# Model documentation and write-up

You can respond to these questions either in an e-mail or as an attached file (any common document format is acceptable such as plain text, PDF, DOCX, etc.) **Please number your responses.**

1. Who are you (mini-bio) and what do you do professionally?

I graduated as a biomedical engineer, and currently employed as an image processing software engineer in a medical startup company located in Los Angeles, California. My job requires me to design computer vision algorithms to automatically analysis and quantify medical X-ray images. Lately, I found that deep learning can perform better computer vision task with higher accuracy and higher robustness. Therefore, I started to invest some of my time to learning deep learning and begin my journey in the deep learning.

|  |
| --- |
| **If you are on a team, please complete this block for each member of the team.** |

1. High level summary of your approach: what did you do and why?

First I set this problem as a classification problem. The goal to is classify the materials of the roof as more accurate as possible.

For data preparation, I segment out each roofs and save into individual images with their mask. The reason to have a mask file is to let the model focus more on the predicting roof.

For training, I used transfer learning of resnet18, resnet50, and resnet101. The final submission file is the ensembled average of the result from the three models.

1. Copy and paste the 3 most impactful parts of your code and explain what each does and how it helped your model.

Part 1. At the first, I’m using matlab’s big image functions to crop out the individual roof images, and I was not getting a good result using the images. After a exhausting long search, I found that matlab’s big image functions is not able to accurately map the coordinates in the geojson file to the tiff image. Therefore causing the cropped image to be totally missing the target roof. After some research, I managed to use geopands and rasterio to accurately map the roof coordinated onto the roof, which greatly increase the accuracy of the prediction.

*with rasterio.open(fpath\_tiff) as tiff:*

*tiff\_crs = tiff.crs.data*

*df\_roof\_geometries['projected\_geometry'] = (df\_roof\_geometries['geometry'].to\_crs(tiff\_crs))*

Part 2. By using customized random rigid transformation and controlled shear during training, the model is able to train for more epochs without showing signs of over fitting. This should improve the model’s robustness.

%% Rotation

tform = randomAffine2d('Rotation',[-180 180]);

outputView = affineOutputView(size(image),tform,'BoundsStyle','FollowOutput');

image\_augmented = imwarp(image,tform,'OutputView',outputView);

outputView\_mask = affineOutputView(size(image\_mask),tform,'BoundsStyle','FollowOutput');

mask\_augmented = imwarp(image\_mask,tform,'OutputView',outputView\_mask);

%% Reflection

tform = randomAffine2d('XReflection',true,'YReflection',true);

outputView = affineOutputView(size(image\_augmented),tform);

image\_augmented = imwarp(image\_augmented,tform,'OutputView',outputView);

outputView\_mask = affineOutputView(size(mask\_augmented),tform,'BoundsStyle','FollowOutput');

mask\_augmented = imwarp(mask\_augmented,tform,'OutputView',outputView\_mask);

%% Shear

tform = randomAffine2d('XShear',[-5 5]);

outputView = affineOutputView(size(image\_augmented),tform,'BoundsStyle','FollowOutput');

image\_augmented = imwarp(image\_augmented,tform,'OutputView',outputView);

outputView\_mask = affineOutputView(size(mask\_augmented),tform,'BoundsStyle','FollowOutput');

mask\_augmented = imwarp(mask\_augmented,tform,'OutputView',outputView\_mask);

%% Final result

[row\_list, col\_list] = find(mask\_augmented > 0);

image\_augmented = image\_augmented(min(row\_list):max(row\_list), min(col\_list):max(col\_list), :);

mask\_augmented = mask\_augmented(min(row\_list):max(row\_list), min(col\_list):max(col\_list));

Part 3. I tried using transfer learning with resnet18, resnet50 and resnet101. However, it shows that by increase the complexity of the network, doesn’t necessarily increase the result. After reading some articles about model ensemble, I use a simple average of the result from all three networks and is able perform better than any network individually.

%% Ensemble the data by simply finding the average

T1 = readtable('resnet18\_result.csv');

T2 = readtable('resnet50\_result.csv');

T3 = readtable('resnet101\_result.csv');

T\_final = T1;

T\_final.concrete\_cement = (T1.concrete\_cement + T2.concrete\_cement + T3.concrete\_cement)/3;

T\_final.healthy\_metal = (T1.healthy\_metal + T2.healthy\_metal + T3.healthy\_metal)/3;

T\_final.incomplete = (T1.incomplete + T2.incomplete + T3.incomplete)/3;

T\_final.irregular\_metal = (T1.irregular\_metal + T2.irregular\_metal + T3.irregular\_metal)/3;

T\_final.other = (T1.other + T2.other + T3.other)/3;

writetable(T\_final,'submission.csv');

1. What are some other things you tried that didn’t necessarily make it into the final workflow (quick overview)?

At the very beginning, I noticed that the training dataset is extremely unbalanced. The dataset is mainly consist of healthy metal and irregular metal.

I tried to augmented the data using brightness, color shift, other than the more basic shape transformation. However, the augmentation did not increase the accuracy. The suspected reason is that the augmented roof image doesn’t necessarily represent the type of roofs in the testing dataset.

1. Did you use any tools for data preparation or exploratory data analysis that aren’t listed in your code submission?

No

1. How did you evaluate performance of the model other than the provided metric, if at all?

Other than the accuracy, I looked at 5 more performance indicators of the model on the validation dataset, which are ROC curve, sensitivities, specificities, PPV and NPV.

1. Anything we should watch out for or be aware of in using your model (e.g. code quirks, memory requirements, numerical stability issues, etc.)?

Training from transfer learning should stopped around 8-10 epochs to prevent overfitting.

1. Do you have any useful charts, graphs, or visualizations from the process?

No

1. If you were to continue working on this problem for the next year, what methods or techniques might you try in order to build on your work so far? Are there other fields or features you felt would have been very helpful to have?

Firstly, I will look into ways to finding the effective augmentation method that should represent the difference of the roof material and distributions of the roof material in terms of area, location and etc.

Secondly, I might also look into finding ways to extract the texture of the roof, by isolate the roof texture, therefore remove a lot of the noises in the data.

Fields like household income, postcode and population density might also be some interesting factor in classifying the roof material.