Classifying Roof Material From Drone Imagery
An Approach to the Open AI Caribbean Challenge

Johannes Leonhard Rüther

10. Januar 2020

1 Introduction (Context and Challenges)

1.1 Context

Regions like the Carribean are regularly hit by rainstorms, floods or earthquakes. Despite being so prone, many houses in those areas are unable to withstand these natural hazards due to poor construction quality. This exposes their inhabitants to a great risk of becoming homeless during the next natural disaster.

International programs such as the World Bank's Global Program for Resilient Housing are making attempts to retrofit houses to the natural forces they are exposed to. In these large and often informal settlements it is difficult to assess which houses pose especially high risks due to their construction or are damaged and need repair. Exploring these areas on the ground is time-consuming and costly. This is why the possibilities of image processing for automatic recognition of vulnerable houses on the basis of drone imagery are explored. Such a technology could assist building inspectors and narrow down large areas to those that are worth a closer inspection on the ground. The material that a roof is made up of is a central indicator for how well a house is prepared against natural disasters. Therefore, classifying roof material from aerial images is a key step to identify precarious houses.

1.2 Open AI Challenge

The above background led to the initiation of the *Open AI Caribbean Challenge: Mapping Disaster Risk from Aerial Imagery*, which was conducted between October and December 2019 on *drivendata.org*. This report describes an approach to solve this challenge.

1.3 Previous Work

In many applications, the identification of roofs is considered useful. Roof segmentation is often done with LiDAR data, as presented in [1]. Other papers such as [2] have successfully attempted roof segmentation using only drone imagery, which is less costly. This step will at some point become relevant for the task at hand. The approach discussed in this report however uses images of roofs that have already been segmented.

The identification of roof defects has been addressed in previous works, e.g. [3], in which water stagnation on roofs was measured. Multiple patents (e.g. [4]) describe the use of aerial images to evaluate damage on individual roofs for insurance purposes. To my best knowledge, academic works on roof material and condition classification from drone imagery on a large scale have however not been published. Similar processing routines using pretrained neural networks have been widely used for other classification tasks, the probably most similar presented in [5].

Abbildung 1: Thumbnail of stitched drone image from Dennery, St.Lucia.

2 Data Description

2.1 Images

The data provided for the challenge consists of high resolution ($\sim$ 4cm) drone imagery of five patches of land: two from Soacha, Colombia, two from Mixco, Guatemala and one from Dennery, St. Lucia. For every region, there is one stitched cloud-optimized GeoTIFF file, ranging from 500 to 1800 Megapixels in size. A thumbnail of the Dennery settlement is shown as example in Fig. 1.
General information about the images is summarized in table 1.

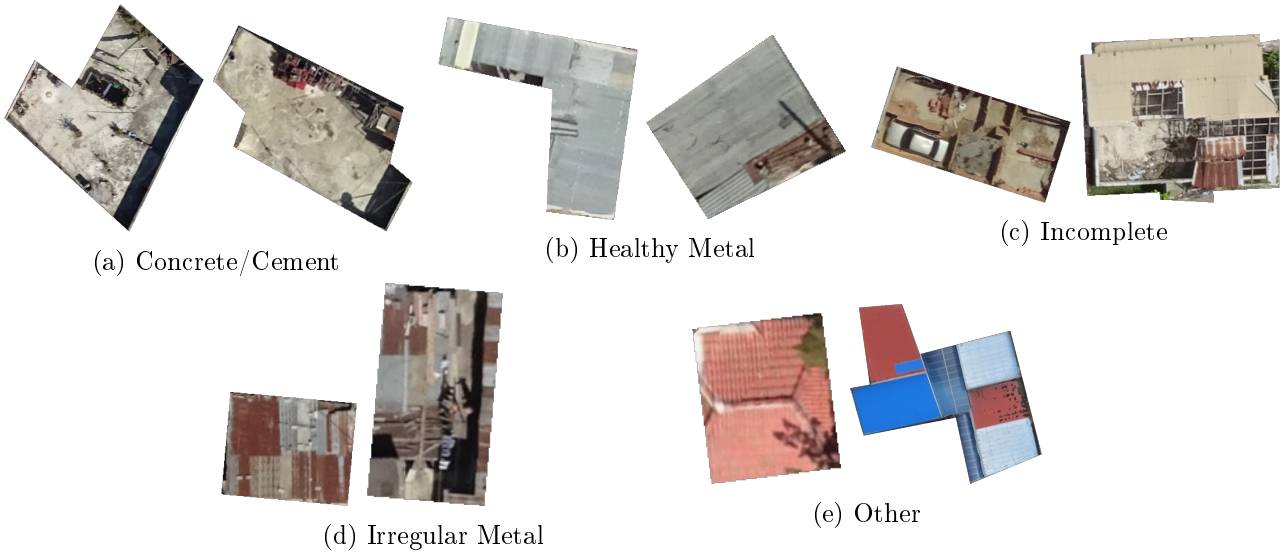
Platform	WeRobotics (private drone)
Source	DrivenData Competition https://www.drivendata.org/competitions/58/disaster-response-roof-type/data/
Acquisition Method	Drone Photography
SRS	Ellipsoid (EPSG:32616, 32618, 326120)
Spatial/Spectral resolution	3.8-4.5cm, RGB
Format	Cloud-optimized GeoTIFF

