3D Modelling Approach for Ancient Floor Plans' Quick Browsing

Wassim Swaileh, Michel Jordan, Dimitris Kotzinos

3D Modelling Approach for Ancient Floor Plans' Quick Browsing[⋆]

Wassim Swaileh[1,2], Michel Jordan[1], and Dimitris Kotzinos[1]

University of Rennes, INSA of Rennes, IRISA, CNRS[2], 35000 Rennes - France
`wassim.swaileh@irisa.fr`
ETIS, UMR 8051, CY Cergy Paris Université, ENSEA, CNRS[1], 95000 Cergy - France
`michel.jordan@cyu.fr`
`dimitrios.kotzinos@cyu.fr`

Abstract. Although 2D architectural floor plans are a commonly used way to express the design of a building, 3D models provide precious insight into modern building usability and safety. In addition, for a historical monument like the Palace of Versailles, the 3D models of its ancient floor plans help us to reconstruct the evolution of its buildings over the years. Such old floor plans are hand made and thus present some problems in automatic creation of a 3D model due to the drawing style variability. In this paper, we introduce a fully automatic and fast method to compute 3D building models from a set of architectural floor plans of the Palace of Versailles dated to the 17th and 18th century. First, we detect and localise walls in an input floor plan image using a statistical image segmentation model based on the U-net convolutional neural network architecture and a binary wall mask image is obtained. Secondly, using the generated wall mask image, the 3D model is built upon the linear edge segments representing the detected wall sides in the mask image. In order to cope with the lack of accurate ground truth information for the 3D models of ancient floor plans, we use a dedicated semi-automatic software to build a set of reference 3D models that describe plans' wall projections from three sides of view. We evaluate the performance of our approach on an input floor plan image by measuring the overlapping between the 3D reference model and our 3D model. Our fast and fully automatic approach performs efficiently and produces quite accurate 3D models with 84.2% of IoU score in average. Furthermore, its performance surpasses the performance of the state of the art approach in the wall detection task.

Keywords: 3D modelling · Convolutional neural network · Ancient architectural archives

1 Introduction

Since the ancient architectural plans of a historical monument, such as the Palace of Versailles, reflect the evolution of the monument's buildings over the years,

---

[⋆] First and second authors contributed equally to this research

the conversion of these 2D plans into 3D models automatically, allows the visualisation of certain disappeared or unrealised places of the monument. Such old plans present several difficulties and challenges regarding drawing styles and paper format variabilities, in addition to foreground and background variable colour contrast. Due to these variabilities, building a 3D model from such floor plan images is not trivial. Actually, even creating the 3D model manually is not trivial either, and requires skill and time [24] In the literature, most of the related works detect walls in modern architectural drawings for building representative 3D models. Such drawings respect a set of standard protocols and symbols. However, architectural drawings in ancient floor plans show sometimes shapeless walls of variable drawing style [21]. The navigation through a virtual 3D environment becomes a common practice in order to to access the past of the historical monuments and is nowadays commonly used in various domains and applications such as cultural heritage, archaeology, virtual tourism, etc. (see for instance [6, 10]). Virtual reality and augmented reality, combined with semantic analysis, allow professionals, as well as the general audience, to visualise any kind of data including cultural heritage and historic data [14].

In this article, we present a 3D modelling approach that provides access to historical architectural archives by building 3D models in a fast and fully automated way. The paper is organised as follows: related works are presented in the following section. The proposed approach is then introduced in section 3. Finally, we present the evaluation protocol and system performance evaluation results in section 4 before introducing the conclusion and future works.

2   Related work

In the literature we observe that the 3D modelling process of scanned floor plans consists of two main steps: the wall detection process and the 3D model generation process. Indeed, while the wall detection is not the main task but can be a sub-task, there are some works which are only focused on wall detection and we discuss those works in the following section.

2.1   Wall detection

Traditionally, the wall detection task consists of several low-level image processing techniques including floor plan image noise reduction, graphics/text splitting, and drawing vectorisation. For detecting walls in ancient floor plan images, [22] used mathematical morphology operators in order to detect the main walls' drawings in a given input floor plan image, then built the corresponding 3D model. The proposed method requires human intervention to tune a threshold parameter in order to accurately detect thick walls and thin ones. In the same context, [21] introduced a U-Net convolutional neural network model for wall segmentation in ancient floor plan images. The model is trained to produce main wall mask images from grayscale floor plan images with no need for human intervention. We are using this work in our fully automatic 3D modelling processing chain.

