Architecture and Planning Journal (APJ)

Volume 28 Issue 3 *ASCAAD 2022 - Architecture in the Age of the Metaverse – Opportunities and Potentials*
ISSN: 2789-8547

Article 6

March 2023

GETTING A HANDLE ON FLOOR PLAN ANALYSIS - DOOR CLASSIFICATION IN FLOOR PLANS AND A SURVEY ON EXISTING DATASETS

JOÃO DAVID
INESC-ID/Instituto Superior Técnico, University of Lisbon, Portugal, joaodavid@tecnico.ulisboa.pt

ANTÓNIO LEITÃO
INESC-ID/Instituto Superior Técnico, University of Lisbon, Portugal, antonio.menezes.leitao@tecnico.ulisboa.pt

GETTING A HANDLE ON FLOOR PLAN ANALYSIS - DOOR CLASSIFICATION IN FLOOR PLANS AND A SURVEY ON EXISTING DATASETS

Abstract

Floor plan interpretation and reconstruction is crucial to enable the transformation of drawings to 3D models or different digital formats. It has recently taken advantage of neural-based architectures, especially in the semantic segmentation field. These techniques perform better than traditional methods, but the results depend mainly on the data used to train the networks, which is often crafted for the specific task being performed, making it hard to reuse for different purposes. In this paper, we conduct a literature survey on the existing datasets for floor plan analysis, and we explore how information regarding door placement and orientation can be recovered without having to change the initial data or model. We propose a two-step recognition method based on image segmentation followed by classification of cropped zones to allow data augmentation during training. In the process, we generate a dataset consisting of 35000 annotated door images extracted from an existing dataset.

Keywords

Floor Plan Analysis, Data Engineering, Machine Learning, Neural Networks, Dataset Survey.

GETTING A HANDLE ON FLOOR PLAN ANALYSIS
DOOR CLASSIFICATION IN FLOOR PLANS AND A SURVEY ON EXISTING DATASETS

JOÃO DAVID AND ANTÓNIO LEITÃO

INESC-ID/Instituto Superior Técnico, University of Lisbon, Portugal
joaodavid@tecnico.ulisboa.pt,
antonio.menezes.leitao@tecnico.ulisboa.pt

ABSTRACT

Floor plan interpretation and reconstruction is crucial to enable the transformation of drawings to 3D models or different digital formats. It has recently taken advantage of neural-based architectures, especially in the semantic segmentation field. These techniques perform better than traditional methods, but the results depend mainly on the data used to train the networks, which is often crafted for the specific task being performed, making it hard to reuse for different purposes. In this paper, we conduct a literature survey on the existing datasets for floor plan analysis, and we explore how information regarding door placement and orientation can be recovered without having to change the initial data or model. We propose a two-step recognition method based on image segmentation followed by classification of cropped zones to allow data augmentation during training. In the process, we generate a dataset consisting of 35000 annotated door images extracted from an existing dataset.

Keywords: Floor Plan Analysis, Data Engineering, Machine Learning, Neural Networks, Dataset Survey.

ملخص

يعد تفسير وإعادة بناء المخططات والمساقط الأفقية أمرًا بالغ الأهمية لتمكين تحويل الرسومات إلى نماذج ثلاثية الأبعاد أو تنسيقات رقمية مختلفة. وقد استفادت هذه التقنيات مؤخرًا من البنى القائمة على العصبية، خاصة في مجال التجزئة الدلالي. تؤدي هذه التقنيات أداءً أفضل من الطرق التقليدية، لكن النتائج تعتمد بشكل أساسي على البيانات المستخدمة لتدريب الشبكات، والتي غالبًا ما يتم تصميمها للمهمة المحددة التي يتم تنفيذها، مما يجعل من الصعب إعادة استخدامها لأغراض مختلفة. في هذه الورقة البحثية، نجري مسحًا أدبيًا حول مجموعات البيانات الحالية لتحليل المخططات والمساقط الأفقية، ونستكشف كيف يمكن استرداد المعلومات المتعلقة بوضع واتجاه الأبواب فى المسقط الأفقى دون الحاجة إلى تغيير البيانات الأولية أو نموذج المبنى. نقترح منهج للتعرف الآلى على خطوتين بناءً على تجزئة الصورة متبوعة بتصنيف المناطق التي تم اقتصاصها للسماح بزيادة البيانات أثناء التدريب. في هذه العملية، نقوم بإنشاء مجموعة بيانات تتكون من ٣٥٠٠٠ صورة باب مشروحة مستخرجة من مجموعة بيانات موجودة.

الكلمات المفتاحية: تحليل المساقط الأفقية، هندسة البيانات، التعلم الآلى، الشبكات العصبية، مسح لمجموعات البيانات.

1. INTRODUCTION

Floor plan interpretation and reconstruction have been a field of research for many years. Often, the goal was to enable the conversion of hand-made floor plan drawings to modern formats used in Computer-Aided Design (CAD). Compared to the traditional paper-based practices, having these models in digital formats is advantageous in terms of storage and sustainability. Furthermore, the conversion also enables the creation of 3D models from existing floor plans, making it possible to bring existing building designs into digital spaces such as the metaverse.

