces the output for detecting walls and openings, the RoomNet, consisting of the Self-Constructing Graph (SCG) module and a Graph Convolutional Network (GCN), predicts the room classes. The dotted red lines indicate skip connections.

floorplan, the encoder is following a ResNet architecture as introduced by He et al. [43], more precisely a ResNet-101. Each decoder addresses a different semantic segmentation task: The RoomNet is designed to predict pixels $c \in C_1$ from different room types $C_1 = \{living\ room, bathroom, bedroom, balcony, closet, hall, background\}$, whereas the WallNet is dedicated to identify pixels $c \in C_2$ with $C_2 = \{wall, opening, background\}$ (cf. Fig. 3). We opted for a two-branched architecture combining the strengths of CNN and GCN, which are elaborated in the following. A proof of the superiority of this design choice over a single-branch method is provided in Section 4.3.

The WallNet module builds upon a UNet architecture [44], where the decoder is characterized by a mirrored structure and skip connections (cf. Fig. 4 dotted red lines). We opted for a CNN as these networks have proven to yield good results in such tasks. The mirroring allows for a robust prediction of the pixels in the border of the underlying image, whereas the skip connections tackle the well-known degradation problem [43]. Here, we applied ReLU as activation function and used batch normalization.

The RoomNet on the other hand consists of two consecutive modules in the context of Graph Neural Networks. The first one is the Self-Constructing Graph module [8,37]. It allows for an automatic learning of latent graph structures directly from the floorplan's 2D feature map $X \in \mathbb{R}^{n \times d}$ without relying on an a-priori customized graph representation. This has been generated by the ResNet encoder in the first step. Thus, $n = h \times w$ corresponds to the size of the encoder result and d represents the number of the induced features. The resulting graph $G = (A, X)$ serves as input which the subsequent Graph Convolutional Network [45] requires. Herewith, A represents an adjacency matrix. To avoid over-smoothing, the GCN module is composed of only two layers. The layers learn new embeddings for the n nodes by performing convolutions on the underlying graph structure. In the first layer, a batch normalization is applied and ReLU is used as activation function. The second layer results in as many features as room types C_2. After a subsequent unpooling step, the prediction for each pixel from the input image is performed (cf. Fig. 4). The strength of GCNs can be described as the ability to capture long-range dependencies in an image. Obviously, this could also be performed by a large receptive field using, e.g., (1) deeper network architectures or (2) a bigger kernel size. This, however, often leads to overfitted models for (1) or a higher number of parameters for (2) leading to a higher computational complexity. In general, many approaches such as global pooling could be applied and are also worth of investigation. We opted, however, for a graph-based approach in order to confirm the intuition suggesting the suitability of such frameworks for the semantic segmentation of floorplans.

The SCG module itself consists of an encoder (ENC) and a corresponding decoder (DEC) as depicted in Fig. 5, which provides a closer look into the SCG's architecture. Based on the feature map $X \in \mathbb{R}^{h \times w \times d}$, the encoder module ENC computes a new embedding Z based on Gaussian parameters (μ, σ) calculated by two distinct convolutional layers respectively. Assuming a centered isotropic Gaussian prior distribution over the parameters, the latent variables are regularized taking the Kullback–Leibler divergence between this distribution and the embedding Z as loss function $\mathcal{L}_{\mathrm{kl}}$ which has to be minimized. In the subsequent decoder module DEC, the aforementioned weighted and undirected adjacency matrix A is computed based on an inner product between the acquired embedding Z. Basically, the decoder performs a similarity check between the feature vector of each node in the 2D map. In this manner, the learned weights in A_{ij} between two nodes v_i and v_j represent a prior for their similarity and, hence, a connection in the resulting graph allowing bilateral information sharing (cf. Fig. 5). To ensure a sound self-similarity, a diagonal logarithmic regularization is defined as a further loss function $\mathcal{L}_{\mathrm{dl}}$. All in all, the loss function for the SCG part is defined as:

$$\mathcal{L}_{\mathrm{SCG}} \leftarrow \mathcal{L}_{\mathrm{kl}} + \mathcal{L}_{\mathrm{dl}} \tag{1}$$

For more details on the SCG module, the interested reader is referred to Liu et al. [8].

3.2. Loss functions

In order to learn a semantic segmentation in a supervised manner, datasets with annotated floorplans as ground truth are required. Our method has been evaluated based on two existing and widely used benchmarks. The first one is the R2V dataset containing 815 images which have been labeled [18] and are originally stemming from the large-scale LIFULL HOME's dataset[1] from Japan. The second one is the R3D dataset composed of 214 floorplan images from London, published by Liu et al. [36], augmented by additional 18 images located in New York [6]. While R2V contains mainly rectangular floorplans, R3D includes also non-linear, round shaped apartments and is characterized by walls with a high variety of thicknesses. For comparability reasons, we followed the same splitting ratios into train and test data as performed by Zeng et al. [6] for Deep FP. For R2V, 715 images have been used for training and the remaining 100 images for testing. Concerning the R3D dataset, 179 floorplans have been used for training and 53 serve as testing data.

In general, floorplans are characterized by highly imbalanced classes in terms of both their frequency and sizes, i.e. number of pixels belonging to them. This legitimates the intuition behind using the Adaptive Class Weighting Loss (ACW, Liu et al. [8]) for both aforementioned modules, i.e. WallNet and RoomNet. ACW has been designed to deal with such circumstances and has proven to be superior to other loss

[1] https://www.nii.ac.jp/dsc/idr/en/lifull/.

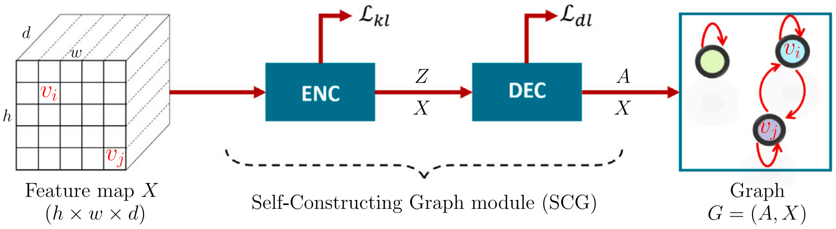


Fig. 5. Schematic overview of the Self-Constructing Graph module, adapted from [37]. Given the feature map X, an Encoder (ENC) and Decoder (DEC) are used to infer a graph structure.