In modern architectural drawings, many works are reported on wall detection and/or room segmentation. [16] used Hough transform to detect walls and doors. Subsequently wall polygons are created by the Hough lines and are partitioned iteratively into rooms, assuming convex room shapes. [1] first segment wall footprints in high-resolution images according to their line thickness. Then, they used geometrical reasoning to find room segments. SURF descriptors are used to detect doors. Convolutional neural networks were recently adopted to achieve the wall detection task. [4] used a simple neural network for denoising the plan image then used a fully convolutional neural network (FCN) for detecting walls. This approach is one of the state-of-the-art methods and performs well on multi-color contemporary floor plan images. [15] used a combination of deep neural networks and integer programming, where they first identify junction points in a given floor plan image and then join the junctions to locate walls in the floor plan. The method can only handle walls aligned with the two major axes of the floor plan image. Therefore, it can recognise layouts with only rectangular rooms and walls of uniform thickness, this is a critical limitation in the context of historical layouts where the shapes of the rooms are not always rectangular and often round shaped rooms are used. [23] also trained an FCN to label pixels in a floor plan image with multiple classes. The classified pixels formed a graphic model and were used to recover houses of similar structures.

As stated earlier all these works use different methods ranging from low level image processing to heuristics and furthermore to deep learning. However, the success of these methods in modern architectural documents cannot be replicated in the case of ancient architectural documents.

2.2   3D modelling

Nowadays, 3D representations are a useful tool for architects because they provide an intuitive view on the architects' work and help them to present their projects to their clients. Thus, the research about 3D model computation from architectural drawings is an active research field since more than 20 years [24]. Contemporary architectural floor plans are composed of geometric shapes (straight lines, curves) representing the building structure (external and inner walls), while various symbols represent openings, stairs, heating and furniture elements, *etc.* A lot of textual annotations provide information about the building's use, its rooms, dimensions, *etc.* This standard drawings give a strong basis for a lot of research works that succeeded in the floor plan analysis and automatic 3D modelling tasks. In [5], authors from the LORIA lab present a system for the analysis of architectural drawings, with the aim of reconstructing in 3D the represented buildings. Authors follow a set of graphics recognition steps including image processing and feature extraction. This system demonstrates some robustness though it requires moderate human assistance. Siu-Hang Or *et al.* [19] developed a system to solve a slightly simplified problem that considers only walls, doors, and windows. The system distinguishes walls as inner structures from building outlines and uses it to match neighbouring floors; during vectorisation, the system extracts outlines of black pixels in the raster image and matches

them with walls of various shapes. An improved example-driven symbol recognition algorithm is proposed for CAD engineering drawings in [11]. Firstly, in order to represent the structure of symbols, Guo *et al.* involve the text entity as one of the basic elements and redefine the relation representation mechanism. Then, the structure graph and a constrained tree is established for the target symbol, using the knowledge acquisition algorithm. In the recognition process, the nodes with the same type as the key features are located first in the drawing. The authors of [12] propose a database for structural floor plan analysis and the associated ground truth tool. The ground truth tool especially focuses on wall segmentation using the Hough line transform combined with an alignment heuristic fitting, and room detection.

Beside the 3D model computation, another field of interest concerning floor plan analysis is the need to refer to reference drawings when designing new buildings and to find solutions for similar architectural problems. In [2], authors present a system for semantic search from an architect's sketch: the semantic information of reference floor plans is extracted thanks to a "divide and conquer" strategy that separates text from graphics, and then graphics between thick lines (mainly walls) and thin lines (symbols). A further structural analysis allows us to perform the room detection and labelling. The DANIEL architecture presented in [20] is a deep learning model that learns the representative floor plan features using a convolutional neural network (CNN); the existing datasets provide the examples for the learning step. The authors of this paper proposed the ROBIN dataset that they used for evaluation.

The historical floor plans such as the ones we are working with have some drawing and digitizing specific features that do not allow their processing by one of the system described in previous section: the symbols for openings, doors, stairs, furniture, *etc.*, are different depending of the epochs and architects, and the digitization retains marks of the document state (for instance creases, since these documents have been kept in folders during long periods) or stamps that are on the drawing's reverse side. For these reasons, some of us developed in a previous project a system which copes with these particularities and help to build an accurate 3D model from the floor plans, thanks to a limited number of user interactions [22]. This system demonstrates a good ability to perform this task, but is not appropriate when processing a lot of drawings.

In the field of cultural heritage and architecture, some projects deal with text and image extraction from archived documents [18], other projects intend to build 3D models from various data sources such as photographs, laser scanning or point clouds [9]. The European Time Machine project intends to build 3D models of historical European cities across the centuries and analyses for this purpose various types of data including texts and graphics such as cadastral data; this allows to build city 3D models but not to enter the building interiors. In fact, it seems that the architectural drawing collection describing the palace of Versailles between 17th and 18th centuries is exceptional and lead to particular problems and particular algorithmic solutions.

