

# Project - Final Presentation

---

14th of June 2019

Grégoire Clément

# Summary

- I. What are the **MOTIVATIONS**
  - II. **HOW** we tackle the problem
  - III. What are the final **RESULTS**
  - IV. Further **IMPROVEMENTS**
-

# I. Motivations

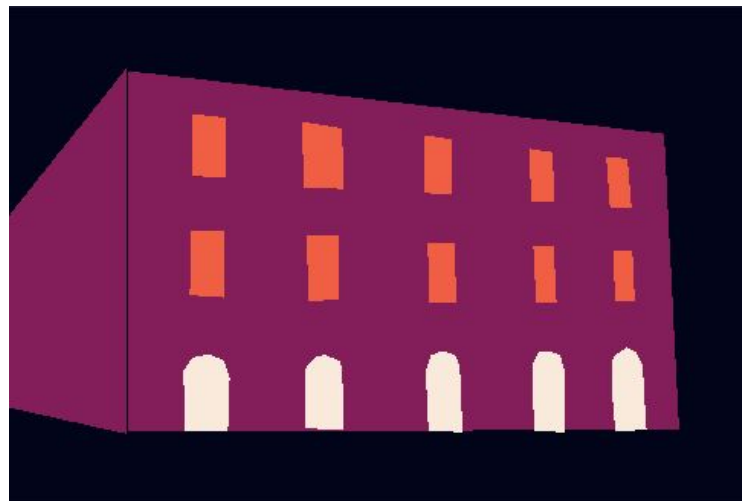
One part of a bigger project in collaboration with civil engineers.

After an **earthquake**, we need to **evaluate damage** fast.

We want to provide faster and easier **facade-parsing** to civil engineers.



