# Bouygues Floor Plan Detection Update 1

*Kaihong Wang, Chi Zhang, Qitong Wang, Jiangshan Luo*
*{kaiwkh, czhang1, wqt1996, Jasonluo}@bu.edu*
***Boston University Department of Computer Science***

## 1. Project Task

The goal of this project is to develop a machine learning model to estimate the surface area for each room in an apartment, given its floor plan. More precisely, the objectives are:

- Create a model that reads a PDF floor plan and convert it into an image floor plan, (see Fig 1.).
- Use the provided segmentation tool to segment each room of a given image floor plan.
- Implement a model calculates the surface area of each room, based on the given scaling legend.
- Design an algorithm to detects the type of each room, such as living room, kitchen, and bedroom.
- Output all predictions and saves them in a CSV file (see Table 1.).

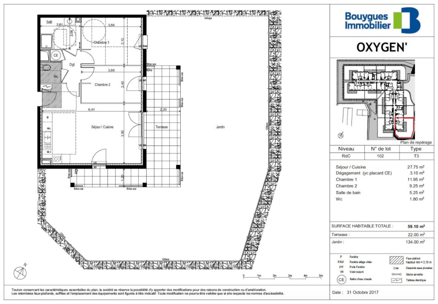


Figure 1. Input PDF Floor Plan

| ID | Building Name | Unit Num | Num of Rooms | Living Room | … ... |
|---|---|---|---|---|---|
| 1 | oxygen | 102 | 5 | 27.75 $m^2$ | … ... |

Table 1. Output Table Template

## 2. Related Work

Some previous works used deep neural network geometric and semantic information into a set of junctions [1] [3]. Then they implemented an integer programming aggregated junctions into a set of simple primitives (which are the contour of the floor plan). However, we would like to try a different approach. For room type detection, previous works mainly had two strategies: one way is using a neural network with a set of primitives [1]; the other way is creating a text layer and extracting text information from the input image [4].

## 3. Dataset and Metric

Bouygues Group provided total 133 training floor plans, which contain two buildings “Eko” and “Equation” with 49 and 84 pdf floor plan files, respectively, along with the training dataset ground truth table. The testing dataset contains 139 floor plans from the “Oxygen” building. The template table for the testing dataset result is provided, however, the table requires us to manually fill in the floor plan information (Number of Rooms, Surface Areas, etc) for each apartment. Note that all words and labels in the training and testing floor plans are in French, Bouygues Group also provided a reference room type chart, which translates French (some with abbreviations) to English. For example, (see Table 2.).

| | | | |
|---|---|---|---|
| French | Chambre | ch | ch1 |
| English | Bedroom | Bedroom | Bedroom |

Table 2. French Reference Table

Besides, Bouygues Group provided a template for the output of floor detection, where 29 types of rooms are mandatory targets and 20 room’s types are optional tasks.

### 3.1 Metrics

For surface area detection, we calculate Intersection Over Union (IOU) as the accuracy of each room surface area and set the distance to be $1-IOU$, as previously introduced in [1]. For room type labeling, we simply check whether our prediction room name matches with the ground truth (GT) label, The accuracy function is defined:

$$Accuracy(Room\ labeling) = \frac{Number\ of\ correct\ predictions}{GT\ number\ of\ rooms} \quad (m1)$$

In addition, the accuracy function of the number of main rooms is defined:

$$Accuracy(Room\ numbers) = G(|Predicted - GT\ num\ of\ rooms|)$$

m2, where G is a Gaussian-shape function, G(0) = 1, G(±∞)=0

Note that IOU accuracy is an independent metric. While m1 and m2 are correlated, if m1 equals to 1 then m2 equals to 1. If m2 does not equal to 1, m1 must not equal to 1. We hope our model can achieve accuracies higher than 80% on surface area, room type labeling, and the number of rooms predictions.

## 4. Approach

### 4.1 Room Area Calculation

Our approach integrates a contour detector via OpenCV, a contour area calculator, and a classifier model to predict the type of each room, for example, a Convolutional Neural Network(CNN) model, features involve room area, room shape, and objects information.

Preliminarily, we planned to build a contour detector extracts each room from the input PDF file and calculates the surface area by using erode and dilate method in OpenCV [5]. After meeting with TA Dharmesh, we decided to use the provided floor plan segmentation tool. Then, we calculate contour areas of the segmented floor plan using OpenCV methods.