3 Proposed approach

Our approach relies on three main steps: the floor plan digitization step, the wall mask generation step and the 3D modelling step. There is no need for human intervention during the whole processing chain and the result is obtained in a few seconds. Such fast and automatic 3D model construction system allow fast access to disappeared or non-realised architectural projects that interest specialists as well as the general public (sections 3.1-3.2-3.3). Figure 1 illustrates our processing pipeline from the floor plan image as input to the resulting 3D model.
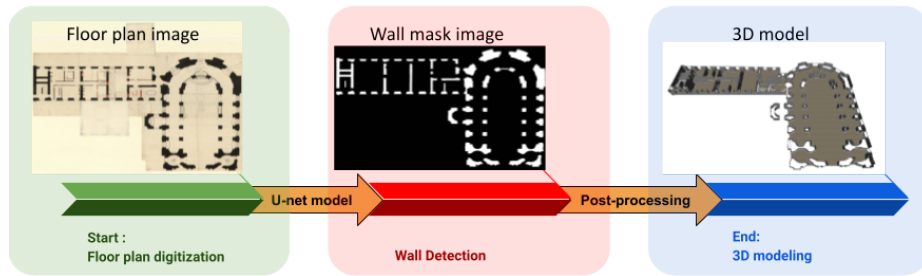

Fig. 1. Summary scheme of the processing pipeline

3.1 Floor plan digitization

The ancient floor plans of the Palace of Versailles have various sizes and drawing styles. This is due to the lack of standards for the drawing styles and paper formats at that age, and to the long period (about 120 years) covered by these plans. The images of these floor plans, including those of Versaille-FP dataset, were captured in a very high resolution. Our approach receives a multi-colour floor plan image as input. Then, it converts it into grayscale in order to reduce the colour noise and increase the intensity between the foreground and background colours at the input image. After that, it scales the grayscale image down to images of 512×512 pixels. To maintain the aspect ratio of the original input image, we square padded the input image to its largest dimension before applying the scaling process. Once the scaled image is obtained, we apply the wall segmentation model based on the U-net convolutional neural network to obtain the wall mask image. The wall mask image is a binary image where the foreground colour represents the wall regions detected by the segmentation model in the scaled input image.

3.2 Wall mask generation

At the second step of our proposed approach we used a U-Net architecture of convolutional neural network (CNN) to generate the wall mask image from the

input image. We used the same CNN model architecture introduced by [21] and illustrated in figure 2. We choose this architecture due to its precious proprieties of high speed and performance with small amount of data, according to [8]. The CNN model consists of two symmetric paths connected at a bottle neck layer. The first path namely called contracted path extracts contextual and semantic information from the input image. The second path (expanding path) achieves the accurate localization, by copying, cropping and concatenating information that are observed by the contracting path toward the expanding path. The model
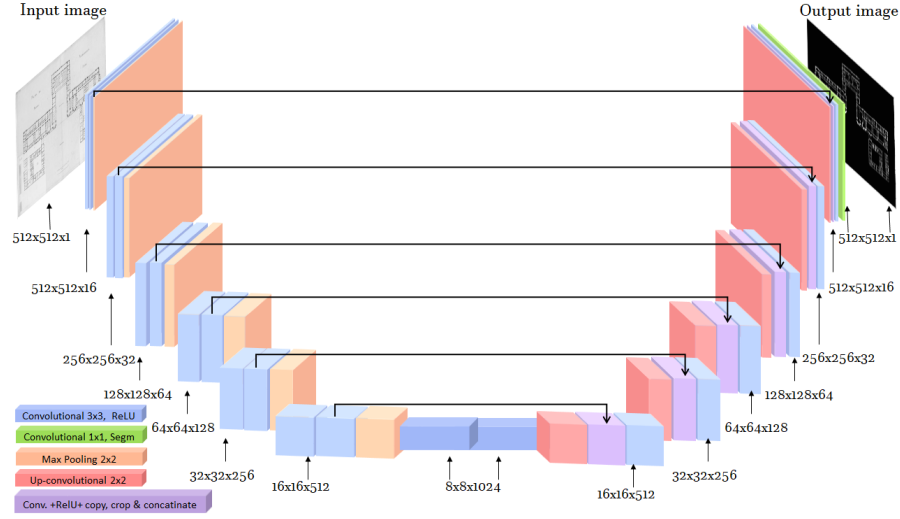

Fig. 2. U-net architecture for detecting walls in ancient floor plans

is trained in two steps of sequential learning where data augmentation is applied. The first step consists in learning an initial model on the modern CVC floor plan dataset samples [13]. As the Palace of Versailles dataset samples have a more noisy background than the CVC-FP ones, the white background in the CVC original samples are replaced by Versailles coloured plans background. This step provides a good initial training for the final model [21]. At the end of the first learning step, an optimised initial model is obtained. Next, the weights of the optimised initial model are used as initial weights for the final model to be trained on Versailles-FP dataset[1] samples during the second step of the sequential training.