# I. Motivations



semantic segmentation vs polygons

## II. How - Data

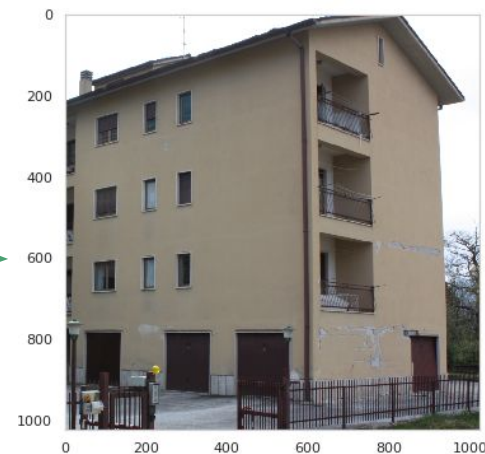
- 418 labeled images
- 4 labels

{ wall, window, door, ~~object~~, background }



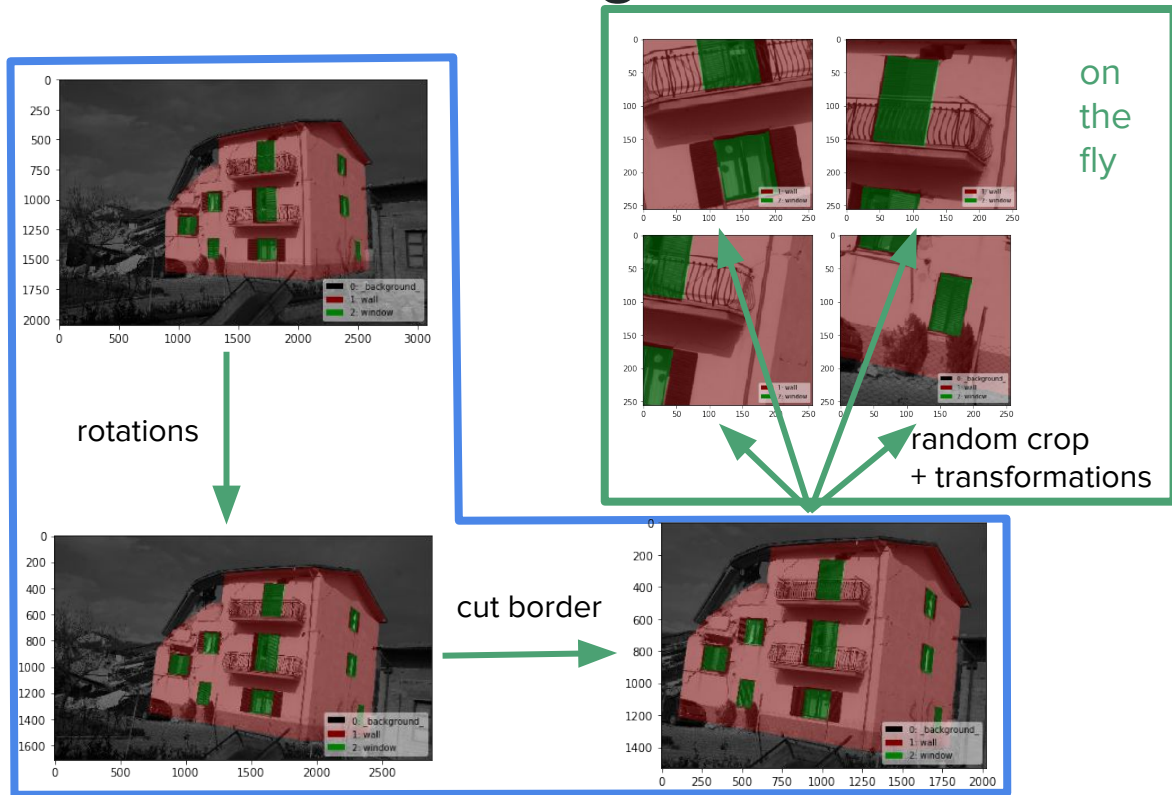
## II. How - Issues

- labels quality
- consistency

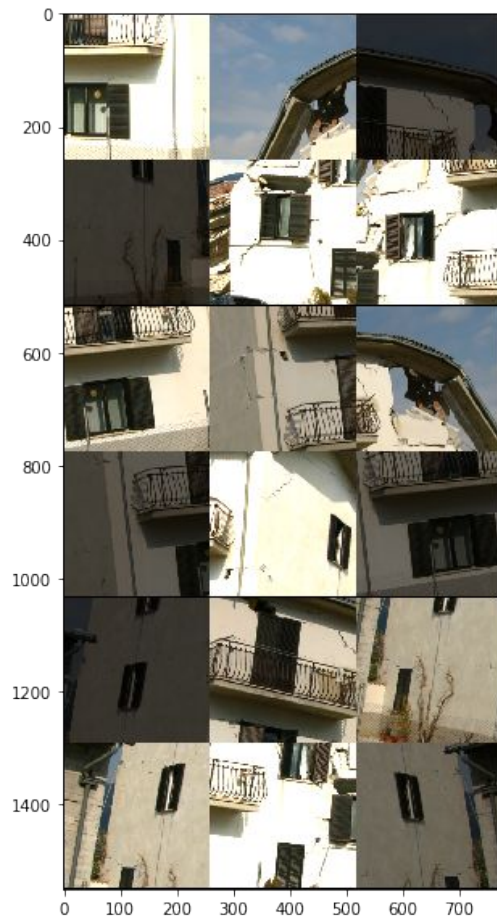




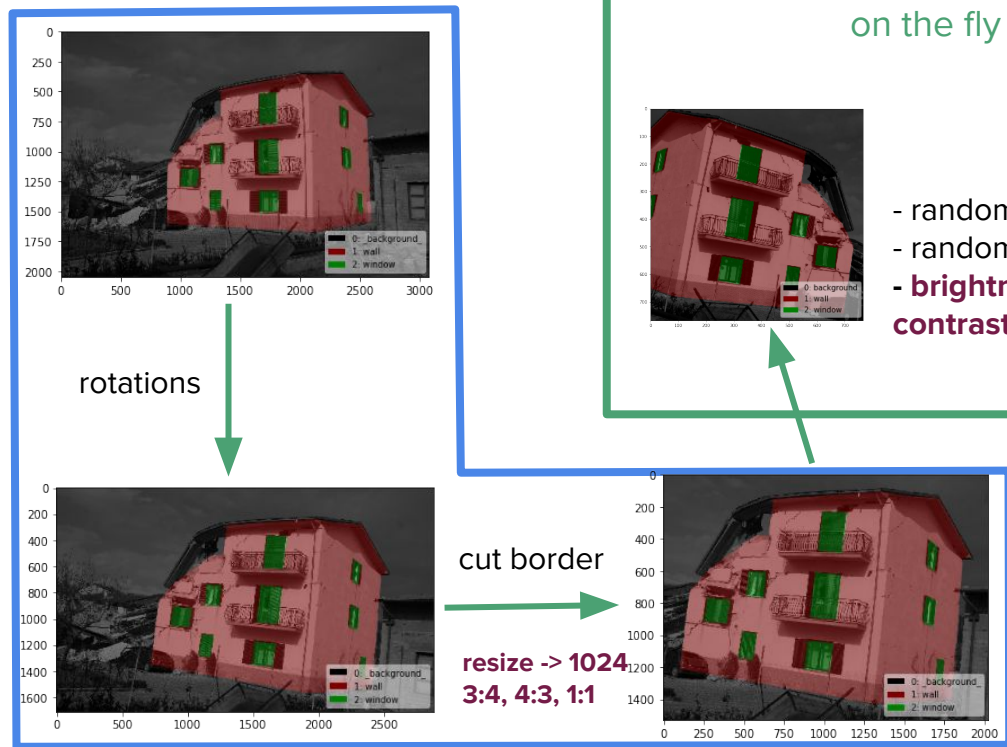