Initially, floor plan interpretation was performed using low-level image processing techniques and hand-crafted rules. By taking advantage of design conventions that dictate how floor plans should be represented, e.g., using thick lines for walls and thin lines for other elements, a common approach was to take an input image and separate the elements depending on line thickness (Dosch et al., 2000; Macé et al., 2020; Ahmed et al., 2012). Other image processing techniques were also used, such as the employment of the Hough Transform (Macé et al. 2010) to extract lines from the drawings. Then, symbols could be recognized using traditional recognition methods, such as, subgraph isomorphism (Llados, Marti and Villanueva, 2001).

More recently, image classification and segmentation have been dominated by deep-learning-based approaches, especially with Convolutional Neural Networks and Vision Transformers. These methods have been shown to perform and generalize better than traditional approaches based on hand-crafted rules. However, they also have disadvantages. The first drawback is the large amount of annotated data necessary to train a deep-learning model. As the performance of these models depends mainly on the data that are fed onto them, the data must be carefully annotated in large quantities, which is time consuming. Unfortunately, floor plan data is not as widely available as data for other tasks such as image classification, which has well-established benchmarking datasets, e.g., ImageNet (Deng et al., 2009). Not only is there a lack of benchmarking datasets for floor plan interpretation, but the available datasets also do not follow consistent annotation conventions. The annotations come as a necessity for each individual task, instead of a generalized description of the images. For example, door annotations usually only comprise an opening instead of the whole door symbol, as authors usually only target structural reconstruction. This can be seen in Figure 1.

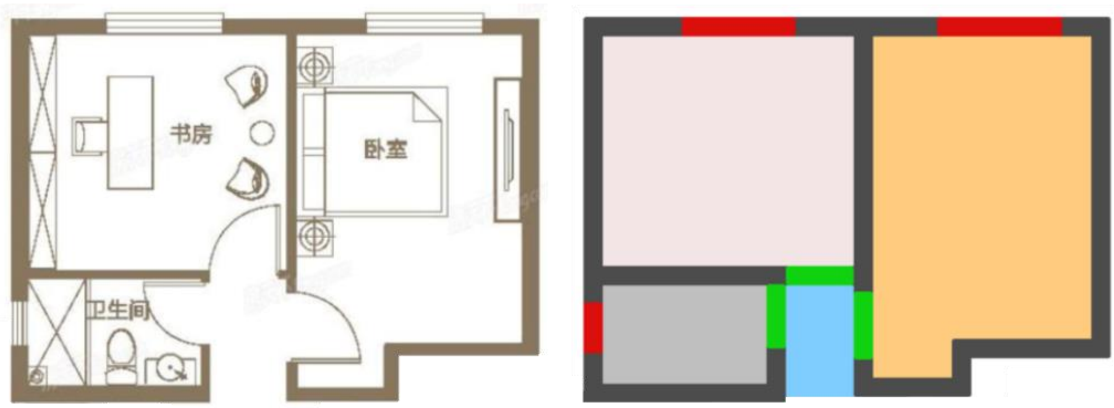


Fig.1: Ground truth annotation of a floor plan image. The annotation of the doors does not include the door swing area. Example adapted from (Lv et al., 2021).

More recent works (Fan et al., 2021; Simonsen et al., 2021; Song and Yu, 2021) have also explored the use of Graph Neural Networks (GNNs) for the classification of floor plan images or CAD drawings. These, unlike raster-based methods, can retrieve the segments that make up each component without the loss of information caused by methods based on convolution and pooling. Nonetheless, GNNs are often more resource intensive than raster-based methods.

In this paper, we focus on recovering the door type information lost after applying segmentation models. Due to the importance of data on these methods, in the next section we present a literature survey on the available datasets. Given that door classification depends on orientation, we propose a method that is invariant to geometric transformations, and we create a dataset consisting of 35000 annotated door symbols extracted from CubiCasa5K (Karlevo et al., 2019). Our code, in the form of a CubiCasa5K GitHub fork, is made publicly available to serve as a basis for future developments.[1]

2. LITERATURE REVIEW ON AVAILABLE DATASETS

Deep learning techniques require large amounts of data to perform well. In this section, we present a literature review on the most relevant floor plan datasets, which are summarized in Table number 1.

The CVC-FP (Heras et al., 2015) is one of the first datasets of floor plans. It consists of 122 real floor plan images divided into 4 categories depending on their origin and style. The elements are encoded with polygons in SVG format, which include walls, doors, windows, parking doors, and separations. Besides the visual elements, relationships between objects are also encoded, such as neighboring rooms and incident elements.

With the objective of aligning floor plans with in-house photos, Liu et al. (2015) introduce Rent3D, consisting of 215 floor plans and 1570 photos. The floor plans contain an annotated real-world scale, rooms and their respective types, walls, doors, windows, axis-aligned bounding boxes for icons, and relationships between rooms. For the image alignment task, the photos are each assigned to room elements. This dataset was recently extended by Vidanapathirana et al. (2021), creating the Rent3D++ dataset. This version does not add any images, but fixes and adds some categories and relationships between elements. The annotations are encoded in JSON files describing the different elements and CSV files containing the assignment of photos to rooms. Furthermore, the authors also introduce a dataset of room surface textures.

Dodge, Xu and Stenger (2017) introduce the R-FP dataset.[2] This dataset consists of 500 floorplan images. The ground truth only includes the wall segmentation, but a subset was annotated to perform object detection.

Liu et al. (2017), in their research, collected and annotated 870 images from the LIFEFULL HOME'S dataset.[3] The ground truths are encoded in text files describing the rooms, walls, doors, and some objects, which are all described by rectangles. While rooms are classified depending on their type, this dataset makes no distinction between doors and windows, which are distinguished later using hand-crafted rules. However, only the annotations are available.