The loss function L_D of Dice [17] is used to train the model. The Dice Co-efficient is an overlapping measure between two samples. It ranges from 0 to 1 where a Dice coefficient of 1 denotes perfect and complete overlap. The Dice loss is simply equals to $1-$ *Dice coefficient*. The L_D loss value is calculated as

---

[1] https://www.etis-lab.fr/versailles-fp/

in equation 1.

$$L_D = 1 - \frac{2 \times |A \cap B| + 1}{|A| + |B| + 1} \tag{1}$$

where A is the neural network prediction about the segmentation and B is the ground truth value. We compared the performance of the U-net model using different loss functions including the Dice loss with which we obtained the best results. Similarly, we selected the Rmsprop optimisation algorithm for learning our model. We used a learning rate that start with $10e^{-4}$ with batch size of 25 and we selected the best model during the learning process. We applied learning rate decay whenever validation accuracy does not improve. Although our model training is guided by the validation datasets, learning rate adjustment based on validation accuracy still works well according to our experience. It is worthy to mention that our model differs from the state art model [21] by the models' hyper-parameters fine tuning strategies, specially with regards to the batch size and the learning rate decay. The model training is achieved in 4 days using 2 x GPU Nvidia GTX 1080 11Go. We used early stopping with patience of fifty epochs.

To produce wall mask image for an input floor plan image, we applied the final CNN model on the scaled grayscale image at the input. The generated wall mask image is then feed to the next process of the pipeline in order to produce the corresponding 3D model. In the next section, we describe in detail the 3D model construction process and its related sub-processes.

3.3   3D model generation

We develop in this section an algorithm for a fast computation of 3D models from a wall mask image such as the one provided by the deep learning architecture in section 3.2. This algorithm includes the following steps:

- pruning of the residual small structures in the wall mask image;
- computation of edges;
- polygonalization of the edges;
- computation of the 3D model based on the wall polygons.

These steps consist of an arrangement of well-known image processing algorithms, whose parameters are fixed.

Small structures pruning Though the wall image computation is an accurate and carefully designed process, it may persist some pixels either isolated or in thin structures. The first step of our algorithm is a morphological opening with a rectangular 3×3 structuring element. We heuristically fixed the morphological opening number of iterations according to the wall image size. This operation allows not only to discard noisy pixels in the wall image, but also to remove thin structures such as dividing walls (*cf.* figure 3).
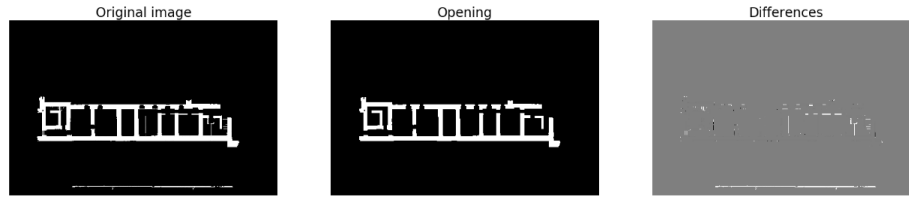
Fig. 3. Discarding thin structures: original wall image (left), after processing (center), difference image (right).

Edge detection and polygonalization The center image of Figure 3 represents the basis for the 3D model extrusion. The walls in the 3D model have inner and outer sides, since in future steps they could be covered with more or less realistic textures. We compute wall contours by means of the Canny edge detector [3] with σ parameter 0.1, and then perform edge polygonalization using a probabilistic Hough line detector [7] (*cf.* figure 4).

Fig. 4. Edge (left) and straight line (right) images computed from center image in fig. 3.

3D model building The 3D model is computed as an extrusion upon the straight lines of figure 4, at an arbitrary height that we fixed as 0.15 of $\max(H, W)$ where H and W are the initial image dimensions, in order to get a clearly visible 3D model. This height will be easily modified in further processes, depending on the final objective of the 3D modelling. Each line resulting from the polygonalization step gives an edge and two vertices for the 3D model, two other vertices are positioned vertically upon the first two, and these four vertices finally shape a vertical face (*cf.* figure 8).