functions in domains with less frequent classes [8]. The ACW loss is defined as

$$\mathcal{L}_{\mathrm{ACW}} = \frac{1}{|Y|} \sum_{i \in Y} \sum_{j \in C} \left(\tilde{w}_{ij} \cdot p_{ij} - \log_e(\mathrm{mean}\{d_j | j \in C\}) \right) \tag{2}$$

for each pixel $i \in Y$ and each individual class $j \in C$. Herewith, $\tilde{w}_{ij}$ represents a normalized weight which takes imbalances between individual classes $j \in C$ for each batch and in each training step $t \in \{1, 2, \dots, t_{max}\}$ into consideration:

$$\tilde{w}_{ij} = \frac{w_j^t}{\sum_{j \in C} w_j^t} \cdot (1 + y_{ij} + \tilde{y}_{ij}), \tag{3}$$

based on the prediction $\tilde{y}_{ij} \in (0, 1)$ for a class j at a pixel $i \in Y$ and the corresponding ground truth $y_{ij} \in \{0, 1\}$. Beforehand, the weights w_j^t can be calculated as iterative median frequency:

$$w_j^t = \frac{\mathrm{median}(\{f_j^t | j \in C\})}{f_j^t + \epsilon}, \tag{4}$$

where $\hat{f}_j^t$ is the pixel frequency of the class j including all preceding training steps:

$$f_j^t = \frac{\hat{f}_j^t + (t-1) \cdot f_j^{t-1}}{t} \tag{5}$$

To take also negative examples into account, a *Positive and Negative Class Balanced Function* (PNC, Liu et al. [8]), incorporating the squared error $e_{ij} = (y_{ij} - \tilde{y}_{ij})^2$ between prediction and ground truth is involved in $\mathcal{L}_{\mathrm{ACW}}$ as follows:

$$p_{ij} = e_{ij} - \log_e \left(\frac{1 - e_{ij}}{1 + e_{ij}} \right) \tag{6}$$

Further, the *dice*-coefficient d_j as presented by Milletari et al. [46] is considered for the $\mathcal{L}_{\mathrm{ACW}}$ as well:

$$d_j = \frac{2 \cdot \sum_{i \in Y} (y_{ij} \cdot \tilde{y}_{ij})}{\sum_{i \in Y} y_{ij} + \sum_{i \in Y} \tilde{y}_{ij}} \tag{7}$$

More details on the adaptive class weighting loss can be found in the publication by Liu et al. [8].

For the WallNet, the loss $\mathcal{L}_{\mathrm{WallNet}}$ is reduced to calculating the $\mathcal{L}_{\mathrm{ACW}}$ incorporating the three target classes C_1, i.e. walls, openings and background, and, therefore:

$$\mathcal{L}_{\mathrm{WallNet}} = \mathcal{L}_{\mathrm{ACW}} \tag{8}$$

For the corresponding loss of the RoomNet, however, the loss functions of the SCG module (cf. Section 3.1) are additionally taken into consideration for the underlying classes C_2 accordingly:

$$\mathcal{L}_{\mathrm{RoomNet}} = \mathcal{L}_{\mathrm{SCG}} + \mathcal{L}_{\mathrm{ACW}} \tag{9}$$

Hence, the total loss for the multi-task network is defined as a linear combination of the aforementioned losses:

$$\mathcal{L}_{\mathrm{total}} = \alpha \cdot \mathcal{L}_{\mathrm{RoomNet}} + (1 - \alpha) \cdot \mathcal{L}_{\mathrm{WallNet}} \tag{10}$$

4. Experimental evaluation

4.1. Settings and parameters

The presented network was trained on an Nvidia RTX A5000 with 24 GB memory. We used the ADAM optimizer [47] with a learning rate of 5×10^{-4}, a weight decay amounting 2×10^{-5} and a batch size of 8 for training and testing. Moreover, we applied ReLU as activation function and used batch normalization. In our experiments, a value of 0.5 for the coefficient α from Eq. (10) turns out to deliver the best results which corresponds to an equal weighting. The input images have a resolution of 512 × 512 which is retained for our experiments, allowing us to capture information on thin lines possibly representing walls or wall openings. To accelerate the training process, we used as mentioned a pretrained ResNet-101 as encoder. A total number of 30k training steps is conducted while evaluating every three epochs in order to maintain and report the best results. For the comparison with the state-of-the-art methods based on the two benchmark datasets presented in Section 3.2, we report the class-wise results from the respective publication.

4.2. Qualitative and quantitative evaluation

Our quantitative evaluation builds upon three standard and commonly used metrics: The overall accuracy

$$\mathrm{overall\_accuracy} = \frac{\sum_{j \in C} N_j}{\sum_{j \in C} \hat{N}_j}, \tag{11}$$

the per-class accuracy

$$\mathrm{per\_class\_accuracy}(j) = \frac{N_j}{\hat{N}_j}, \tag{12}$$

and the mean intersection over union (IoU, cf. Eq. (13)). N_j represents the number of correctly predicted pixels for the class $j \in C$, whereas $\hat{N}_j$ denotes the total number of pixels in the ground truth. While the per-class accuracy reflects a good class-wise assessment, the overall accuracy as global metric tends to hide worse results in classes with a lower number of pixels, which motivates using the mean intersection over union (IoU) as further global metric for quality assessment. The mean IoU is calculated over all classes incorporating true positives (TP), true negatives (TN) and false negatives (FN):

$$\mathrm{IoU} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP} + \mathrm{FN}} \tag{13}$$

We extensively compare our approach with the three most successful methods to date for the semantic segmentation of floorplans, namely Deep FP, DLAK-GAN and OffsetGA. We also investigate the feasibility of our method compared to a general semantic segmentation network, DeepLab v3 [48], which deals with image segmentation in a more general context and without explicitly addressing floorplans.

Fig. 6 presents a qualitative analysis of exemplary floorplans – limited to the ones published in [7] – and their segmentation results acquired from the different approaches based on the R3D dataset. Since for OffsetGA [27] the authors showed different qualitative results they are not included in this comparison. The results of each method are