### 4.2 Room Type Detection

For detecting the type of each room, our CNN model uses a multiclass softmax regression and its loss layer uses cross entropy loss, where the hypothesis and the loss function defined as follow [6]:

$$p(y^{(i)} = j|x^{(i)};\theta) = \frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^{k} e^{\theta_l^T x^{(i)}}}$$

F1. Hypothesis Function

$$J(\theta) = -\frac{1}{m}\sum_{i=1}^{m}\sum_{j=1}^{k}[1\ \{y^{(i)} = j\}\ log(\frac{e^{\theta_j^T x^{(i)}}}{\sum_{l=1}^{k} e^{\theta_l^T x^{(i)}}})]$$

F2. Loss Function without regularization

Note that in F2, 1{.} is the indicator function, so that 1{a true statement} = 1, and 1{a false statement} = 0.

For the CNN room's type model, we cogitate on two approaches: scanning the important objects in each room, (for example, the hotplates in the kitchen); and extracting the room type words from the input floor plan (which using the Google Vision API [7]).

## 5. Initial Results

We used the provided segmentation tool to extract the segmented floor plan from the original image (Fig 2.). Firstly, we used erode and dilate methods in OpenCV to extract all contours of a given segmented image, and set all contours as a binary black and white image floor plan (Fig 3.). Secondly, we calculate each contour's pixel size area by using OpenCV, then we use our template scaling legend to convert a pixel size room area to a room area in square-meter. The conversion function is defined:

$$Scaling\ Constant\ (S\_const) : how\ many\ pixels\ per\ meter$$

$$Room\ area\ (m^2) = \frac{Room\ area\ (pixel)}{(S\_const)^2}$$

Thirdly, we print room areas in square-meter for all rooms on the segmented floor plan (Fig 4.). Based on our initial result (Table 3.), we achieved around 90% accuracy on the surface area detection task.

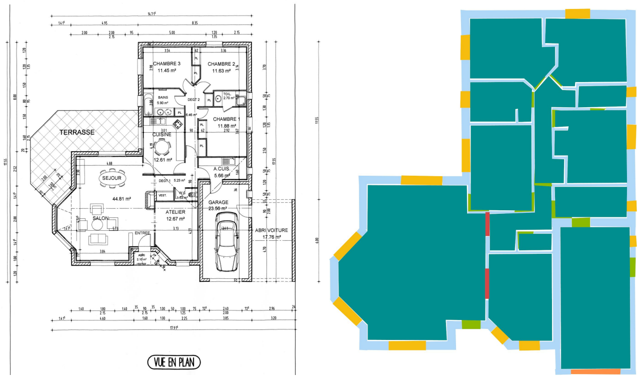


Figure 2. Input floor plan(left), segmented floor plan(right)

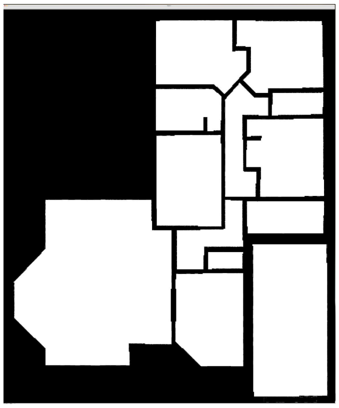


Figure 3. Convert a segmented floor plan to a black-white contour image

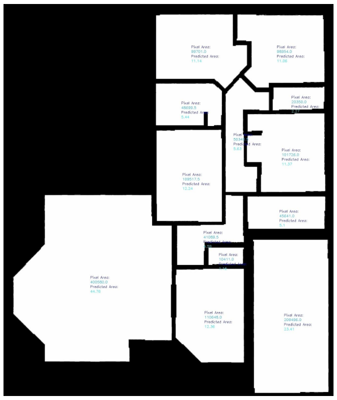


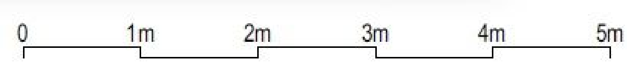
Figure 4. Print all room areas in square meter

| Unit $m^2$ | Living Room | Kitchen | Bedroom 1 | Bedroom 2 | ... |
|---|---|---|---|---|---|
| GT Area | 44.81 | 12.61 | 11.88 | 11.63 | ... |
| Predicted Area | 44.76 | 12.24 | 11.37 | 11.06 | ... |

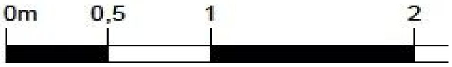
Table 3. Ground truth areas V.S Predicted areas for main rooms

## 6. Next Steps

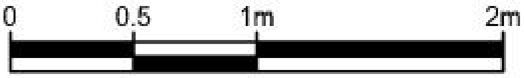
There are mainly three types of scaling legend in the given floor plans:



Type 1 Oxygen Building.



Type 2 EKO Building.



Type 3 Equation Building.