Zeng et al. (2019) introduce two datasets, R2V and R3D, for floor plan image recognition using deep learning. The R2V dataset consists of the pixel-wise annotation of 815 images from the dataset in Raster-to-Vector (Liu et al., 2017). Similarly, R3D is the pixel-wise annotation of the Rent3D (Liu et al., 2015) dataset with 18 additional images. The pixel-wise classifications include wall, door, window, and room-type segmentation.

H. Kim, S. Kim and Yu (2021) propose a method for recovering spatial information in large floor plan images. The authors introduce a dataset consisting of 230 floor plan images from the Seoul National University (SNU). For the annotations, they use polygon boundaries for walls, doors, windows, staircases, and elevators. The dataset is available upon request to the corresponding author.

Wu et al. (2019) focus on generating house interior plans given a building boundary. They introduce the RPLAN dataset consisting of over 80000 pixel-annotated images from real houses. The annotations include room types and distinguishes between interior and exterior walls as well

[1] https://github.com/joaocmd/CubiCasa5k
[2] https://rit.rakuten.com/data_release/, 2022/04/03.
[3] https://www.nii.ac.jp/dsc/idr/en/lifull/, 2022/04/03.

as between main door and remaining doors. The authors also provide a toolbox to interact with the dataset; however, the original images are not publicly available.

CubiCasa5K (Karlevo et al., 2019) is a publicly available dataset, containing 5000 annotated floor plan images. Each raster image is associated with an SVG file that contains the geometric and semantic annotations of more than 80 object classes. The images are divided into 3 categories depending on their style, but there is a large amount of variation even within each category.

Lu et al. (2021) introduce RuralHomeData, which they use for floor plan understanding. This dataset includes the segmentation of 800 real-world floor plans of rural homes. Lv et al. (2021) also propose a system for converting raster floor plan images to a vector format, using their RFP (Residential Floor Plan) dataset, which is made up of 7000 images retrieved from the Internet. The annotations contain the segmentation of walls, doors, windows, and room types. Objects, scales, and text elements are also annotated, as well as doors and window openings. As far as we know, these two last datasets are not publicly available.

Simonsen et al. (2021) follow the newer graph-based approaches and propose a method to recognize doors in plans directly in DXF files. The authors introduce the Repository of Unique Buildings (RUB), collecting 81 floor plans in the DXF format. The files are used to generate two datasets, an image-based one containing the bounding box of the door elements and a graph-based one with node-wise classification.

The FloorPlanCAD (Fan et al., 2021) is another dataset of vector CAD drawings. As of May 2022, it contains 15663 drawings. The drawings are provided in SVG and PNG formats, where the elements are segmented and classified primitive-wise into 60 classes. However, private information that could identify the buildings was removed from the floor plans, and the plans are cut into squared blocks, where only 50% of them are kept. This is not a serious limitation as research have shown that good model results can be achieved with partitioned inputs (Lu et al., 2021).

The Systems Evaluation Synthetic Documents (SESYD) (Delalandre et al., 2010) is a public database with 1000 synthetic vector floor plans. Although the floor plans do not correspond to real-world buildings, their style is consistent with floor plan images of other datasets. The annotations are encoded in XML files with axis-aligned bounding boxes identifying the symbols, as well as some information regarding their orientation. The symbols include common objects, doors, and windows.

The FPLAN-POLY (Rusiñol, Borràs and Lladós, 2010) dataset contains 42 floor plans in the DXF format that have been vectorized from raster images. The dataset is public and aims to provide a framework for evaluating symbol spotting. Thus, the annotations correspond to labels assigned to the convex hull of the symbols.

The Repository of Building Plans (ROBIN) (Sharma et al., 2017) is a public dataset developed for floor plan retrieval tasks. It contains 510 floor plan images, each classified by its number of rooms and overall building shape. Another version called ROBIN++ is also available, consisting of the same floor plans but in a roughly sketched style.

The Building Plan Repository for Image Description Generation (BRIDGE) (Goyal et al., 2019) contains over 13000 floor plan images, including some that are present in other public datasets such as SESYD (Delalandre et al., 2010) and ROBIN (Sharma et al., 2017). The dataset aims to support tasks such as symbol spotting, caption generation, and description synthesis. As such, the annotations include bounding boxes and labels for the symbols, and free-text captions for parts of the images.

As we have seen in most of the datasets, the annotations vary largely, which is a challenge for accurate and complete reconstruction. Specifically, we explore the problem of correctly identifying door types and their orientation, which is often disregarded in the annotated data. In the next section, we cover how we define a convention for door classification and how we can make use of the available datasets to build a classifier.