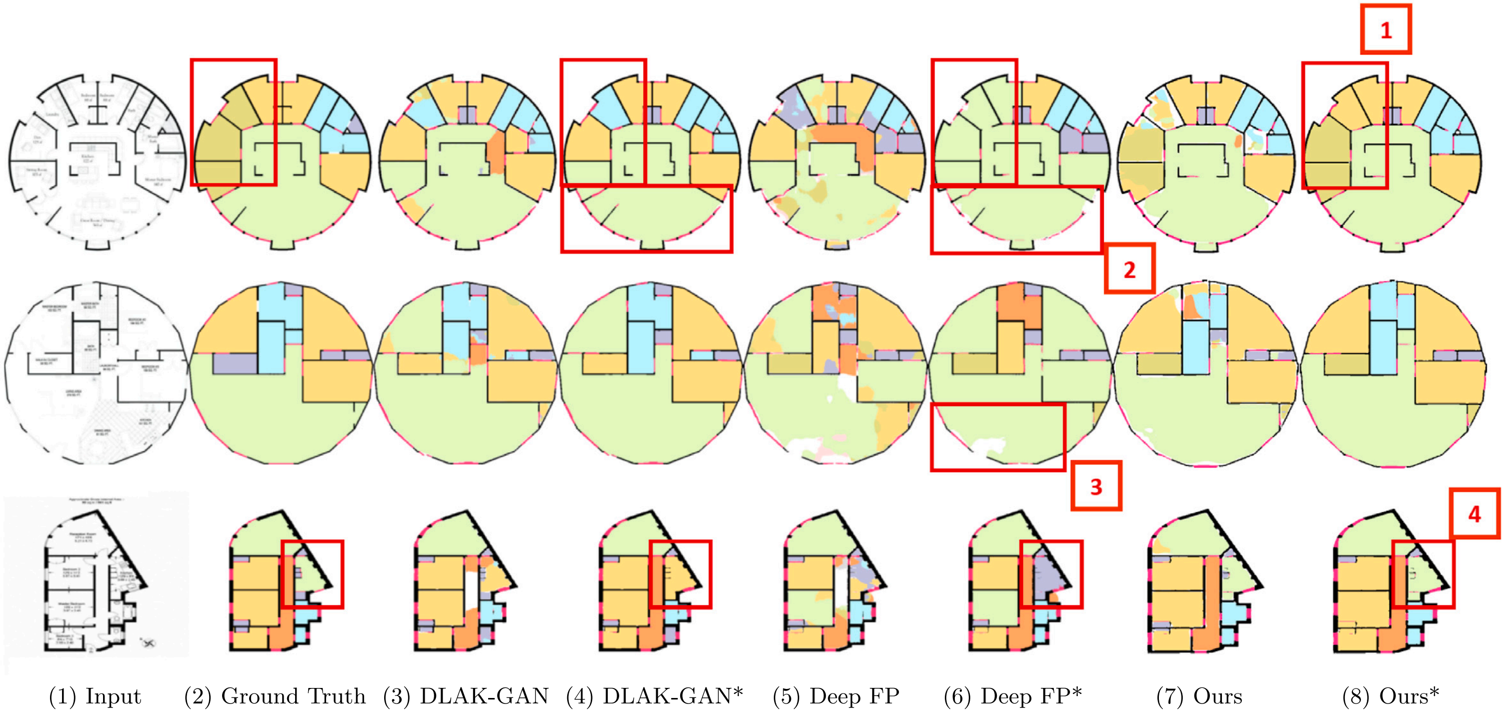


Fig. 6. Qualitative evaluation of the resulting semantic segmentation from our approach (7 & 8) and the state-of-the-art methods DLAK-GAN (3 & 4, Zhang et al. [7]) and Deep FP (5 & 6, Zeng et al. [6]) based on the R3D dataset [36]. The original RGB image and the corresponding ground truth are shown in the two first columns (1 & 2). The * denotes a postprocessing step as first used for Deep FP by Zeng et al. [6] for the according method.

Table 1

Overview of the results achieved by different methods without and with applying the postprocessing method from [6]: (1) DeepLab v3 [48], (2) Deep FP [6], (3) DLAK GAN [7] and our approach. (4) OffsetGA [27] reported only values without postprocessing.

(a) Results for the R2V dataset.

		Without postprocessing					With postprocessing			
		(1)	(2)	(3)	(4)	Ours	(1)	(2)	(3)	Ours
Mean IoU		0.69	0.74	0.81	0.77	**0.85**	0.67	0.76	0.81	**0.84**
Per-class accuracy	Wall	0.80	0.89	0.84	**0.90**	**0.90**	0.80	0.89	0.84	**0.90**
	Door/Window	0.72	**0.89**	0.83	**0.89**	0.87	0.72	**0.89**	0.83	0.87
	Closet	0.78	0.81	0.84	**0.88**	**0.88**	0.85	0.92	0.91	**0.95**
	Bathroom	0.90	0.87	0.88	0.92	**0.93**	0.90	0.93	0.94	**0.98**
	Living room	0.85	0.88	0.86	**0.94**	0.90	0.84	**0.91**	0.88	**0.91**
	Bedroom	0.82	0.83	0.91	0.96	**0.97**	0.65	0.91	0.96	**0.98**
	Hall	0.55	0.68	0.87	0.84	**0.90**	0.87	0.84	**0.96**	**0.96**
	Balcony	0.87	0.90	0.90	0.91	**0.95**	0.45	0.92	0.95	**0.99**
	Background	–	–	–	–	0.95	–	–	–	0.92
Overall accuracy		0.88	0.89	0.91	0.93	**0.94**	0.87	0.90	0.92	**0.94**

(b) Results for the R3D dataset.

		Without postprocessing					With postprocessing			
		(1)	(2)	(3)	(4)	Ours	(1)	(2)	(3)	Ours
Mean IoU		0.50	0.63	0.74	0.74	**0.78**	0.44	0.66	0.75	**0.80**
Per-class accuracy	Wall	0.93	**0.98**	0.95	0.95	0.97	0.93	**0.98**	0.93	0.97
	Door/Window	0.60	0.83	0.82	0.83	**0.88**	0.60	0.83	0.82	**0.88**
	Closet	0.24	0.61	0.65	**0.71**	0.68	0.05	0.54	0.57	**0.67**
	Bathroom	0.76	0.81	**0.93**	0.90	**0.93**	0.57	0.78	0.87	**0.94**
	Living room	0.76	0.87	0.91	**0.93**	**0.93**	0.90	0.93	**0.98**	0.97
	Bedroom	0.56	0.75	0.86	**0.89**	0.87	0.40	0.79	0.86	**0.87**
	Hall	0.72	0.59	0.80	**0.87**	0.82	0.44	0.68	0.87	**0.88**
	Balcony	0.08	0.44	0.76	0.83	**0.87**	0.00	0.49	0.69	**0.81**
	Background	–	–	–	–	0.98	–	–	–	0.98
Overall accuracy		0.85	0.89	**0.94**	0.91	**0.94**	0.83	0.90	0.94	**0.95**

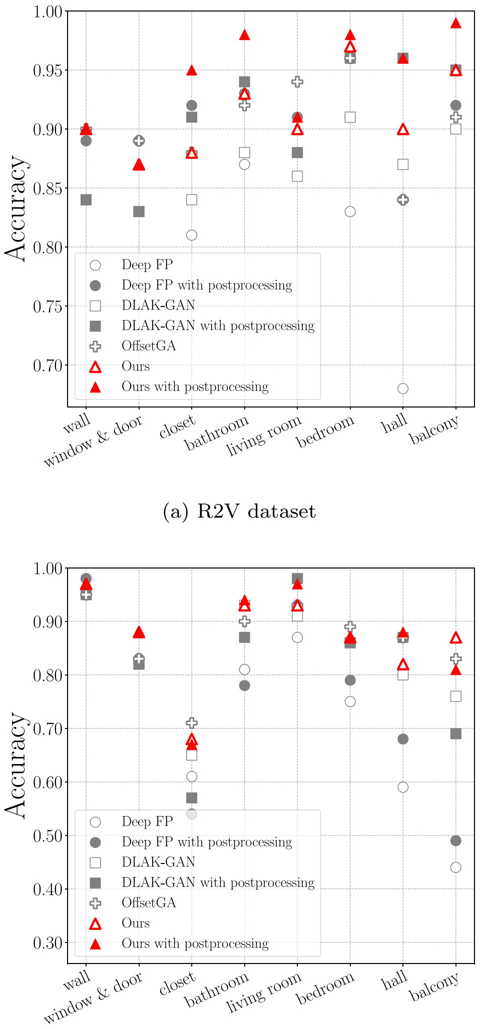
shown with and without applying a postprocessing method as first published for Deep FP. In this step, a closing operation first removes noise in the results. Subsequently holes in the predictions are filled, e.g. clusters of background pixels between the rooms and the wall. Moreover, a region enclosed by walls and openings, i.e. a room, is enforced to contain only pixels of one room class. A closer look at Fig. 6 reveals that our method can identify walls and openings even under challenging circumstances and moreover predict correct room types which have been misclassified by the other approaches. The semantic segmentation of walls and windows of our network overcomes deficiencies from preceding approaches even in challenging arrangements (boxes 2 & 3). Moreover, the superiority of our method in correctly predicting the room types is shown (boxes 1 & 4).