Tabelle 1: General information about provided data.

2.2 Labels

Roofs are labeled as one of five classes, examples of which are given in Fig. 2. The footprints of roofs are provided for both training and test set, i.e. roof segmentations are always given. There is a total of 14852 training examples with highly unbalanced classes. The log-loss as score for the *Open AI Challenge* was computed on 7320 test images, for which class membership probabilities needed to be calculated.

1. Concrete and Cement (1385 training samples): Roofs made out of concrete or cement.
2. Healthy Metal (7370): Roofs of metal that are intact but may be corrugated or galvanized.
3. Incomplete (668): Roofs that are severely damaged or under construction.
4. Irregular Metal (5236): Roofs that are slightly damaged, rusted or patched.
5. Other (193): Roofs that do not fit into other categories (include tiles, red painted, other materials).



(a) Concrete/Cement (b) Healthy Metal (c) Incomplete

(d) Irregular Metal (e) Other

Abbildung 2: Example images for each material class. (Scales differ)

2.3 Label Noise

Unfortunately, class membership is sometimes ambiguous in the provided groundtruth data. First, labels are used inconsistently. This is especially true for ïrregularänd "healthy metal". Where rusty roofs should be labeled as ïrregular", many of them carry a "healthy metal"label. Moreover there is confusion about properties that make a roof ïrregularör ïncomplete". These inconsistencies might stem from different annotators labeling images.

Second, labels are sometimes clearly incorrect, e.g. concrete roofs are labeled as "healthy metal". In that regard, some regions seem to be annotated with more care than others.

It is impossible to quantify the extent to which annotations are noisy. Some examples of these ambiguities are shown in Fig. 3. The implications for the results are discussed in section 5.

3 Proposed Processing Routine

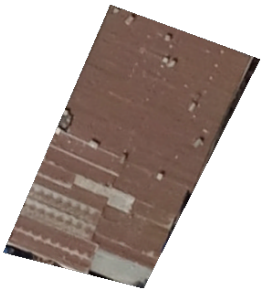
3.1 Pixel-based Baseline

As a baseline for comparison, several statistical metrics were extracted as "naive"features and classified using a Support Vector Machine (SVM) or Random Forest Model (RF). The mean was used to assess the dominant colors on a roof. Heterogeneity was tried to be measured using the standard deviation and value range. Every metric was calculated on every color channel and on the absolute of the Sobel-filtered image in x- and y-direction of every channel (see Fig. 4). This results in a 18-dimensional feature vector (3 metrics $\times$ 3 channels on original and gradient image).