DOI: 10.54729/2789-8547.1201

Table 1. Summary of datasets covered in Section 2. ✓* indicates that the dataset is available through an application and ✓ that only the annotations are available.**

Dataset	Size	Content	Public
CVC-FP (Heras et al., 2015)	122	Objects, structural elements, relationships	✓
Rent3D++ (Vidanapathirana et al., 2021)	215	Objects, structural elements, images, textures	✓*
R-FP (Dodge, Xu and Stenger, 2017)	500	Wall pixel segmentation, objects	✓*
Raster-to-Vector (Liu et al., 2017)	870	Room type, walls, openings, objects	✓**
R2V (Zeng et al., 2019)	815	Room type, walls, openings pixel segmentation	✓**
R3D (Zeng et al., 2019)	232	Room type, walls, openings pixel segmentation	✓
SNU (H. Kim, S. Kim and Yu, 2021)	230	Walls, doors, windows, stairs, elevators	✓**
RPLAN (Wu et al., 2019)	80000+	Room type, walls, openings pixel segmentation	✓**
CubiCasa5K (Kalervo et al., 2019)	5000	Objects, structural elements, room types	✓
RuralHomeData (Lu et al., 2021)	800	Room type, walls, openings pixel segmentation	✗
RFP (Lv et al., 2021)	7000	Room type, walls, doors, windows, objects, scales, text	✗
RUB (Simonsen et al., 2021)	81	Labeled CAD doors	✓
FloorPlanCAD (Fan et al., 2021)	15663	CAD primitive-wise segmentation	✓
SESYD (Delalandre et al., 2010)	1000	Objects, windows, doors	✓
FPLAN-POLY (Rusiñol, Borràs and Lladós, 2010)	42	Annotated objects' convex hulls	✓
ROBIN (Sharma et al., 2017)	1020	Floor plan shape/type classification	✓
BRIDGE (Goyal et al., 2019)	13000+	Objects, room and building captions	✓

3. DESCRIBING DOORS

Finding a method to describe doors is a challenging problem as it depends on orientation and, thus, on the point of view of the observer. In this section, we propose a solution to that problem.

First, we observe how every door can be defined by its two endpoints: *S* and *E* (*Start* and *End*). Then, the door symbols are described according to the position of the door axis relative to these two points. Visually, this can be thought of as moving the door so that it sits on top of the x axis on the Cartesian plane with S to the left of the y axis and E to the right of it. Then, the *Left*, *Right*, *Forward*, and *Reverse* labels depend on which quadrant the door opens to. This rationale is illustrated in Figure 2.

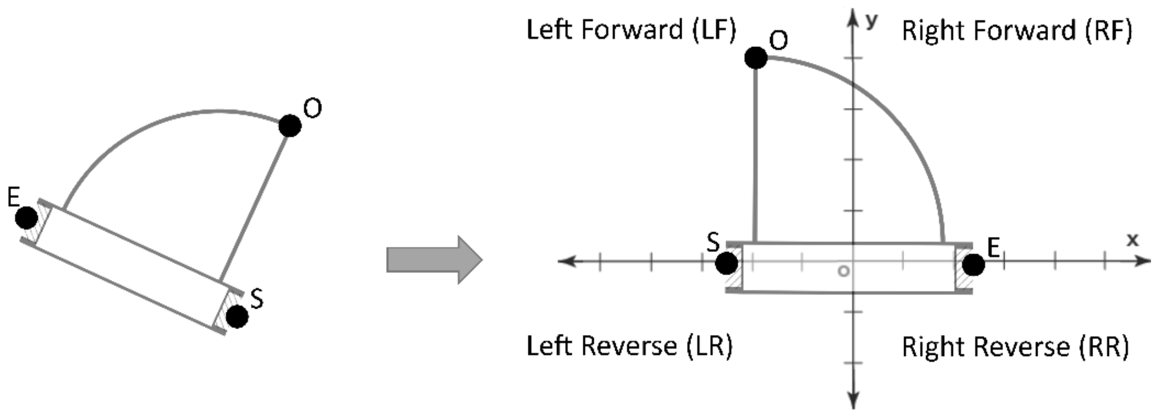


Fig.2: Process to describe a door symbol. S and E are placed on top of the x axis so that S is on the left. The quadrant of point O determines the door classification. This example corresponds to a Left-Forward (LF) door.

In addition to normal swing doors, we also describe other related symbols. Some swing doors can open in both directions and, sometimes, the door axis changes depending on the opening direction. For that reason, we define two door axes for these doors, one for each direction. Double doors consist of two swing doors that open in a single direction, where one has an axis on the left and the other on the right. Thus, these doors are described by their opening direction. Other types of doors such as sliding doors and roll up doors are also given a label, but we do not assign any specific information regarding opening direction. Nonetheless, the same ideas can be applied to classify them if that information is needed, as well as to classify other elements such as windows.

The following sections use these concepts to create a dataset and build a classifier capable of identifying door types in floor plan images.

4. DATASET DESCRIPTION AND GENERATION

To address the problem of recovering door information using the available data of previous works, we created a dataset based on the label description elaborated in the previous section. The resulting dataset consists of **35000** door images, varying in resolution and aspect ratio.

To abide by the previous description, we first define a total ordering of all coordinates, from the bottom left corner to the upper right corner. If two points are at the same value on the x axis, the first point is the one with the smallest y value. That is, let $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ then:

$$P_1 < P_2 \Leftrightarrow x_1 < x_2 \vee (x_1 = x_2 \wedge y_1 < y_2) \quad (1)$$

We use this total order to define each door by its two endpoints: S and E (start and end), where $S < E$. The imposed order is not necessary to describe a door symbol, but it is necessary to guarantee consistency in annotation across all symbols and images.

By applying this total order, we get the final labels which are illustrated in Figure 3. However, when we consider how the images are labeled, the dependency on a total order introduces a problem. Notice how, if point 7 moved right under point 6, their order would change and so would the classification of the doors between them, as the segment would be like the leftmost case. For this reason, these labels cannot be used if the input images are to be rotated during the training process, which is a common data augmentation technique. As a solution, the doors can be given a general class during segmentation and the specific class can be retrieved later by running a classifier on each isolated door symbol. This is our rationale for performing door orientation classification after segmentation, instead of trying to perform both at once during the segmentation.