More detailed class-wise quantitative analyses using the introduced metrics and including results of OffsetGA [27] are shown in Table 1 based on the R2V and the R3D dataset respectively. For the R2V dataset, the improvement in overall accuracy amounts 0.03 and 0.02 when post-processing is applied. Taking the per-class accuracy into account, the results of our network outperform the preceding approaches in nearly all classes by gaining 0.01 to 0.06 in pixel accuracy. Only for the openings the result is slightly worse than Deep FP, and for the living room compared to OffsetGA. The postprocessing improves the results by up to 0.06. The visual comparison of these numbers in Fig. 7(a) again shows the substantial improvement, e.g. for balcony and bedroom, in pixel accuracy. Additionally, taking the global IoU into account, the gain is consistently 0.04 compared to the best approach to date, DLAK-GAN. For OffsetGA, no results with postprocessing were reported, since the authors designed their network specifically to omit this step. Nevertheless, we conducted the comparison with our results without postprocessing, since it was performed like that in their publication.

For the R3D dataset (cf. Table 1b), the overall accuracy is equal to DLAK-GAN and OffsetGA or slightly higher. Although previous approaches can yield better accuracies in some classes, the IoU, which also considers imbalances in the dataset, shows again a substantial improvement gaining 0.04 without and 0.05 with postprocessing. A visual comparison of the pixel accuracies can be found in Fig. 7(b). One of the highest improvements relates to the balconies, which are a particularly challenging class regarding their different styles and very thin boundaries. Applying the postprocessing improves our result for most classes, ranging from 0.01 to 0.06, but it also causes a drop in the pixel accuracy for closet and balcony, 0.01 and 0.06 respectively. However, the same pattern can be observed in the results of the preceding methods, even more significant and also affecting more classes. Nevertheless, we applied the postprocessing step for two reasons: (1) To ensure comparability with the other approaches and (2) due to the substantial improvements with regard to the global quality measures, i.e. IoU and overall accuracy in the R3D dataset. This also holds for the per-class accuracy, in the R3D dataset only except for the two mentioned classes, i.e. closet and balcony, and even for all classes for the R2V dataset. Hence, despite the drop in the two classes the postprocessing is in general worth to be applied.

An even broader comparison with, to the best of our knowledge, all recently published networks that have tested their approaches with at least one of the two datasets is shown in Table 2. Since for the additional approaches only the global measures, i.e. overall accuracy and mean IoU, were reported, the class-wise comparison is omitted. Upadhyay et al. [29] reported per-class accuracies, however, since the network was trained on a different dataset it is not fully comparable. Again, it can be observed that our approach performs best, in particular with significant gains of at least 0.04 with respect to the mean IoU.

Overall, our approach achieves substantial improvements, in particular in challenging classes such as balconies and closets (cf. Table 1a and b). Furthermore, our results turn out to be markedly better in terms of global quality measures, in particular the mean IoU. This reflects the superiority of our approach against the state-of-the-art methods.



(a) R2V dataset

(b) R3D dataset

Fig. 7. Graphical visualization of the quantitative evaluation from the state-of-the-art approaches and our method.

Table 2
Results for IoU and Overall Accuracy as reported for (1) DeepLab v3 [48], (2) Deep FP [6], (3) DLAK GAN [7], (4) OffsetGA [27], (5) FPNet [29], (6) VecMultimodInf [31], (7) VectorFloorSeg [30] and our approach.

		(1)	(2)	(3)	(4)	(5)	(6)	(7)	Ours
R2V	Mean IoU	0.69	0.74	0.81	0.77	–	0.80	0.81	**0.85**
	Overall accuracy	0.88	0.89	0.91	0.93	–	0.92	0.90	**0.94**
R3D	Mean IoU	0.50	0.63	0.74	0.74	0.73	–	–	**0.78**
	Overall accuracy	0.88	0.89	**0.94**	0.91	**0.94**	–	–	**0.94**

4.3. Ablation study

In this section, first the influence of the two-branched architecture compared to a single-task network is investigated, followed by a closer look at the effect of training both branches separately. Additionally, we modify the split between training and test data of the R3D dataset to investigate the generalization ability of our network.

Ad-hoc, the task at hand tempts to use a single-branch architecture. However, the ablation study emphasizes that the use of a single branch, i.e. the RoomNet with the ResNet Encoder leads to worse predictions for particular classes, such as openings and walls as can be seen in Table 3. Moreover, for the balcony class significantly worse results are reported, possibly due to the poorer recognition of the corresponding boundaries. Likewise, it would be possible to apply WallNet with the ResNet Encoder as a single-branch network. That would be, however, an ordinary CNN which yields worse results as proven in the previous publication by Zeng et al. [6]. This legitimizes our design decision of using a two-branch network with a customized branch, i.e. WallNet, as a more suitable architecture for walls and openings (cf. Table 4) as this contributed to much better results. It can be stated that our prediction results of the boundary structures, i.e. walls, doors and windows, significantly outperforms the method of Zhang et al. [7], DLAK-GAN, in both datasets already. Nevertheless, they are comparable with Deep FP with regard to the R2V dataset. The latter demonstrated that this method is superior compared to state-of-the-art edge detection approaches. We achieve, however, more accurate boundary results which even outperforms Deep FP on the latter dataset by training the RoomNet separately. The results without postprocessing are shown in Table 4. However, computing both segmentations separately and combining them afterwards leads to worse global quality measures compared to a joint training.

Table 3
Overview on the results on the R3D dataset with our complete network (1) and only using our RoomNet with the ResNet Encoder (2).