## II. How - Data Augmentation



done once and saved on disk



## II. How - Data Augmentation v2



on the fly

- random crop (**768**)
- random flip
- **brightness and contrast changes**

batches of size 4



done once and saved on disk

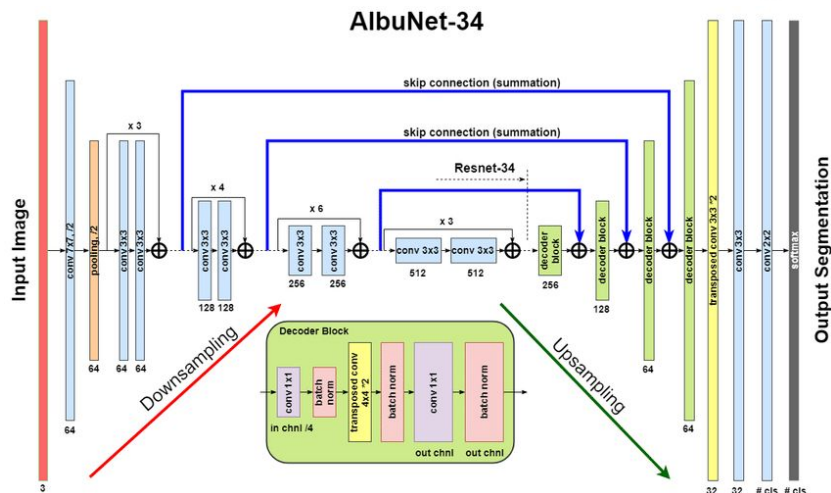
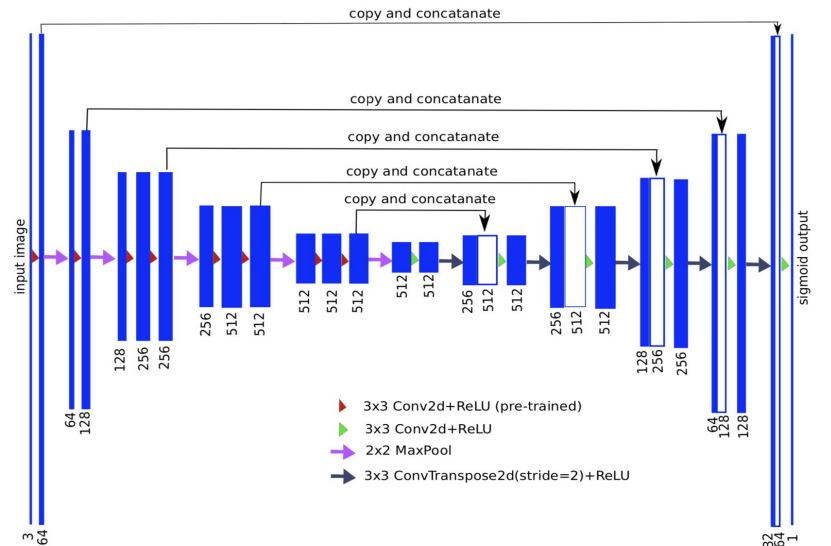


## II. How - Which Models

- An U-Net based model

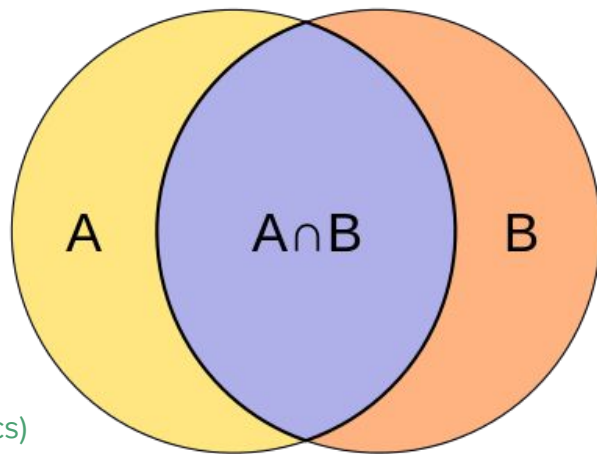
“AlbuNet”