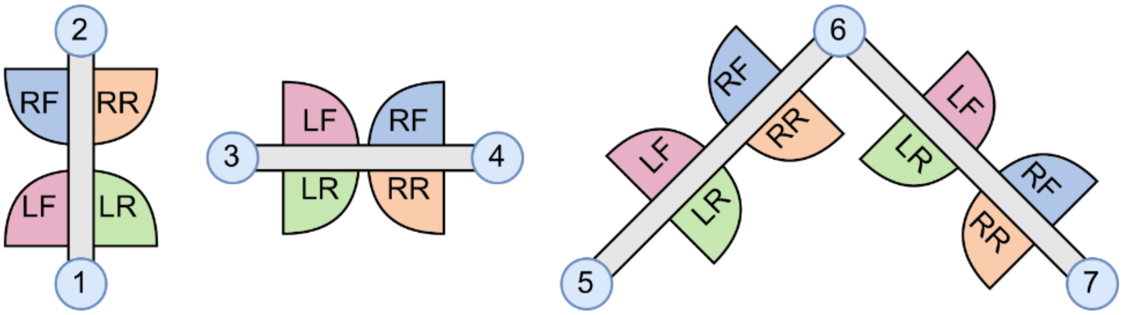


Fig.3: Door labels in different orientations. The characters mean "(L)eft", "(R)ight", "(F)orward", "(R)everse", and are combined to form 4 different classes. The numbers in the points correspond to their index considering the ordering criteria introduced in this section.

To create our data, we apply this process to the images in the CubiCasa5K (Karlevo et al., 2019) dataset. In that dataset, the annotations are encoded in SVG files, which were explored to extract the new images. The authors of CubiCasa5K use the SVG *id* attribute to classify each element; however, they included additional information in other fields. Namely, the *class* attribute identifies the type of door, of which we found: *Swing Beside*, *Swing Opposite*, *Slide*, *ParallelSlide*, *RollUp*, *Zfold*, and *None*.

The *Swing Beside* class is used for normal swing doors. We extract S and E, the start and end points, respectively, from the SVG element. The points are sorted according to the previously defined criteria, attending to the fact that image coordinates on the y axis are reversed. The representation of swing doors in the SVG files includes an arc, which we use to find the point reached by the door when fully open, which we will refer to as O. By calculating the distance between O and S and between O and E, we assign the label "L*(eft)*" if O is closest to S, and "*R(ight)*" otherwise. To assign the label "*F(orward)*" or "*R(everse)*", we must find if O sits above or below the $\overline{SE}$ segment, which can be done through the cross product between $\overrightarrow{SE}$ and $\overrightarrow{SO}$, after setting the z value of both vectors to zero, and then looking at the sign of the resulting z component to identify the door facing direction. The four possible scenarios are shown in Figure 4.

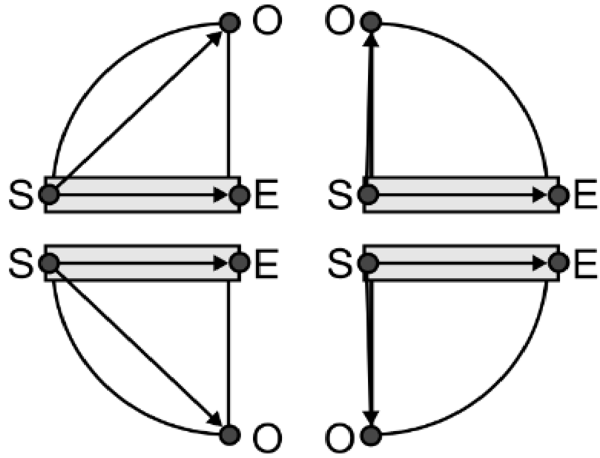


Fig.4: Possible scenarios for forward/reverse classification. The sign of the cross product is positive on the two top examples and negative on the bottom examples.

The other classifications build on the same concepts. Double doors are also annotated as *Swing Beside*, but have two arcs instead of one. In that case, we perform the forward/reverse classification on one of the arcs and classify them as double doors with the given direction.

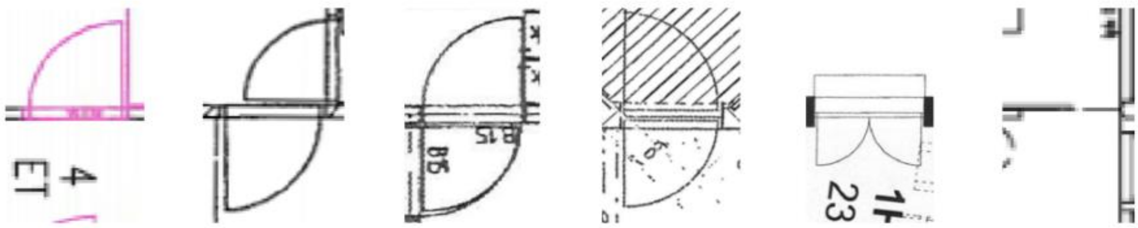
The *Swing Opposite* doors consist of single doors that swing in both directions. For these, we perform left/right classification for both arcs. To guarantee consistency, we start by labeling the forward arc and only then the reverse one. Figures 5c and 5d show two examples of such classes. As we have a defined order for classifying this type of doors, we can, for example, abbreviate "*RF-LR*" into "*RL*" and "*LF-LR*" into "*LL*".