		(1)	(2)
Mean IoU		0.78	0.73
Per-class accuracy	Wall	0.97	0.89
	Door/Window	0.88	0.76
	Closet	0.68	0.64
	Bathroom	0.93	0.88
	Living room	0.93	0.95
	Bedroom	0.87	0.89
	Hall	0.82	0.80
	Balcony	0.87	0.75
	Background	0.98	0.99
Overall accuracy		0.94	0.93

Table 4
Per-class accuracy of walls, doors and windows for (1) DeepLab v3 [48], (2) Deep FP [6], (3) DLAK GAN [7], our approach and our WallNet trained separately (WN).

		(1)	(2)	(3)	Ours	WN
R2V	Wall	0.80	0.89	0.84	0.90	**0.93**
	Door/Win	0.72	0.89	0.83	0.87	**0.91**
R3D	Wall	0.93	0.98	0.95	0.97	**0.98**
	Door/Win	0.60	0.83	0.82	**0.88**	0.87

To further investigate the generalization ability of our network, we exploit a specific property of the R3D dataset: Zeng et al. [6] added additional 18 images stemming from New York to the dataset originally consisting of 214 images from London. Although the presentation and drawing of both floorplans are similar, there are significant differences. First, the New York floorplans consist only of round and irregular shapes, whereas the London floorplans do not contain any round buildings. Second, the housing units also differ with respect to their layout and the occurrence of certain classes of rooms: E.g., no balconies can be found in the New York data, but in return, there often exist walk-in closets that do not appear in the London data in this way. Moreover, only one hall can be found, which also involves an annotation error. Hence, the results presented in Table 5 focus on the remaining classes. It can be seen that the accuracy for the closet class is lower, whereas the bedroom is detected better. The other classes show only minor changes. Hence, the network is not overfitting on the London floorplans but shows the ability to generalize on the images from New York.

5. Applications, strengths and limitations

5.1. Application: Building a knowledge base for automatic building design

As shown in Fig. 1 different applications for the accurately segmented floorplan images exist. For example, learning-based methods

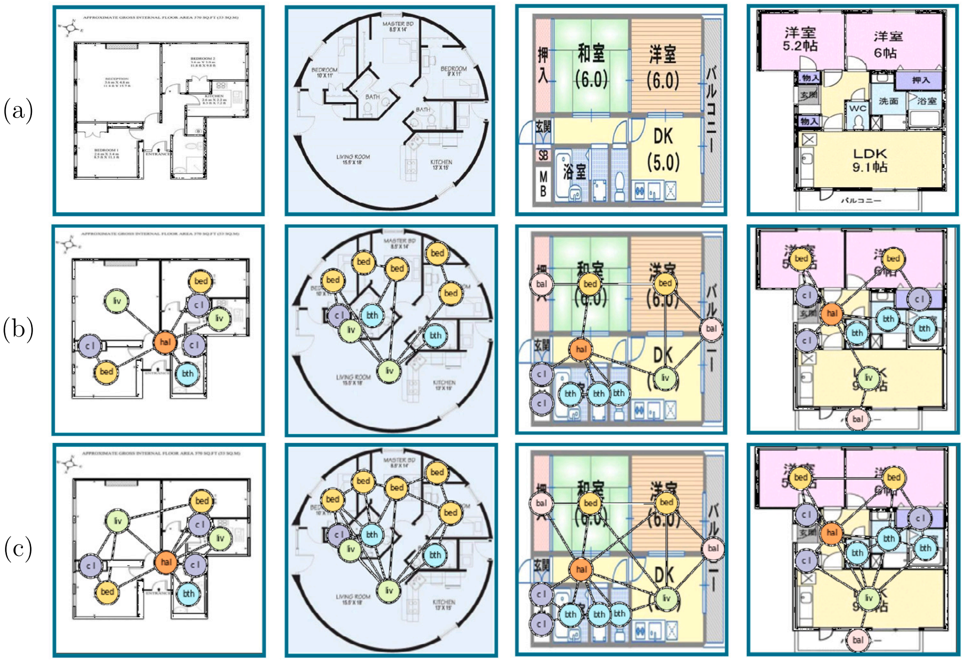


Fig. 8. Layout graphs retrieved from the result of our semantic segmentation: (a) input image, (b) connectivity graph (c) adjacency graph.

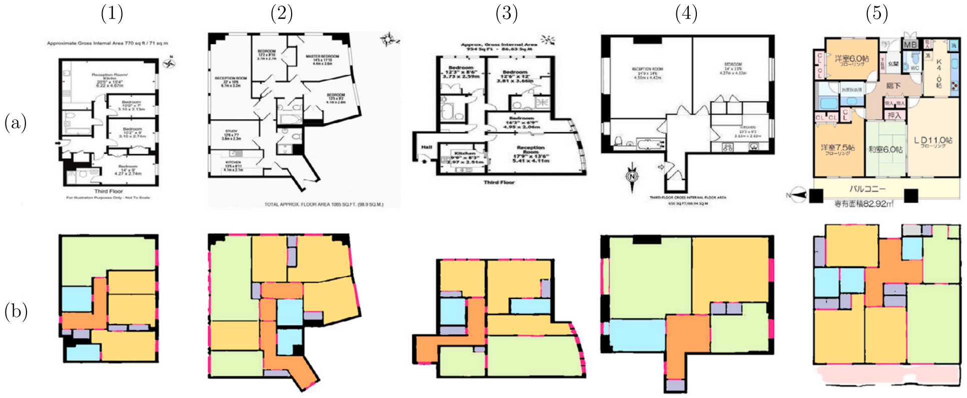


Fig. 9. Examples of our achieved results: (a) original images, (b) semantic segmentation results of our network. Room types, walls of different thickness and variety, exceptional door representations and challenging window arrangements can be detected.

Table 5
Overview on the results for the frequently occurring classes on the R3D dataset with the data split as used in previous publications (1) and using the modified split by the image origin (2).

		(1)	(2)
Per-class accuracy	Wall	0.97	0.96
	Door/Window	0.88	0.87
	Closet	0.68	0.62
	Bathroom	0.93	0.94
	Living room	0.93	0.95
	Bedroom	0.87	0.92
	Background	0.98	0.98
Overall accuracy		0.94	0.94

for the automated design process for new buildings require knowledge on the topological relations between the underlying structures. This can be expressed as bubble diagrams or layout graphs. The accurate results of the semantic segmentation performed by our network allow for obtaining this knowledge. Fig. 8 exemplarily presents different retrieved graphs. Given an input image (a) and the result of the semantic segmentation of our network, a connectivity graph (b) can be induced. Two nodes, i.e. rooms, are connected if a common door exists. Adjacency graphs (c) consist of rooms sharing a common wall accordingly.