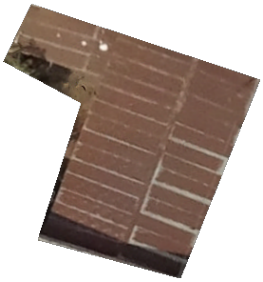
3.2 Petrained Neural Network as Feature Extractor

The proposed processing routine uses a pretrained neural network as feature extractor. Different publicly available architectures, such as ResNet50 [6], InceptionV3 [7], DenseNet201 [8] and VGG16 [9] were tested. The employed architectures were each trained on more than a million images of 1000 categories from the ImageNet database [10].

ImageNet is a collection of natural images of everyday objects and animals. Although roofs are of course different from animals and other objects, natural images share a lot of common features. The features which are relevant to distinguish classes in ImageNet are very likely to be also relevant to discriminate



(a) "healthy metal"



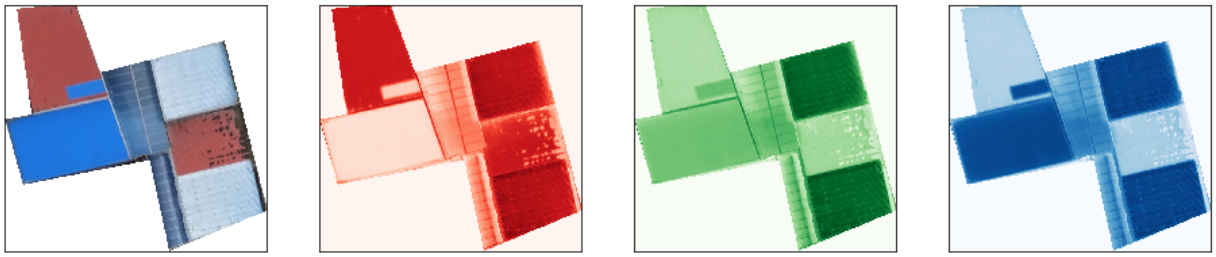
(b) ïrregular metal"

(c) Labeled ïrregular metal"where clearly concrete

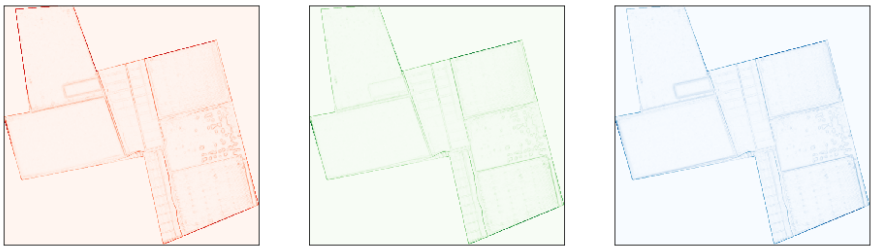
(d) Labeled ïrregular metal"

(e) Labeled ïncomplete"

Abbildung 3: (a) and (b) named differently due to incoherent labeling of rusty roofs. Similarly unclear boundary between ïrregular metaländ ïncomplete"((d) and (e)). Obvious labeling error in (c).



(a) RGB Image and individual color channels



(b) Derivative images (absolute of Sobel-filtered images in x- and y-direction)

Abbildung 4: Original image classified as öther"with red, green and blue individual channels (top row). Gradient images of each channel (bottom row)