Finally, for the remaining classes, we just use the *class* attribute directly. The final images are extracted by rotating the original images so that their corresponding walls are horizontal and then cropped to our region of interest.

We have, until this point, described how the data is extracted. We now focus on how the data is grouped and augmented. The data is split into a train set, a validation set, and a test set, according to the original floor plan images dataset. The images on each set correspond to the door symbols extracted from the corresponding floor plan images. As the floor plans in CubiCasa5K (Karlevo et al., 2019) are varied in style, the resulting dataset shares that same feature, and the images correspond to elements found in real floor plans with different surrounding contexts and noise.

Our defined classes allow us to use the same image for different classes. For example, a normal swing door with the label "*LF*" can be mirrored horizontally to create an "*RF*" image, mirrored vertically to create a "*LR*" image, and mirrored in both directions to create a "*RR*" image. The same is applicable to *Swing Opposite* doors, where "*LL*" can be transformed into an "*RR*" image, and "*LR*" into an "*RL*" image. Furthermore, by vertically mirroring an "*LL*" example, a different "*LL*" example is created, doubling the number of examples. For the other classes in which we did not consider any orientation, those can be mirrored four times to generate examples with a slightly different appearance. This, in turn, can be used to balance the number of examples for each class.

Example images can be seen in Figure 5. Dataset statistics are summarized in Figure 6, where we can see that single doors are the most common by a large margin, and folding doors are rarely used.



(a) Single-RF (b) Single-LR (c) Opposite-RL (d) Opposite-LL (e) Double-R (f) Slide

Fig.5: Examples of our labeled images. Each caption corresponds to the given image class.

DOI: 10.54729/2789-8547.1201

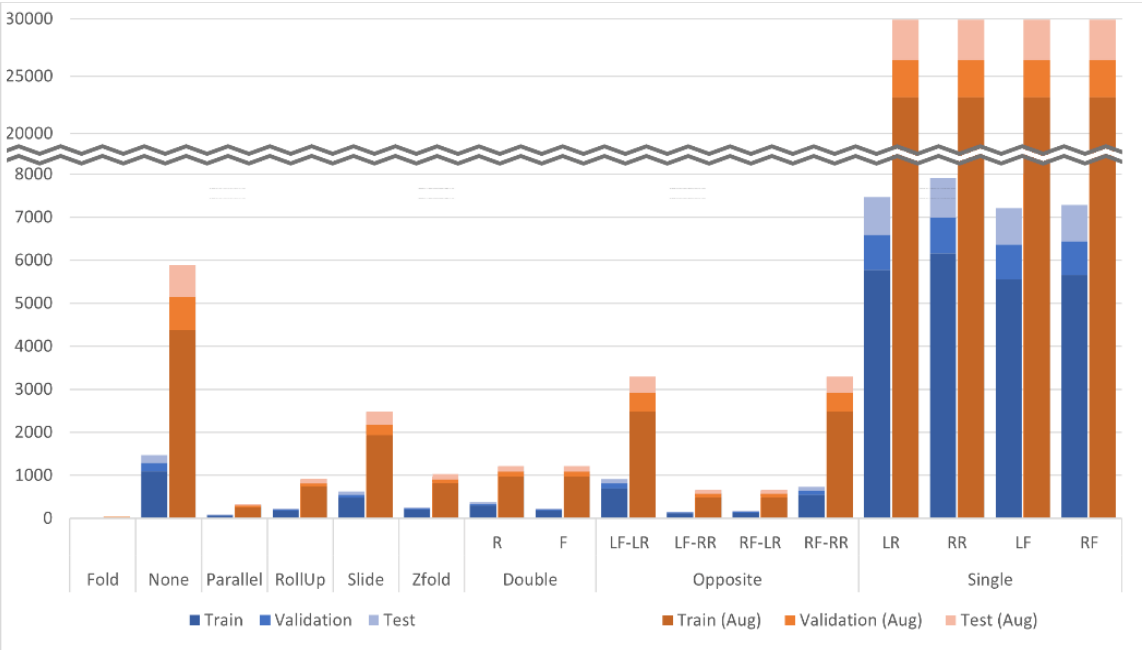


Fig.6: Number of examples for each label on the original and augmented datasets.

5. APPLICATION AND EVALUATION

The approach described in the previous section was developed to allow us to train a classifier for door symbols, as that information is lost during the segmentation process. In this section, we describe how we can combine both the segmentation and the classification processes.

We use the data generated in the previous section, augmenting all but the single swing doors as those largely overwhelm the other classes, and group all sliding doors into a single class. For our classifier, we chose to use a pre-trained ConvNeXt-Tiny (Liu et al., 2022) due to its small size, simple architecture, and good performance. We perform transfer learning on the generated data, using normal data augmentation techniques such as color shifting and geometric transformations.

The model achieves 90.7% overall accuracy on the test set. As we can see from Figure 7, it struggles with the distinction of the *slide* and *none* classes, having achieved only 66.8% accuracy on the *none* class by wrongly classifying most of the remaining as *slide*. However, we observe that these examples are also hard for a human to categorize, as the *none* class was used for wall openings, but other strange cases are present, such as crossed-out symbols or symbols with no clear category that could be easily confused. In most buildings, simple swing doors are used, and the model achieves excellent results for those, with an average of 97.7% accuracy and 94.6% precision for these classes.