4   Results and Evaluation

In this section, we present the experimental framework we followed to evaluate the performance of our fast and automatic 3D modelling approach. First, we introduce the ancient floor plan dataset that we used. Then, we present the
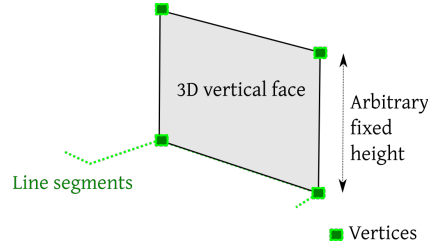
Fig. 5. A 3D face is computed upon a polygonal line.

evaluation protocol and results for a set of 15 3D models obtained by our approach and evaluated against reference 3D models that were produced using the VERSPERA semi-automatic 3D modelling tool [22].

4.1 Dataset

In this study, we used the ancient floor plan *"Versailles-FP"* dataset which is based on the french VERSPERA research project[2]. The dataset consists in 500 annotated images collected from scanned floor plans that belongs to the Palace of Versailles constructions dated of 17th and 18th century. The VERSPERA research project was started in 2013 with the aim of digitizing a large amount of graphical documents related to the construction of the Palace of Versailles during the 17th and 18th centuries. There is a large corpus of floor plans including elevations and sketches present in the collection of French National Archives. The total number of documents in this archive is around 6,500 among which about 1,500 are floor plans. An ambitious project to digitize this varied corpus started in 2014, extraordinary technical capability is needed to achieve this task due to fragile and varied nature of the paper documents (for example some document can be as big as 3m × 4m). The digitized plans of the Palace of Versailles consist of graphics that illustrate the building architecture such as walls, stairs, halls, king rooms, royal apartments, *etc.* in addition to texts and decorations. Since the palace of Versailles digitized floor plans cover 120 years (1670-1790) of architectural design, different drawing styles clearly appear in the corpus.

4.2 Evaluation protocol

Since most of the used ancient floor plans do not refer to real buildings of the Palace of Versailles of today, a straightforward numerical evaluation of the produced 3D models is not applicable. In fact, there is no absolute 3D ground truth for these data: 3D measurements in real rooms is not ever possible, since these floor plans represent parts of the monument that have been rebuilt or destroyed; even in the case of the current state of the monument is the same as the plan drawings, we can not infer that a floor plan of the 18th century is a perfectly

---

[2] https://verspera.hypotheses.org/

scaled representation of the monument. In fact, the only absolute reference would be an entirely manual processing of the floor plan images by computer graphics professionals, which is a time consuming and expensive task. We tried thus to compare some of these 3D models with the ones generated from the same floor plan images by means of the VERSPERA software [22]. This software is an interactive semi-automatic application that we developed in a previous research project; for a rather experienced user, building a 3D model with this application requires between 5 and 10 minutes. The VERSPERA software involves the following steps: *(i)* image denoising and preprossessing, including downsizing to a roughly similar size for all plans; *(ii)* image binarization; *(iii)* wall detection through mathematical morphology tools and manual thresholding; *(iv)* wall footprint extrusion for 3D modelling. The VERSPERA software allows some other processings such as staircase detection and wall decoration, that we do not consider in this study. In the following sections, we compare our method and the VERSPERA software method by means of two criteria: computation time and 3D model accuracy.

4.3   Results

Wall detection evaluation To evaluate the performance of the wall detection model we distinguish between two training schemes; 1) training from scratch where the model weights are initialised arbitrary for training the model on the Versailles-FP dataset samples directly, 2) sequential training in two phases; I) training from scratch the model on the CVC modern floor plan dataset, and II) continue the training of the model on the Versailles-FP training set. For the wall detection task, we used the same 5 fold cross validation protocol introduced in the Versailles-FP dataset paper [21]. We provide in figure 6 a set of good and bad wall detection images (images with black background). Bad results (lower row of the figure) occur when sketch scale differs too much from the learning set (lower left example) or sketch technique is different (pencil — lower right example — against ink in the learning set).

  We applied our software to floor plans of other monuments, freely available in the Gallica database from the French national Library (fig. 7); results appear to be not so good for these low resolution images, it will be interesting to add some of these examples in a transfer learning set to add more versatility to our models.