5.2. Discussion

The conducted experiments and the comparative evaluation with the state-of-the-art approaches have shown that our network outperforms the preceding works. As depicted in Fig. 9, not only room types are accurately predicted, but also walls with different thicknesses and a high variety, i.e. irregular shapes such as parallel or curved lines, are detected. The identification of challenging door types and window arrangements has also been successfully addressed. In the same context, Zeng et al. [6] identified two main problems: Special room structures, such as double-bended corridors and specific icons, e.g. compass symbols, which are often falsely identified as walls. Our

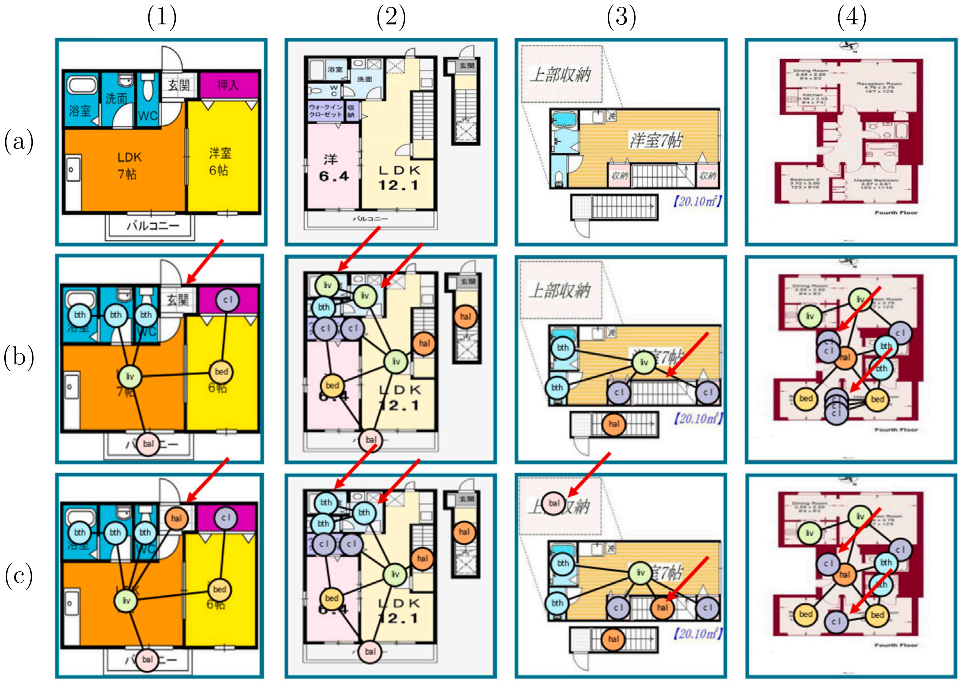


Fig. 10. Inferred layout graphs: Connectivity graphs derived from the ground truth (b) and from the result of our network (c). The red arrows indicate differences.

approach, however, handles both issues (cf. Fig. 9 (1)–(4)). The last column (5) of Fig. 9 reveals one limitation of our approach. Balconies in the R3D dataset are often bounded by very thin lines that diverge from the regular walls. In some cases, boundary lines are not available at all which makes the semantic segmentation of balconies even more challenging.

Some peculiarities of the labeled datasets pose additional problems for the training and influence the subsequent semantic segmentation of the floorplan images. First, especially in the R2V dataset, the pixels of a room in the ground truth do not always extend directly to the surrounding walls. This results in gaps which have been therefore inaccurately mislabeled as background. Second, the manual labeling causes inaccurate and inconsistent assignments of labels. These anomalies directly affect the trained model and consequently the prediction quality. Nevertheless, our method is able to detect and reveal falsely labeled elements in the ground truth. This is exemplarily depicted using the retrieved connectivity graphs from Fig. 10 where (b) shows the graphs derived from the annotated ground truth, whereas (c) visualizes the corresponding results based on our network. The red arrows indicate inconsistencies between both graphs. In the second example, two rooms are correctly predicted as bathrooms despite being mislabeled as living room in the ground truth. In the first and third column, our network correctly detects a hall although it has not been labeled in the ground truth. Likewise, the latter example shows an attic which is not annotated at all. Since our target classes do not include such a type, the network predicts it as a balcony. The last floorplan shows a further limitation of our approach: Even if the closets are correctly detected, they are often handled as one instance although two or more neighbored closets exist.

6. Summary and outlook

This paper presented an approach for the automatic semantic segmentation of floorplan layouts, following a two-branched strategy differentiating between the interior and the boundaries of rooms. The first task distinguished between room types, e.g. living and bed rooms, whereas the second further classifies the outlines into walls, doors and windows. To this aim, we designed a multi-task deep network combining a Convolutional Neural Network and a Graph Convolutional Network which beyond local structural relations allowed for further capturing long-range dependencies. The input graph of the GCN is automatically learned and provided by a Self-Constructing Graph module avoiding its manual design. This turns out to be a successful approach which outperforms state-of-the-art methods based on evaluations using benchmark datasets. Building upon the highly accurate results, we automatically derived both connectivity and adjacency graphs which could serve as prior knowledge for informed sampling of new layouts based on existing ones. Our method is even able to detect and reveal falsely labeled room types in the ground truth.

Inducing and collecting such graphs could not only serve as basis to automatically sample layout graphs for architectural design and planning, but also to predict missing links for unobserved parts for as-built building models and learn important latent topological and architectonic patterns. This represents an ongoing research topic and will be subject of a future publication. These results further pave the way for the application of our approach to predict indoor layouts of existing buildings and, hence, retrieve the according as-built state. To this end, sparse observations following the spirit of Dehbi et al. [49], e.g. window locations and the direction of the sun, could be integrated together with the retrieved topological information from the aforementioned graphs leading to indoor layout hypotheses which could be verified in order to get the underlying existing model.

The vectorization of the semantically interpreted floorplans by using Mixed Integer Linear Programming (MILP) will be subject of future work to provide a good basis for architectural design and planning tasks. Moreover, the automatically inferred graph for the GCN could also easily be augmented by more specific expert knowledge. The impact of such knowledge will also be investigated. Lastly, the successful application of SCGs for the semantic segmentation of floorplans opens up new questions in terms of explainability to better understand their impact. Making the induced latent dependencies from SCG visible and interpretable is another ongoing research topic.

CRediT authorship contribution statement

Julius Knechtel: Conceptualization, Data curation, Investigation, Methodology, Software, Visualization, Writing – original draft, Writing – review & editing, Formal analysis. **Peter Rottmann:** Methodology, Writing – review & editing, Validation. **Jan-Henrik Haunert:** Conceptualization, Resources, Supervision, Writing – review & editing, Project administration. **Youness Dehbi:** Conceptualization, Data curation, Methodology, Resources, Supervision, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The authors do not have permission to share data.

References

[1] Y. Dehbi, J. Knechtel, B. Niedermann, J.-H. Haunert, Incremental constraint-based reasoning for estimating as-built electric line routing in buildings, Autom. Constr. 143 (2022) 104571, http://dx.doi.org/10.1016/j.autcon.2022.104571.