Ground Truth \ Predicted	none	slide	rollup	double_d	double_u	opposite_ul_dl	opposite_ul_dr	opposite_ur_dl	opposite_ur_dr	single_dl	single_dr	single_ul	single_ur
none	497	126	13	6	5	1	0	0	1	19	26	27	23
slide	92	310	5	3	7	0	0	0	0	4	8	5	6
rollup	7	13	74	1	1	0	0	0	0	0	0	0	0
double_d	5	2	1	99	0	0	0	0	0	0	1	0	0
double_u	3	3	1	0	99	0	0	0	0	0	0	1	1
opposite_ul_dl	0	0	0	0	0	345	0	0	0	12	1	8	2
opposite_ul_dr	0	0	0	0	0	0	71	0	1	0	4	2	0
opposite_ur_dl	0	0	0	0	0	1	0	72	0	1	0	0	4
opposite_ur_dr	0	0	0	0	0	0	1	0	353	1	4	1	8
single_dl	8	2	0	0	0	3	0	1	0	859	6	2	1
single_dr	6	0	0	0	0	1	1	0	7	2	899	1	1
single_ul	10	1	1	0	0	4	0	0	0	1	1	820	7
single_ur	9	0	0	0	1	0	0	0	1	0	1	1	833

Fig.7: Confusion matrix of the trained model on the augmented test set.

We then integrate our classifier into a general recognition process, which is shown in Figure 8. Starting with the original image, we extract a pixel-wise segmentation. We use the model provided in the CubiCasa5K repository. The doors can be retrieved by finding the contours of the wall openings and approximating them with a rotated rectangle, then the rectangle is expanded so that it is able to fit any type of door. The start and endpoints are extracted from the rectangles by considering their longest sides. Then, similarly to the dataset generation, the points are ordered to maintain consistency in the image classes.

Fig.8: General process for door classification. The doors are recognized during segmentation, but the semantics are lost. We extract the zone where the door might be and use a classifier to recover that information. Floor plan image adapted from Lv et al.'s (2021) work.

In Figure 9, we see an example of our recognition process in an image from the CubiCasa5K (Karlevo et al., 2019) dataset. The image is part of the dataset's test set and, as such, the door symbols in the image were not used during the door classifier training. From the image, we can see that the only misclassified element was due to a segmentation error which considered only one door instead of two. Right above the misclassified door is a correctly classified element, as the smaller door was not detected by the segmentation network. For reference, we used the model proposed by Karlevo et al. (2019). Considering the segmentation error, the door classification model makes a reasonable prediction.

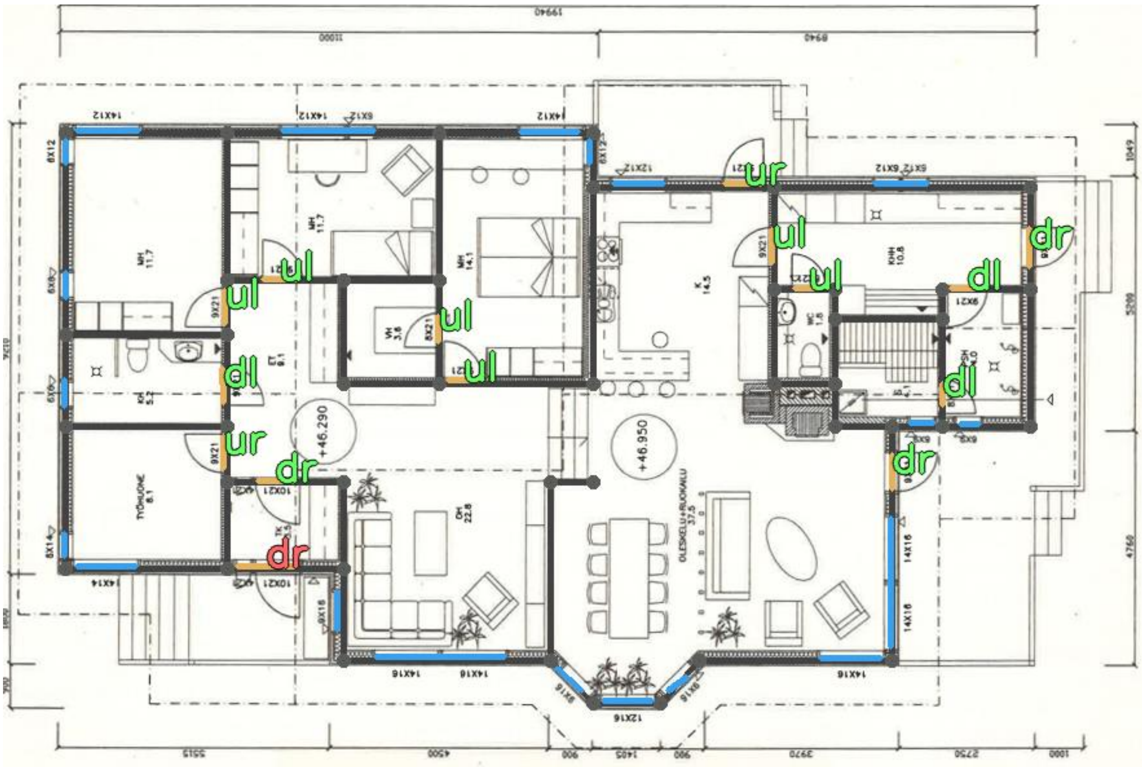


Fig.9: Door recognition results on an example image. In red, a misclassified door.