  Figure 8 presents a comparison of the wall detection task evaluations using the accuracy, IoU and Dice coefficient scores for both cases of "learning from scratch" (blue lines, with best Dice score of 93%), and the "sequential learning" (red lines with best Dice score of 94%). From this two learning tasks, we observe that the sequential learning of the pre-trained model is better than learning from scratch. In addition, our model outperforms the state of the art model. The slit performance enhancement is obtained thanks to the applied training parameter settings that are different of the state of the art ones. We think that the pre-trained model is learning better when using fine tuned learning rate with bigger
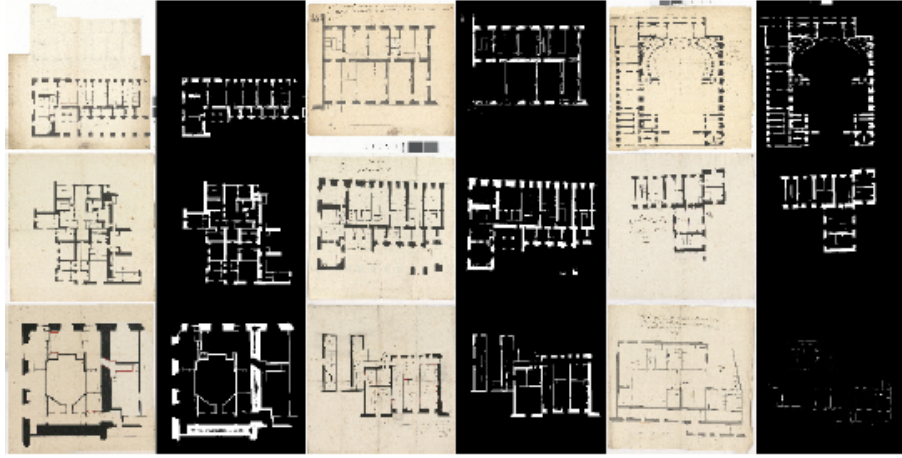
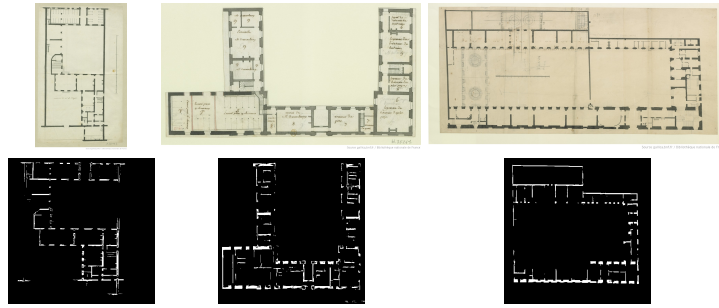**Fig. 6.** Illustration of wall detection good and bad results.

Fig. 7. Some examples of wall mask computation for other monuments.

batch size, recalling that the state of the art batch size is 16 since we used batch size of 25.

Computation time Figure 9 presents some 3D model examples of floor plans we computed using the approach described in section 3.3. Computing 3D models for the 500 wall mask images of the Versailles-FP dataset took around 20 minutes on a laptop with a *Intel Xeon CPU E3-1505M* processor, which gives an average of 2.4 seconds for each image. In comparison with the computation time of the reference 3D models constructed using the interactive VERSPERA software, we observe that our approach is 25 time faster than the interactive one. We give on table 1 a comparison of computation times for both methods.

3D model accuracy We compare first our models by a visual examination and comparison of the resulting 3D models for both automatic and interactive
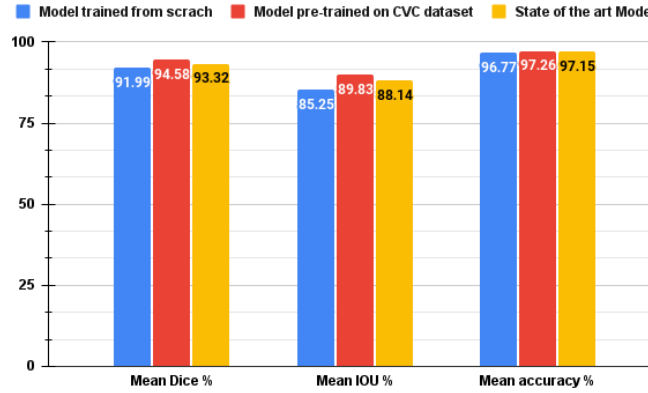
Fig. 8. Wall detection evaluation using pre-trained / trained from scratch model and comparison with the state-of-the-art.

Method	VERSPERA software (interactive)	Our method (fully automatic)
Nb of images	15	500
Overall time	120 min	20 min
Time *per* image	8 min	2.4 sec

Table 1. Comparison of computation times.

methods (see figure 10). Then, in order to get a more accurate and quantitative evaluation, we overlap the polygonalized wall mask image resulting of our approach with the vertical ground projection of the 3D model resulting from the VERSPERA software (fig. 10, right column); in fact, these two images are the basis of the 3D model extrusion, and comparing them gives a rather good approximation of fidelity between both 3D models since, as we discussed, a real reference 3D model is missing when dealing with the ancient floor plans. The right column images show small differences (non-white pixels) between the two masks that do not lead to drastic 3D model differences. For a numeric evaluation, we computed the min, max and average values and standard deviation of the IoU scores for the overlapping masks on a set of 15 models; table 2 sums up these values and shows a 84.2% average value, which is a rather good value depending on the final objective of the 3D modelling.