[2] M. Vidanapathirana, Q. Wu, Y. Furukawa, A.X. Chang, M. Savva, Plan2Scene: Converting floorplans to 3D scenes, 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2021) 10728–10737, http://dx.doi.org/10.1109/CVPR46437.2021.01059.

[3] S. Ahmed, M. Liwicki, M. Weber, A. Dengel, Improved automatic analysis of architectural floor plans, in: Proceedings of the 2011 International Conference on Document Analysis and Recognition (ICDAR 2011), 2011, pp. 864–869, http://dx.doi.org/10.1109/ICDAR.2011.177.

[4] L. Gimenez, S. Robert, F. Suard, K. Zreik, Automatic reconstruction of 3D building models from scanned 2D floor plans, Autom. Constr. 63 (2016) 48–56, http://dx.doi.org/10.1016/j.autcon.2015.12.008.

[5] S. Macé, H. Locteau, E. Valveny, S. Tabbone, A system to detect rooms in architectural floor plan images, in: Proceedings of the 9th IAPR International Workshop on Document Analysis Systems (DAS 2010), 2010, pp. 167–174, http://dx.doi.org/10.1145/1815330.1815352.

[6] Z. Zeng, X. Li, Y. Yu, C.-W. Fu, Deep floor plan recognition using a multi-task network with room-boundary-guided attention, in: Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV 2019), 2019, pp. 9095–9103, http://dx.doi.org/10.1109/ICCV.2019.00919.

[7] Y. Zhang, Y. He, S. Zhu, X. Di, The direction-aware, learnable, additive kernels and the adversarial network for deep floor plan recognition, 2020, http://dx.doi.org/10.48550/arXiv.2001.11194, ArXiv abs/2001.11194.

[8] Q. Liu, M. Kampffmeyer, R. Jenssen, A.-B.r. Salberg, Multi-view self-constructing graph convolutional networks with adaptive class weighting loss for semantic segmentation, in: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW 2020), 2020, pp. 199–205, http://dx.doi.org/10.1109/CVPRW50498.2020.00030.

[9] P. Merrell, E. Schkufza, V. Koltun, Computer-generated residential building layouts, ACM Trans. Graph. 29 (6) (2010) 1–12, http://dx.doi.org/10.1145/1866158.1866203.

[10] W. Wu, L. Fan, L. Liu, P. Wonka, MIQP-based layout design for building interiors, Comput. Graph. Forum 37 (2018) 511–521, http://dx.doi.org/10.1111/cgf.13380.

[11] R. Hu, Z. Huang, Y. Tang, O.V. Kaick, H. Zhang, H. Huang, Graph2Plan: Learning floorplan generation from layout graphs, in: ACM Transactions on Graphics (Proceedings of SIGGRAPH 2020), Vol. 39, 2020, pp. 118:1–118:14, http://dx.doi.org/10.1145/3386569.3392391.

[12] C. Ah-Soon, K. Tombre, Variations on the analysis of architectural drawings, in: Proceedings of the Fourth International Conference on Document Analysis and Recognition (ICDAR 1997), Vol. 1, 1997, pp. 347–351, http://dx.doi.org/10.1109/ICDAR.1997.619869.

[13] P. Dosch, K. Tombre, C. Ah-Soon, G. Masini, A complete system for the analysis of architectural drawings, Int. J. Doc. Anal. Recognit. 3 (2) (2000) 102–116, http://dx.doi.org/10.1007/PL00010901.

[14] J. Zhu, H. Zhang, Y. Wen, A new reconstruction method for 3D buildings from 2D vector floor plan, Comput.-Aided Des. Appl. 11 (2014) 704–714, http://dx.doi.org/10.1080/16864360.2014.914388.

[15] K. Ryall, S. Shieber, J. Marks, M. Mazer, Semi-automatic delineation of regions in floor plans, in: Proceedings of 3rd International Conference on Document Analysis and Recognition (ICDAR 1995), Vol. 2, 1995, pp. 964–969, http://dx.doi.org/10.1109/ICDAR.1995.602062.

[16] L.-P. de las Heras, J. Mas, G. Sánchez, E. Valveny, Wall patch-based segmentation in architectural floorplans, in: Proceedings of the 2011 International Conference on Document Analysis and Recognition (ICDAR 2011), 2011, pp. 1270–1274, http://dx.doi.org/10.1109/ICDAR.2011.256.

[17] L.-P. de las Heras, S. Ahmed, M. Liwicki, E. Valveny, G. Sánchez, Statistical segmentation and structural recognition for floor plan interpretation, Int. J. Doc. Anal. Recognit. (IJDAR) 17 (2013) 221–237, http://dx.doi.org/10.1007/s10032-013-0215-2.

[18] C. Liu, J. Wu, P. Kohli, Y. Furukawa, Raster-to-vector: Revisiting floorplan transformation, in: Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV 2017), 2017, pp. 2214–2222, http://dx.doi.org/10.1109/ICCV.2017.241.

[19] X. Lv, S. Zhao, X. Yu, B. Zhao, Residential floor plan recognition and reconstruction, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021), 2021, pp. 16717–16726, http://dx.doi.org/10.1109/CVPR46437.2021.01644.

[20] I.Y. Surikov, M.A. Nakhatovich, S.Y. Belyaev, D.A. Savchuk, Floor plan recognition and vectorization using combination unet, faster-rcnn, statistical component analysis and ramer-douglas-peucker, in: International Conference on Computing Science, Communication and Security (COMS2 2020), Springer, 2020, pp. 16–28, http://dx.doi.org/10.1007/978-981-15-6648-6_2.

[21] W. Wang, S. Dong, K. Zou, W. Li, Room classification in floor plan recognition, in: 4th International Conference on Advances in Image Processing (ICAIP 2020), 2020, pp. 48–54, http://dx.doi.org/10.1145/3441250.3441265.

[22] S. Dong, W. Wang, W. Li, K. Zou, Vectorization of floor plans based on EdgeGAN, Information 12 (2021) 206, http://dx.doi.org/10.3390/info12050206.

[23] W. Huang, H. Zheng, Architectural drawings recognition and generation through machine learning, in: Proceedings of the 38th Annual Conference of the Association for Computer Aided Design in Architecture (ACADIA 2018), 2018, pp. 156–165, http://dx.doi.org/10.52842/conf.acadia.2018.156.

[24] S. Dodge, J. Xu, B. Stenger, Parsing floor plan images, in: Proceedings of the 15th IAPR International Conference on Machine Vision Applications (MVA 2017), 2017, pp. 358–361, http://dx.doi.org/10.23919/MVA.2017.7986875.

[25] T. Yamasaki, J. Zhang, Y. Takada, Apartment structure estimation using fully convolutional networks and graph model, in: Proceedings of the 2018 ACM Workshop on Multimedia for Real Estate Tech (RETech 2018), 2018, pp. 1–6, http://dx.doi.org/10.1145/3210499.3210528.