6. CONCLUSION AND FUTURE WORK

In this paper, we identified a drawback of the use of pixel-segmentation approaches in the analysis of floor plan images, namely, the information loss that happens regarding door placement and orientation. We proposed a method to classify doors present in floor plans, and a method to gather the necessary data to do so. To support this task, we performed a survey of the datasets that have been used for floor plan analysis.

One drawback of our approach is the need for a two-step classification process. It can be of interest to pursue a classification that can be inferred directly during the semantic segmentation process while still allowing data augmentation techniques to be used during model training. In this paper, we focused on the extraction of semantic content for swinging doors. For future work, we plan to apply the same approaches to the recognition of other elements such as sliding doors, roll up doors, and windows.

ACKNOWLEDGEMENTS

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with references UIDB/50021/2020, PTDC/ART-DAQ/31061/2017.

REFERENCES

- AHMED, S., LIWICKI, M., WEBER, M. AND DENGEL, A., 2012. *Automatic Room Detection and Room Labeling from Architectural Floor Plans.* s.l., s.n., pp. 339-343.
- DELALANDRE, M., VALVENY, E., PRIDMORE, T. AND KARATZAS, D., 2010. Generation of synthetic documents for performance evaluation of symbol recognition AND spotting systems. *International Journal on Document Analysis and Recognition (IJDAR),* 01 September, Volume 13, pp. 187-207.
- DENG, J. ET AL., 2009. *ImageNet: A large-scale hierarchical image database.* s.l., s.n., p. 248–255.
- DODGE, S., XU, J. AND STENGER, B., 2017. *Parsing floor plan images.* s.l., s.n., pp. 358-361.
- FAN, Z. ET AL., 2021. *FloorPlanCAD: A Large-Scale CAD Drawing Dataset for Panoptic Symbol Spotting.* s.l., s.n., pp. 10128-10137.
- GOYAL, S., MISTRY, V., CHATTOPADHYAY, C. AND BHATNAGAR, G., 2019. *BRIDGE: Building Plan Repository for Image Description Generation, and Evaluation.* s.l., s.n., pp. 1071-1076.
- HERAS, L.-P., TERRADES, O. R., ROBLES, S. AND SÁNCHEZ, G., 2015. CVC-FP and SGT: a new database for structural floor plan analysis and its groundtruthing tool. *International Journal on Document Analysis and Recognition (IJDAR),* 01 March, Volume 18, pp. 15-30.
- KALERVO, A. ET AL., 2019. *CubiCasa5K: A Dataset and an Improved Multi-task Model for Floorplan Image Analysis.* Cham, Springer International Publishing, p. 28–40.
- KIM, H., KIM, S. AND YU, K., 2021. Automatic Extraction of Indoor Spatial Information from Floor Plan Image: A Patch-Based Deep Learning Methodology Application on Large-Scale Complex Buildings. *ISPRS International Journal of Geo-Information,* Volume 10.
- LIU, C. ET AL., 2015. *Rent3D: Floor-Plan Priors for Monocular Layout Estimation.* s.l., s.n.
- LIU, C., WU, J., KOHLI, P. AND FURUKAWA, Y., 2017. *Raster-To-Vector: Revisiting Floorplan Transformation.* s.l., s.n.
- LIU, Z. ET AL., 2022. A ConvNet for the 2020s. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).*
- LLADOS, J., MARTI, E. AND VILLANUEVA, J. J., 2001. Symbol recognition by error-tolerant subgraph matching between region adjacency graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Volume 23, pp. 1137-1143.
- LU, Z. ET AL., 2021. Data-driven floor plan understanding in rural residential buildings via deep recognition. *Information Sciences,* Volume 567, pp. 58-74.
- LV, X., ZHAO, S., YU, X. AND ZHAO, B., 2021. *Residential Floor Plan Recognition and Reconstruction.* s.l., s.n., pp. 16717-16726.
- MACÉ, S., LOCTEAU, H., VALVENY, E. AND TABBONE, S., 2010. *A system to detect rooms in architectural floor plan images.* s.l., ACM, p. 167–174.
- RUSIÑOL, M., BORRÀS, A. AND LLADÓS, J., 2010. Relational indexing of vectorial primitives for symbol spotting in line-drawing images. *Pattern Recognition Letters,* Volume 31, pp. 188-201.
- SHARMA, D., GUPTA, N., CHATTOPADHYAY, C. AND MEHTA, S., 2017. *DANIEL: A Deep Architecture for Automatic Analysis and Retrieval of Building Floor Plans.* s.l., s.n., pp. 420-425.
- SIMONSEN, C. P., THIESSON, F. M., PHILIPSEN, M. P. AND MOESLUND, T. B., 2021. *Generalizing Floor Plans Using Graph Neural Networks.* s.l., s.n., pp. 654-658.
- SONG, J. AND YU, K., 2021. Framework for Indoor Elements Classification via Inductive Learning on Floor Plan Graphs. *ISPRS International Journal of Geo-Information,* Volume 10.
- VIDANAPATHIRANA, M. ET AL., 2021. *Plan2Scene: Converting Floorplans to 3D Scenes.* s.l., s.n., pp. 10733-10742.

- WU, W. ET AL., 2019. Data-driven interior plan generation for residential buildings. *ACM Transactions on Graphics (TOG),* Volume 38, p. 1–12.
- ZENG, Z., LI, X., YU, Y. K. AND FU, C.-W., 2019. *Deep Floor Plan Recognition Using a Multi-Task Network With Room-Boundary-Guided Attention.* s.l., s.n.