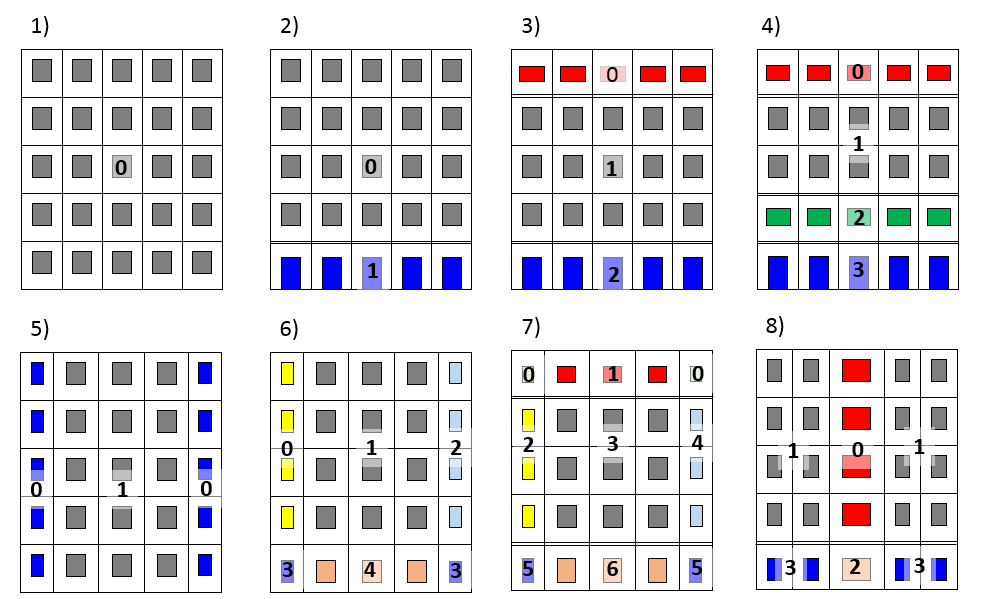
# Façade Grammar



# Update Plan

1. Floor CNNを使って階数、カラム数を推定した後、実際のsplitラインの推定のアルゴリズムを改善する。Efficient structured parsing of facades using dynamic programming by Cohen et al. 2014のアイデアを使えないか？
2. 窓の位置の検出について、もう少し改善したい。特にガラス部分のみを抽出すれば改善されるのではないか？あるいは、GAN論文に書いてたような、ネットワークの修正を行えば、CNNでもっと改善する？  
   Radford et al. adopted generative adversarial networks (GAN)  for the deep CNNs, and proposed several key ideas how to make the training stable.
3. Mass parameter のrefinementで、800x800（？）で実施すると速いが、フルスクリーンだと遅い。なので、フルスクリーンの場合でも、レンダリングサイズを800x800として、高速化したい。
4. 1Fと2Fで、窓のrecognition CNNを分けたい。また、recognitionの精度を上げたい。
5. また、1Fでドアなら、強制的にbottomのサイズを0にしたい。  
   ⇒対応済み。