[26] W. Wu, Architectural floorplan recognition via iterative semantic segmentation networks, in: Proceedings of the 2023 7th International Conference on Computer Science and Artificial Intelligence, CSAI '23, Association for Computing Machinery, New York, NY, USA, 2024, pp. 282–287, http://dx.doi.org/10.1145/3638584.3638636.

[27] Z. Wang, N. Sun, Offset-guided attention network for room-level aware floor plan segmentation, IEEE Access 11 (2023) 63667–63677, http://dx.doi.org/10.1109/ACCESS.2023.3288598.

[28] L. Huang, J.-H. Wu, C. Wei, W. Li, MuraNet: Multi-task floor plan recognition with relation attention, in: M. Coustaty, A. Fornés (Eds.), Document Analysis and Recognition – ICDAR 2023 Workshops, Springer Nature Switzerland, Cham, 2023, pp. 135–150, http://dx.doi.org/10.1007/978-3-031-41498-5_10.

[29] A. Upadhyay, A. Dubey, S.M. Kuriakose, FPNet: Deep attention network for automated floor plan analysis, in: M. Coustaty, A. Fornés (Eds.), Document Analysis and Recognition – ICDAR 2023 Workshops, Springer Nature Switzerland, Cham, 2023, pp. 163–176, http://dx.doi.org/10.1007/978-3-031-41498-5_12.

[30] B. Yang, H. Jiang, H. Pan, J. Xiao, VectorFloorSeg: Two-stream graph attention network for vectorized roughcast floorplan segmentation, in: 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2023, pp. 1358–1367, http://dx.doi.org/10.1109/CVPR52729.2023.00137.

[31] T. Wen, C. Liang, Y.-M. Fu, C.-X. Xiao, H.-M. Xiang, Floor plan analysis and vectorization with multimodal information, in: D.-T. Dang-Nguyen, C. Gurrin, M. Larson, A.F. Smeaton, S. Rudinac, M.-S. Dao, C. Trattner, P. Chen (Eds.), MultiMedia Modeling, Springer International Publishing, Cham, 2023, pp. 282–293, http://dx.doi.org/10.1007/978-3-031-27077-2_22.

[32] P. Schönfelder, F. Stebel, N. Andreou, M. König, Deep learning-based text detection and recognition on architectural floor plans, Autom. Constr. 157 (2024) 105156, http://dx.doi.org/10.1016/j.autcon.2023.105156.

[33] W. Swaileh, D. Kotzinos, S.K. Ghosh, M. Jordan, S. Vu, Y. Qian, Versailles-FP dataset: Wall detection in ancient floor plans, in: Proceedings of International Conference on Document Analysis and Recognition (ICDAR 2021), 2021, pp. 34–49, http://dx.doi.org/10.1007/978-3-030-86549-8_3.

[34] H. Kim, Evaluation of deep learning-based automatic floor plan analysis technology: An AHP-based assessment, Appl. Sci. 11 (2021) 4727, http://dx.doi.org/10.3390/app11114727.

[35] P.N. Pizarro, N. Hitschfeld, I. Sipiran, J.M. Saavedra, Automatic floor plan analysis and recognition, Autom. Constr. 140 (2022) 104348, http://dx.doi.org/10.1016/j.autcon.2022.104348.

[36] C. Liu, A. Schwing, K. Kundu, R. Urtasun, S. Fidler, Rent3D: Floor-plan priors for monocular layout estimation, in: Proceedings of the 28th IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015), 2015, pp. 3413–3421, http://dx.doi.org/10.1109/CVPR.2015.7298963.

[37] Q. Liu, M.C. Kampffmeyer, R. Jenssen, A.-B. Salberg, SCG-Net: Self-constructing graph neural networks for semantic segmentation, 2020, http://dx.doi.org/10.48550/arXiv.2009.01599, ArXiv abs/2009.01599.

[38] W. Zi, W. Xiong, H. Chen, J. Li, N. Jing, SGA-Net: Self-constructing graph attention neural network for semantic segmentation of remote sensing images, Remote Sens. 13 (21) (2021) http://dx.doi.org/10.3390/rs13214201.

[39] Z. Luo, W. Huang, FloorplanGAN: Vector residential floorplan adversarial generation, Autom. Constr. 142 (2022) 104470, http://dx.doi.org/10.1016/j.autcon.2022.104470.

[40] Z. Lu, T. Wang, J. Guo, W. Meng, J. Xiao, W. Zhang, X. Zhang, Data-driven floor plan understanding in rural residential buildings via deep recognition, Inform. Sci. 567 (2021) 58–74, http://dx.doi.org/10.1016/j.ins.2021.03.032.

[41] M.A. Moradi, O. Mohammadrashidi, N. Niazkar, M. Rahbar, Revealing connectivity in residential architecture: An algorithmic approach to extracting adjacency matrices from floor plans, Front. Archit. Res. 13 (2) (2024) 370–386, http://dx.doi.org/10.1016/j.foar.2023.11.001.

[42] A.K. Verma, M. Jadeja, CB-SAGE: A novel centrality based graph neural network for floor plan classification, Eng. Appl. Artif. Intell. 126 (2023) 107121, http://dx.doi.org/10.1016/j.engappai.2023.107121.

[43] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2016), 2016, pp. 770–778, http://dx.doi.org/10.1109/CVPR.2016.90.

[44] O. Ronneberger, P. Fischer, T. Brox, U-Net: Convolutional networks for biomedical image segmentation, in: Proceedings of the Medical Image Computing and Computer-Assisted Intervention (MICCAI 2015), 2015, pp. 234–241, http://dx.doi.org/10.1007/978-3-319-24574-4_28.

[45] T. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: Proceedings of the 5th International Conference on Learning Representations (ICLR 2017), 2017, http://dx.doi.org/10.48550/arXiv.1609.02907.

[46] F. Milletari, N. Navab, S.-A. Ahmadi, V-Net: Fully convolutional neural networks for volumetric medical image segmentation, in: 2016 Fourth International Conference on 3D Vision (3DV), 2016, pp. 565–571, http://dx.doi.org/10.1109/3DV.2016.79.

[47] D. Kingma, J. Ba, Adam: A method for stochastic optimization, in: 3rd International Conference on Learning Representations (ICLR 2015), Conference Track Proceedings, 2015, http://dx.doi.org/10.48550/arXiv.1412.6980.

[48] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder-decoder with atrous separable convolution for semantic image segmentation, in: Proceedings of the European Conference on Computer Vision (ECCV 2018), 2018, pp. 833–851, http://dx.doi.org/10.1007/978-3-030-01234-2_49.

[49] Y. Dehbi, N. Gojayeva, A.R. Pickert, J.-H. Haunert, L. Plümer, Room shapes and functional uses predicted from sparse data, in: Proceedings of the International Society for Photogrammetry and Remote Sensing (ISPRS) Technical Commission IV Symposium, in: ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2018, pp. IV–4:33–40, http://dx.doi.org/10.5194/isprs-annals-IV-4-33-2018.