We would like to develop a generic model, which can detect all types of scaling legend, so that we can read the scaling constant for all floor plans. The pipeline of calculating the habitable areas of a floor plan: 1. Convert the pdf floor plan to a JPG image file. 2. Segment the image floor plan to a segmented floor plan (Fig 2. right). 3. Read the scaling legend in the floor plan. 4. Calculate each room's area using the scaling constant.

After completely implement the surface area calculator, we plan to concentrate on detecting the type of each room. More concretely, we decide to create some features for each room and train a softmax neural network model. In addition, we would like to try to detect the text of room type name, which appears on the given image floor plan, by using Google Vision API. Then, these texts are mapping on the floor plan based on the text location, so that we can find the type of each room.

## 7. Timeline and Roles

| Task | Deadline | Leader |
|---|---|---|
| Implement the contour detection model | 11/06/2018 | Qitong Wang |
| Detect each room and calculate its surface area | 11/14/2018 | Chi Zhang, Kaihong Wang |
| Create a robust detector to read all types of scaling legend | 11/21/2018 | Kaihong Wang |
| Using CNN model to detect the type of each room, by extracting the word from the given room floor plan | 11/21/2018 | Kaihong Wang, Qitong Wang |
| Using CNN model to detect the type of each room, by scanning the objects and other features in each room | 12/01/2018 | Jiangshan Luo, Chi Zhang |
| Prepare the report and presentation | 12/06/2018 | all |

Leaders are responsible for their tasks, however, others may also contribute to these tasks.

## References

1. *Liu, C., Wu, J., Kohli, P., & Furukawa, Y. (2017). Raster-to-Vector: Revisiting Floorplan Transformation. 2017 IEEE International Conference on Computer Vision (ICCV). doi:10.1109/iccv.2017.241*
2. *Dodge, S., Xu, J., & Stenger, B. (2017). Parsing floor plan images. 2017 Fifteenth IAPR International Conference on Machine Vision Applications (MVA). doi:10.23919/mva.2017.7986875*
3. *Ahmed, S., Liwicki, M., Weber, M., & Dengel, A. (2011). Improved Automatic Analysis of Architectural Floor Plans. 2011 International Conference on Document Analysis and Recognition. doi:10.1109/icdar.2011.177*
4. *Ahmed, S., Liwicki, M., Weber, M., & Dengel, A. (2012). Automatic Room Detection and Room Labeling from Architectural Floor Plans. 2012 10th IAPR International Workshop on Document Analysis Systems. doi:10.1109/das.2012.22*
5. *Image Filtering¶. (n.d.). Retrieved October 29, 2018, from https://docs.opencv.org/2.4/modules/imgproc/doc/filtering.html?highlight=erode#erode*
6. *Softmax Regression. (n.d.). Retrieved October 29, 2018, from*

*http://ufldl.stanford.edu/wiki/index.php/Softmax_Regression*

7. *"Vision API - Image Content Analysis | Cloud Vision API | Google Cloud," Google. [Online]. Available: https://cloud.google.com/vision/. [Accessed: 14-Nov-2018].*

## Appendix

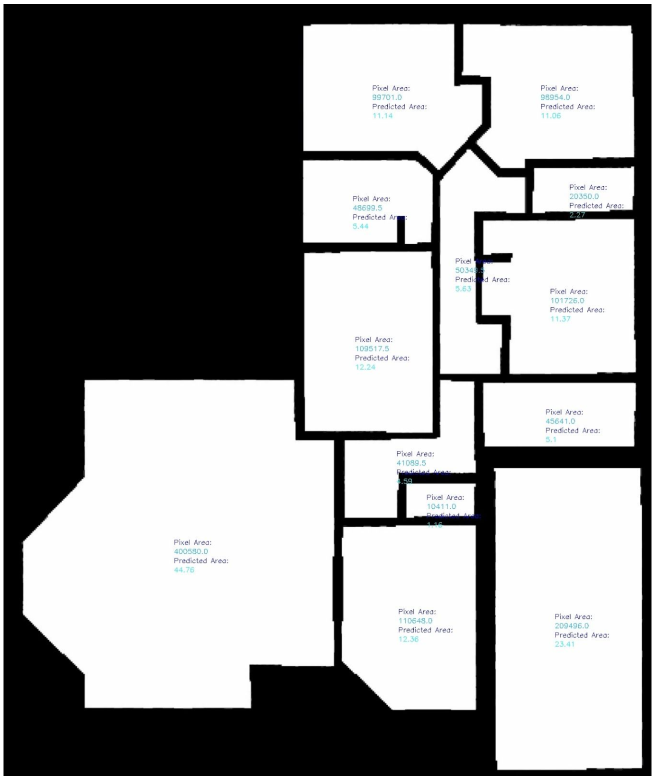


Figure 4.