- Fully convolutional
- using pretrained ResNet as an encoder



### III. Results - Losses and metric

- Jaccard Index
- Generalized Dice Loss (No need to compute statistics)
- (Weighted Cross-Entropy Loss)



$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$

$$L_D = \frac{2|A \cap B|}{|A| + |B|} = \frac{2 \sum_i^N \hat{Y}_i Y_i}{\sum_i^N \hat{Y}_i^2 + \sum_i^N Y_i^2}$$

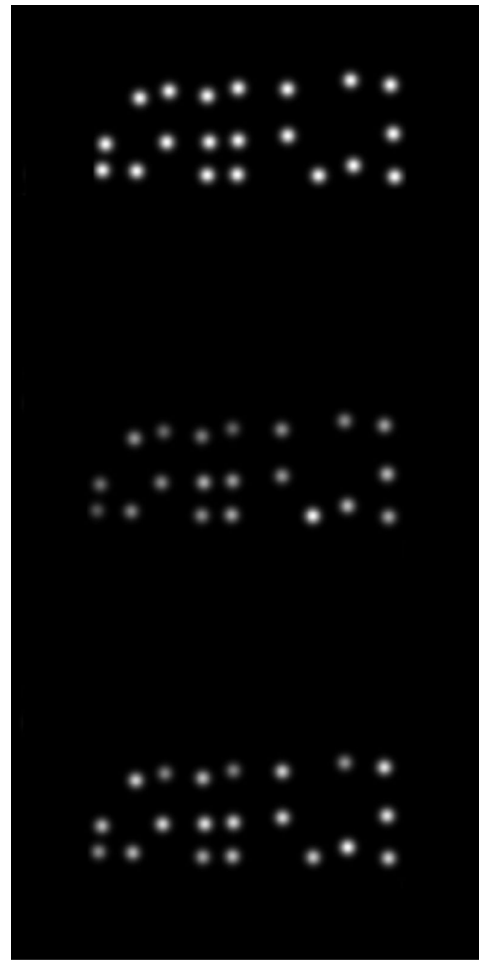
# What's next ?

- Going **from images to the polygons** of the walls/windows/doors directly. How ?

# Heatmaps

- Center, width, height heatmaps for doors and windows
- Segmentation & Regression
- ( + ) Multi-task learning
- ( - ) only rectangles