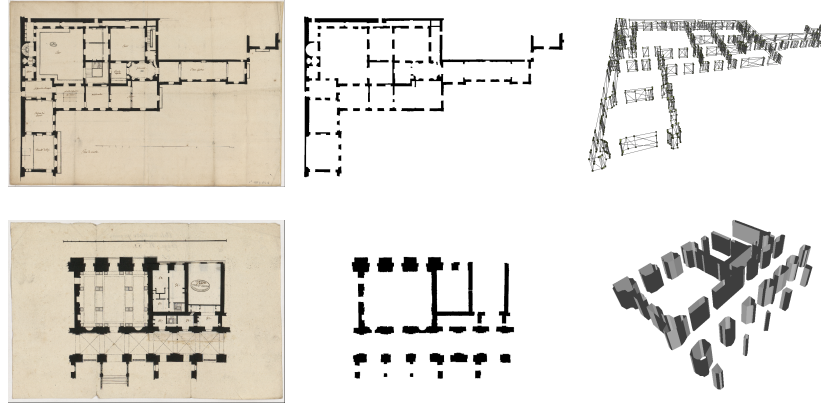| Min	Max	Average	Std dev.
0.780	0.906	0.842	0.041

Table 2. IoU scores computed on 15 models.

Fig. 9. Some 3D models (wireframe or solid screen capture on right) computed from the floor plan (left) and wall mask images (middle).

5 Perspectives and conclusion

We presented in this paper a fast 3D model computation approach for historical floor plan quick browsing. The proposed approach uses the U-net architecture of the convolutional neural neworks for detecting the wall regions in an input floor plan image, resulting in a binary wall mask image. The resulting image is filtered in a way that only the main walls remain, corresponding to the main structures of the monument. Finally, a 3D model is computed by extruding 3D faces upon linear segments corresponding to the wall edges. We applied this method to build 3D models from the 500 floor plan images of the Versailles-FP dataset (floor plans representing the palace of Versailles between 17th and 18th centuries), and were able to build 3D models very fast . This tool gives historians and archivists a mean to have a fast and quick but complete 3D perspective at the floor plans, and thus to get an easy perception and understanding of the volumes of the monument represented on the plan. Further work includes the integration in 3D models of other architectural details such as staircases, by means of a dedicated machine learning approach. We will also apply our algorithms to other historical datasets of architectural plans, if available.
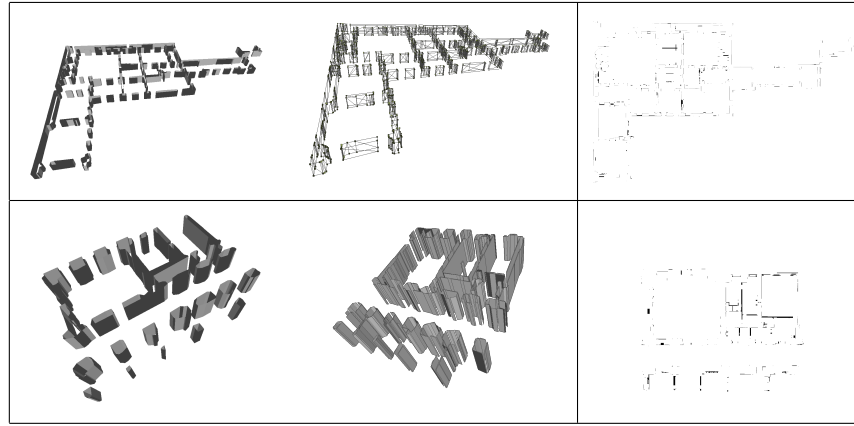
Fig. 10. Comparison of 3D models: VERSPERA semi-manual software (left), our automatic and fast approach (center), and wall mask difference images (right).