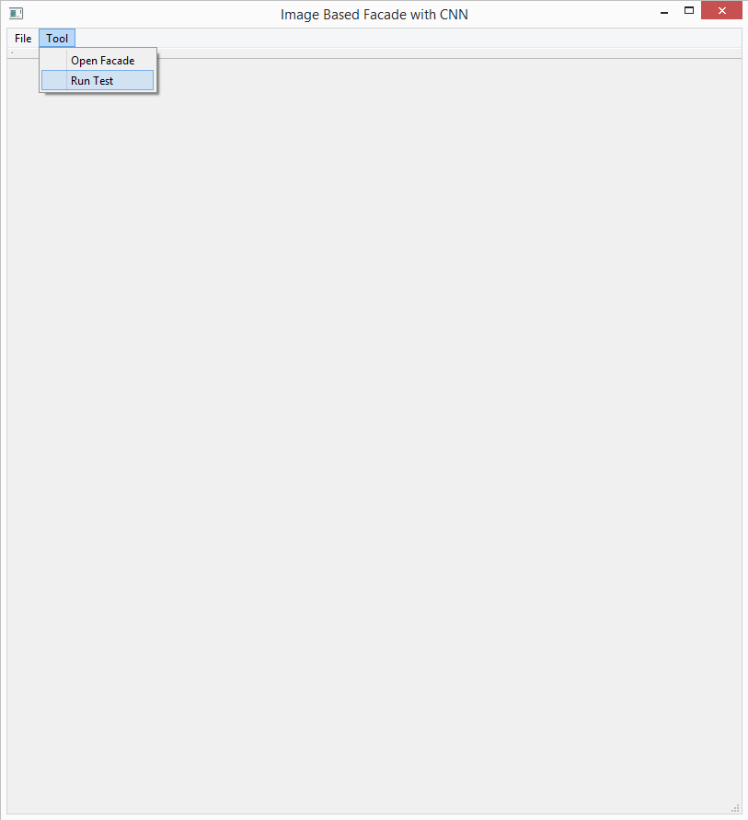
# Façade Simplification

VSプロジェクト：*ImageBasedFacadeWithCNN*

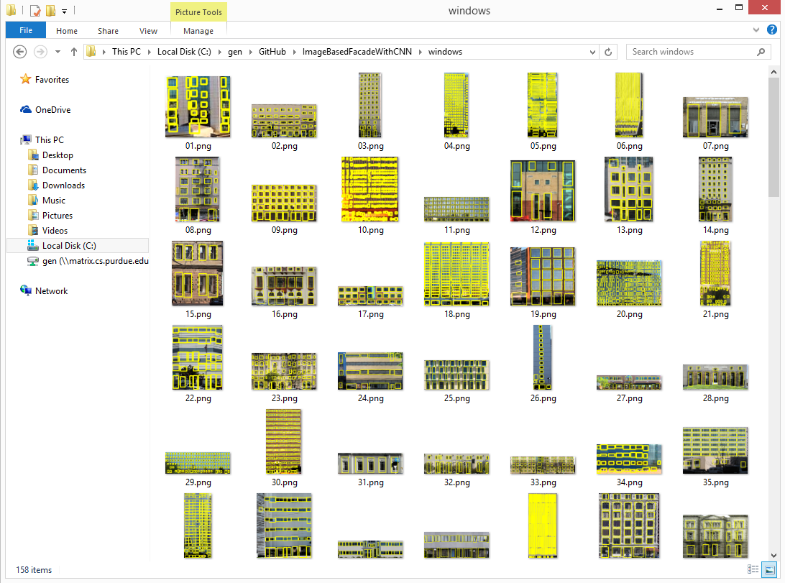
本プロジェクトは、Façade simplificationステップを実装したものである。この実装は、FaçadeCNNの第一ステップとして使用されている。

## [Tool]->[Run Test]

facadeフォルダ内にある画像をsimplifyする。

****

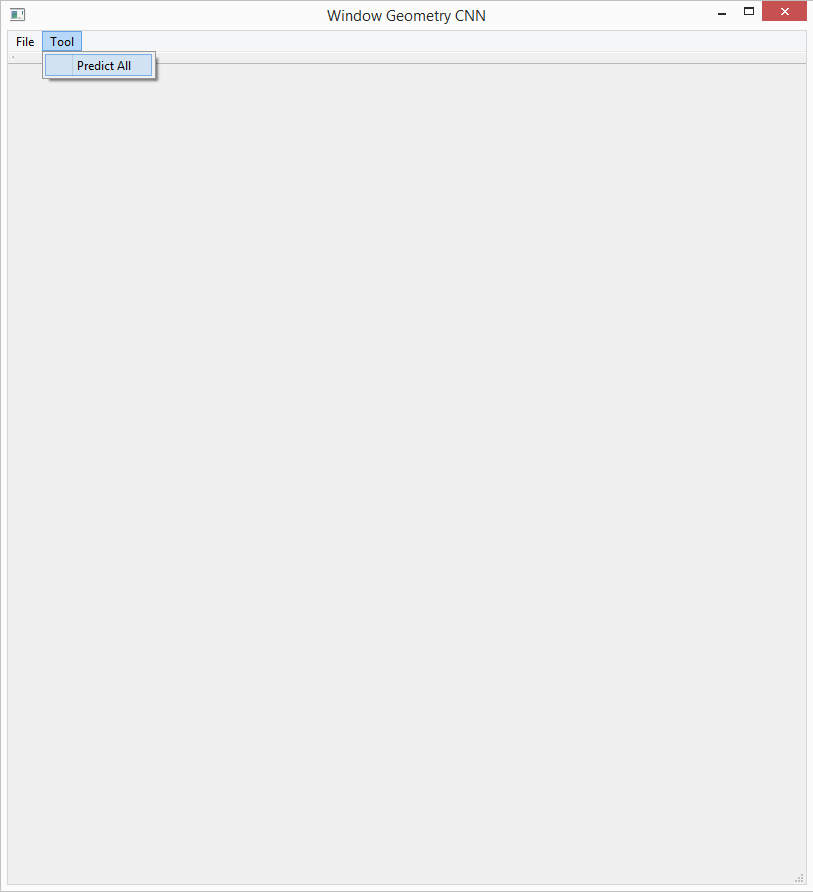
Outputは、windowsフォルダ内に、検知されたwindowが表示された画像が保存される。