$$L_F = \alpha L_D + \beta L_{MSE}$$



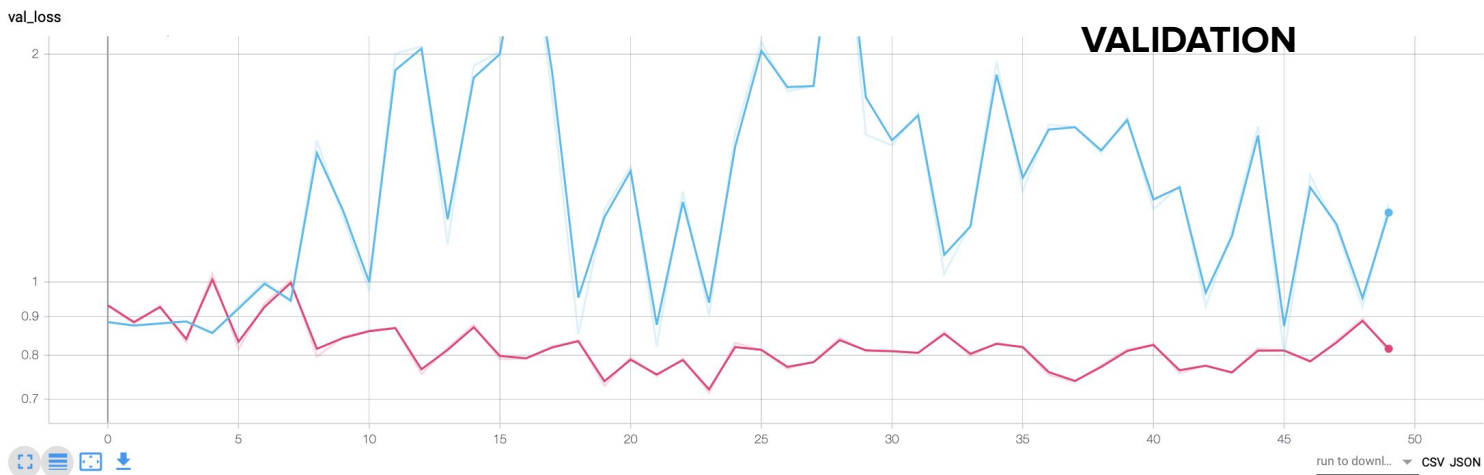
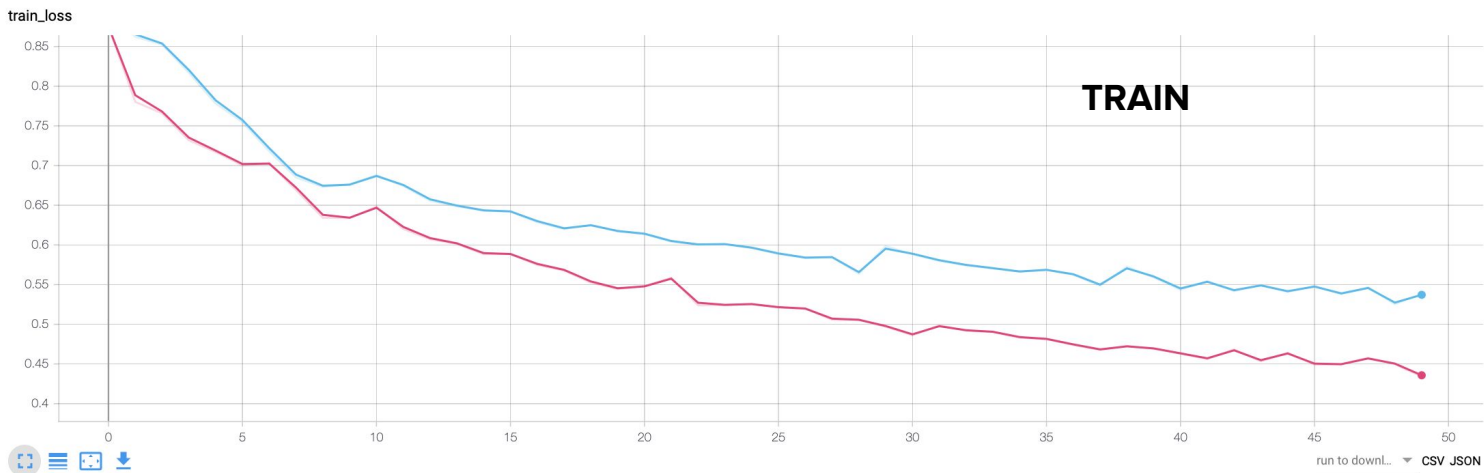
# Losses