References

1. Ahmed, S., Liwicki, M., Weber, M., Dengel, A.: Automatic room detection and room labeling from architectural floor plans. In: 2012 10th IAPR DAS. pp. 339–343 (2012)
2. Ahmed, S., Weber, M., Liwicki, M., Langenhan, C., Dengel, A., Petzold, F.: Automatic analysis and sketch-based retrieval of architectural floor plans. Pattern Recognition Letters **35**, 91–100 (2014), frontiers in Handwriting Processing
3. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence **PAMI-8**(6), 679–698 (1986)
4. Dodge, S., Xu, J., Stenger, B.: Parsing Floor Plan Images. In: MVA (May 2017)
5. Dosch, P., Tombre, K., Ah-Soon, C., Masini, G.: A complete system for the analysis of architectural drawings. International Journal on Document Analysis and Recognition **3**, 102–116 (2000)
6. Fiorucci, M., Khoroshiltseva, M., Pontil, M., Traviglia, A., Del Bue, A., James, S.: Machine learning for cultural heritage: A survey. Pattern Recognition Letters **133**, 102–108 (2020)
7. Galambos, C., Kittler, J., Matas, J.: Progressive probabilistic hough transform for line detection. Proceedings. 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149) **1**, 554–560 Vol. 1 (1999)
8. Gao, H., Yuan, H., Wang, Z., Ji, S.: Pixel deconvolutional networks. arXiv preprint arXiv:1705.06820 (2017)
9. Grilli, E., Özdemir, E., Remondino, F.: Application of machine and deep learning strategies for the classification of heritage point clouds. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences **XLII-4/W18**, 447–454 (2019)
10. Grilli, E., Menna, F., Remondino, F.: A review of point clouds segmentation and classification algorithms. The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences **42**, 339 (2017)

11. Guo, T., Zhang, H., Wen, Y.: An improved example-driven symbol recognition approach in engineering drawings. Computers & Graphics **36**(7), 835–845 (2012), augmented Reality Computer Graphics in China
12. de las Heras, L.P., Terrades, O.R., Robles, S., Sanchez, G.: CVC-FP and SGT: a new database for structural floor plan analysis and its groundtruthing tool. International Journal on Document Analysis and Recognition **18**(1), 15–30 (2015)
13. de las Heras, L.P., Terrades, O.R., Robles, S., Sánchez, G.: CVC-FP and SGT: a new database for structural floor plan analysis and its groundtruthing tool. IJDAR **18**(1), 15–30 (2015)
14. Lampropoulos, G., Keramopoulos, E., Diamantaras, K.: Enhancing the functionality of augmented reality using deep learning, semantic web and knowledge graphs: A review. Visual Informatics **4**(1), 32–42 (2020)
15. Liu, C., Wu, J., Kohli, P., Furukawa, Y.: Raster-to-vector: Revisiting floorplan transformation. In: Proceedings of the IEEE ICCV. pp. 2195–2203 (2017)
16. Macé, S., Locteau, H., Valveny, E., Tabbone, S.: A system to detect rooms in architectural floor plan images. In: Proceedings of the 9th IAPR DAS. pp. 167–174 (2010)
17. Milletari, F., Navab, N., Ahmadi, S., V-net: Fully convolutional neural networks for volumetric medical image segmentation. In: Proceedings of the 2016 Fourth International Conference on 3D Vision (3DV). pp. 565–571 (2016)
18. Oliveira, S.A., Seguin, B., Kaplan, F.: dhsegment: A generic deep-learning approach for document segmentation. CoRR **abs/1804.10371** (2018)
19. Or, S., Wong, K., Yu, Y., Chang, M.: Highly Automatic Approach to Architectural Floorplan Image Understanding & Model Generation. In: Vision, Modeling & Visualization 2005. pp. 25–32. Erlangen, Germany (2005)
20. Sharma, D., Gupta, N., Chattopadhyay, C., Mehta, S.: Daniel: A deep architecture for automatic analysis and retrieval of building floor plans. In: 2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR). vol. 01, pp. 420–425 (2017)
21. Swaileh, W., Kotzinos, D., Ghosh, S., Jordan, M., Vu, N.S., Qian, Y.: Versailles-FP Dataset: Wall Detection in Ancient Floor Plans. In: Lladós, J., Lopresti, D., Uchida, S. (eds.) Document Analysis and Recognition – ICDAR 2021. pp. 34–49. Springer International Publishing (2021)
22. Tabia, H., Riedinger, C., Jordan, M.: Automatic reconstruction of heritage monuments from old architecture documents. Journal of Electronic Imaging **26**(1), 011006 (2016)
23. Yang, S.T., Wang, F.E., Peng, C.H., Wonka, P., Sun, M., Chu, H.K.: Dula-net: A dual-projection network for estimating room layouts from a single rgb panorama. In: Proceedings of the IEEE/CVF Conference on CVPR. pp. 3363–3372 (2019)
24. Yin, X., Wonka, P., Razdan, A.: Generating 3d building models from architectural drawings: A survey. IEEE Computer Graphics and Applications **29**(1), 20–30 (2008)