# Window Grammar Recognition

VSプロジェクト : FacadeImageCNN/WindowImageCNN

本プロジェクトは、FacadeImageCNN/tiles/フォルダ内のタイルイメージをinputとし、CNNでwindow タイプを識別する。識別結果は、results.txtファイルに保存される。また、accuracyが標準出力に表示される。





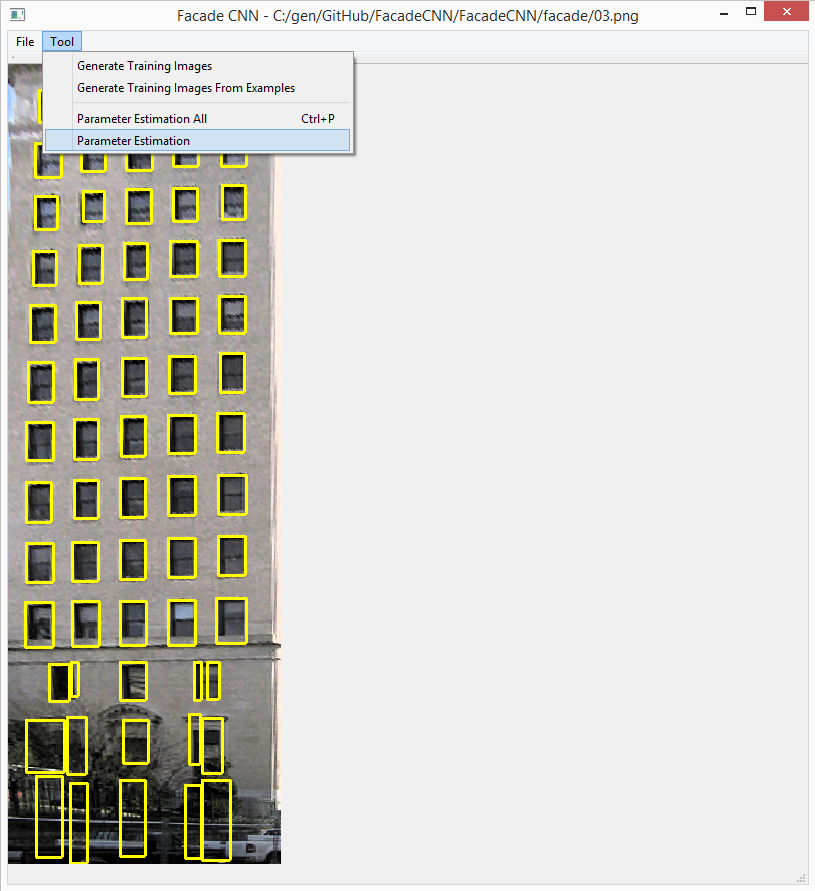
# Façade Grammar Generation

VSプロジェクト: *FacadeCNN*

本プロジェクトは、*ImageBasedFacadeWithCNN*に、façade grammar generation機能を追加したものである。

## [Tool]-> [Parameter Estimation]

façade画像を選択し、それの窓を検知する。

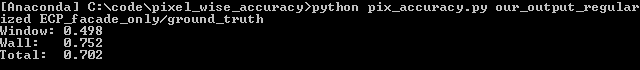


アウトプットファイルは、

|  |  |
| --- | --- |
| ファイル名 | 説明 |
| FacadeCNN/façade\_subdivision.png | Façadeの分割線を表示した画像 |
| FacadeCNN/result.png | Façade grammarが生成する画像 |
| FacadeCNN/window.png | 検知されたwindowの位置を表示した画像 |
| FacadeCNN/window227.png | 上の画像を227x227に縮尺した画像 |

# Pixel-Wise Façade Parsing Accuracy

C:\code\pixel\_wise\_accuracy\フォルダで、以下のコマンドを実行する。



Window、Wall、Totalのclassification accuracyが出力される。

なお、color codeは、Teboul論文に準拠。ただし、WindowとWall以外のラベルは無視してaccuracyを計算する。