alpha = 1

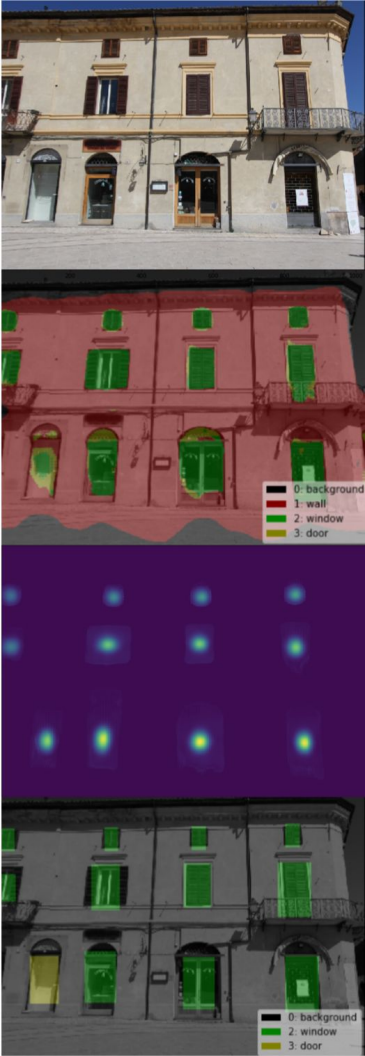
beta = 0.002

UNET

ALBUNET







Input

Segmentation  
mask

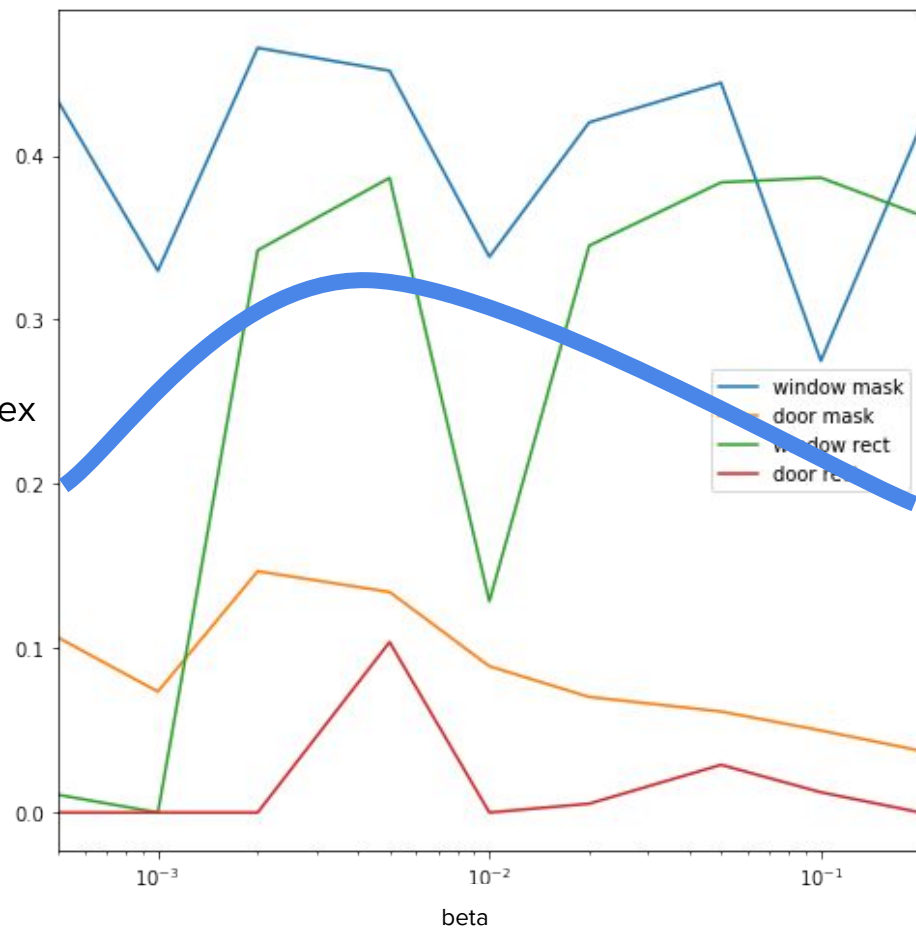
Heatmap  
(center)

Prediction from  
heatmaps

?

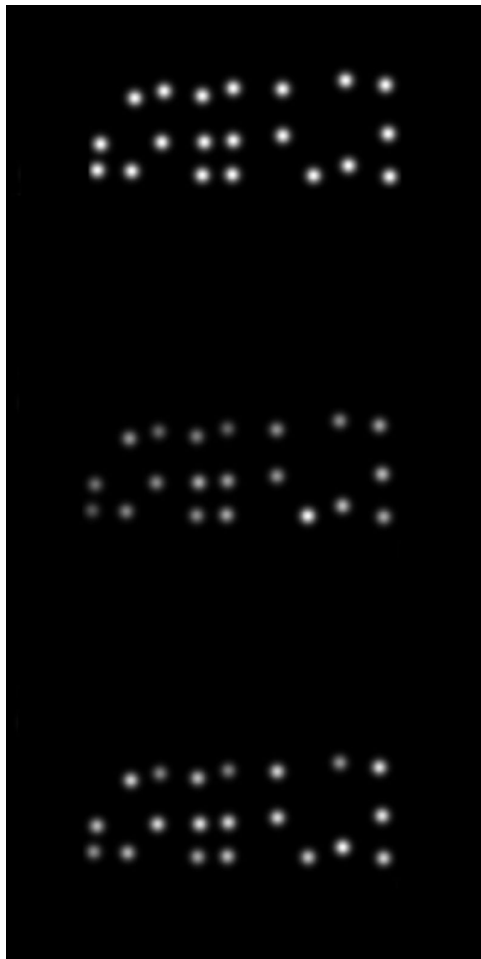
jacc index

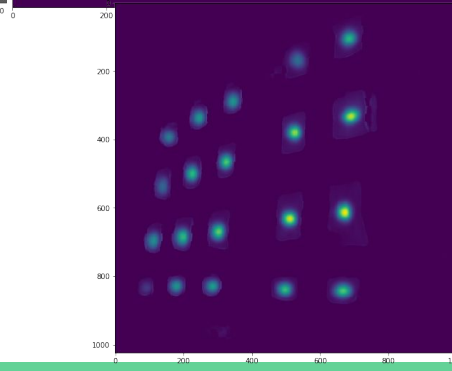
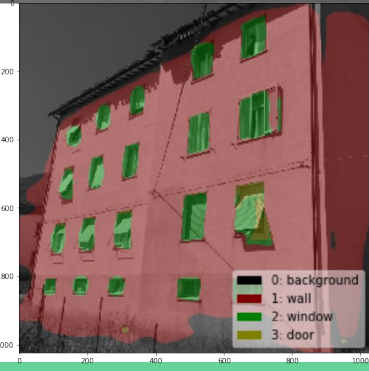
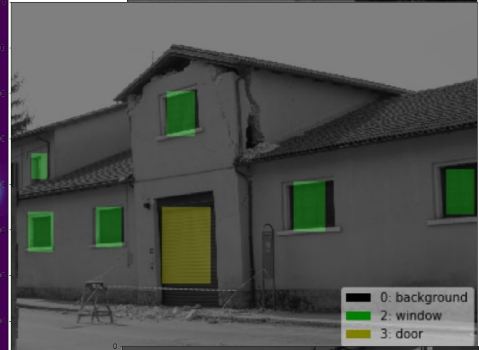
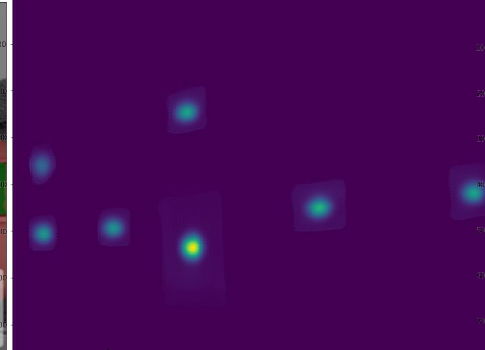
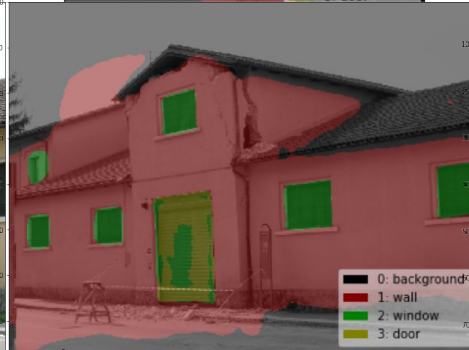
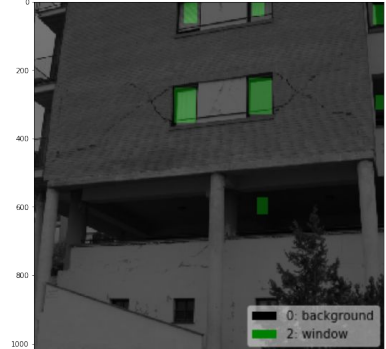
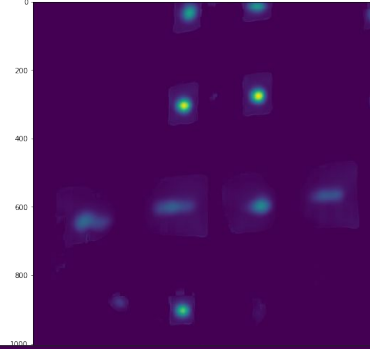
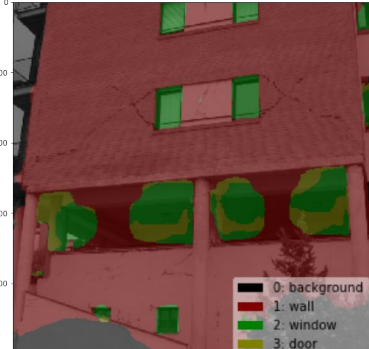
alpha = 1



# From heatmaps to mask

1. Threshold the center map
2. Find the connected components
3. Compute their respective center of mass
4. Use the center of mass to get the width & height  
on the remaining two heatmaps → Polygon
5. Use the segmentation mask to get the label of  
the polygon



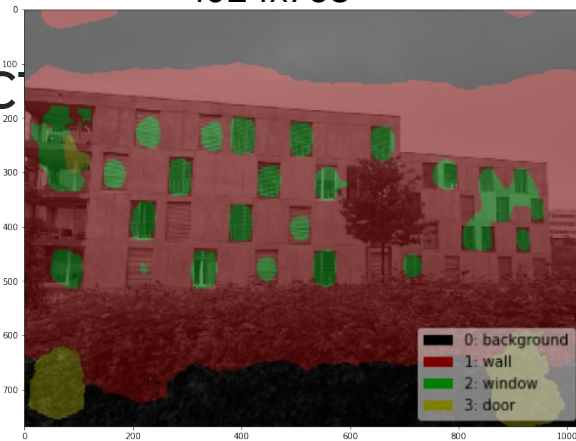


# 'Read world' predic

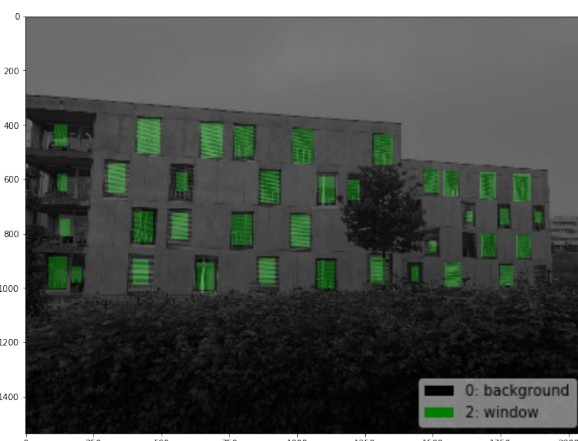
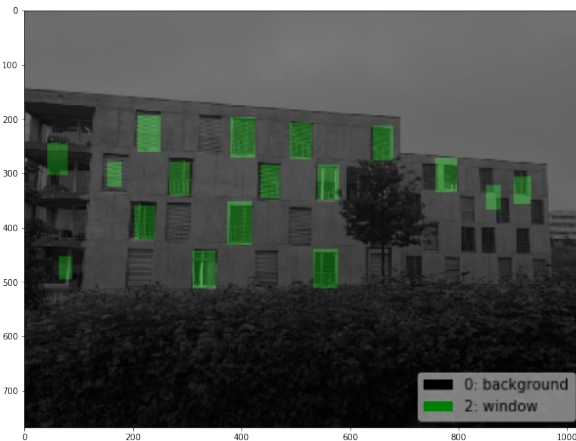
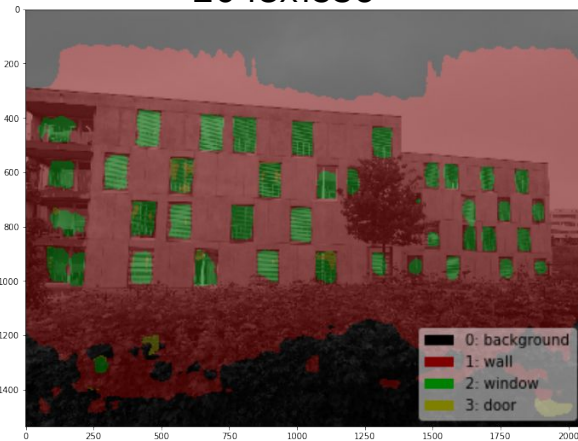


Les Estudiantines -  
St-Sulpice

1024x768



2048x1536



# Further Improvements

- Scaling

Multiple scales predictions or more data augmentation

- Better maxima detection (heatmaps)

Extract maxima with a simple convolution filter

- Perspective deformation

Find it → Compensate for it → Prediction



# Additional Notes

```
[23]: from facade_project.data import TransformedDataset
      from facade_project.geometry.heatmap import info_to_mask
      from facade_project.data.facade_random_rot_dataset import add_heatmaps_target
      from facade_project.data.augmentation import random_brightness_and_contrast, random_
```

- Reusable & modular code base → python library
- Well defined development environment:

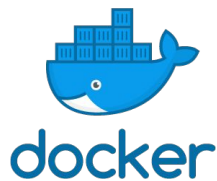


image with all the libs and using CUDA

# This is end,

my only friend, the end. [the doors] 

---

14th of June 2019

Grégoire Clément

<https://github.com/gregunz/FacadeParsing>