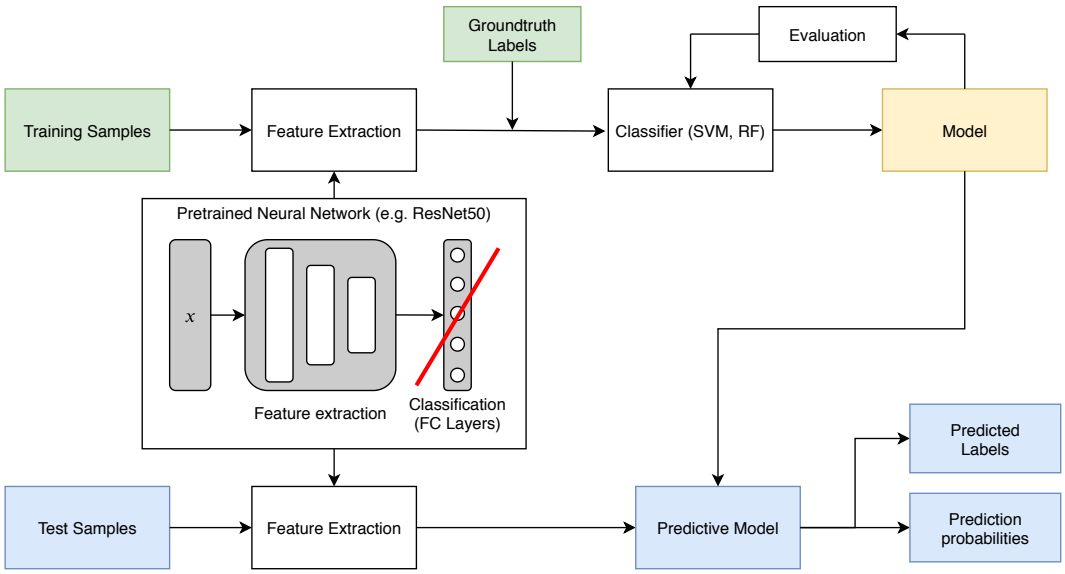


Abbildung 5: Proposed processing routine: The classification part of a pretrained neural network is removed, leaving the feature extraction. A SVM or RF is used to build a model out of the training features and the groundtruth labels in an iterative process. Test features are fed into this model to predict labels and determine prediction confidence.

other types of natural images, such as roofs. In order to obtain features, the classification layer were removed and replaced by an average or maximum pooling layer to reduce the number of features.

In the next step, a a SVM or RF was trained to classify roof materials based on the extracted features. The calculated classification could then be used to predict class membership based on features of test samples. The *scikit-learn* implementation of both SVM and RF provide pseudo-probabilities that take a samples' distance to the decision boundary into account. Those probabilities/confidences were submitted for evaluation on the challenge website. The full classification pipeline is presented in Fig. 5.

4 Results

Before investigating results it has to be clarified that the performance of the suggested system was optimized to perform well in the described *Open AI Challenge*. The predictions to be submitted were class membership probabilities and not hard labels. Since the algorithm is supposed to hint at high risk roofs to be revisited by a human instructor, this seems reasonable. In section 4.1, the system's performance on the challenge test set is presented. Section 4.2 shows results for which the annotated part of the data was used to compare the best-performing system to other approaches using more metrics.

4.1 Online Challenge Performance

For the *Open AI challenge*, system performance was evaluated using the log loss:

$$loss = -\frac{1}{N} \cdot \sum_{i=1}^{N} \sum_{j=1}^{M} y_{ij} \log p_{ij} \tag{1}$$

In that context, N was the number of test samples (=7320), M the number of classes (=5), y_{ij} the true label (i.e. 1 if observation i belonged to class j and 0 otherwise) and p_{ij} the predicted probability that i was of class j.

The best-performing setup consisted of a ResNet50 as feature extractor with average pooling at the end and classification using a SVM with a high regularization parameter $C = 100$. Other pretrained networks performed similarly well, although not quite as good. Results on the test in terms of log-loss are shown in table 2. The top score of 0.5554 corresponded to rank 51 in a field of 1425 competitors (top 4%).

Pretrained model	Log-loss
ResNet50	0.5554
DenseNet201	0.5800
InceptionV3	0.5917
VGG16	0.6152

Tabelle 2: Log-loss on online challenge test set for feature extraction with different pretrained models. Classification was always performed with an SVM using $C = 100$.

4.2 Comparing Performances

Since classification performance measures require groundtruth data, the previously so-called "training set"provided for the challenge was split into new training and test sets for evaluation.

To assess the difficulty of the task and the improvement through features extracted by neural networks, the system performance was compared against the pixel-based approach described in section 3.1. Moreover, the Support Vector Machine as the best-performing classifier was tested against a Random Forest model. For both classification methods, parameters were empirically optimized. For SVM a value of $C = 100$ was found to work well. As for RF, 1000 trees were trained with a maximum depth of 16. Results are summarized in table 3. For every listed approach, evaluation metrics were calculated by cross-validation on five different splits. Besides the log-loss in eq. 1, the average accuracy and average F1-score as mean of per-class score (macro F1-score) were calculated.

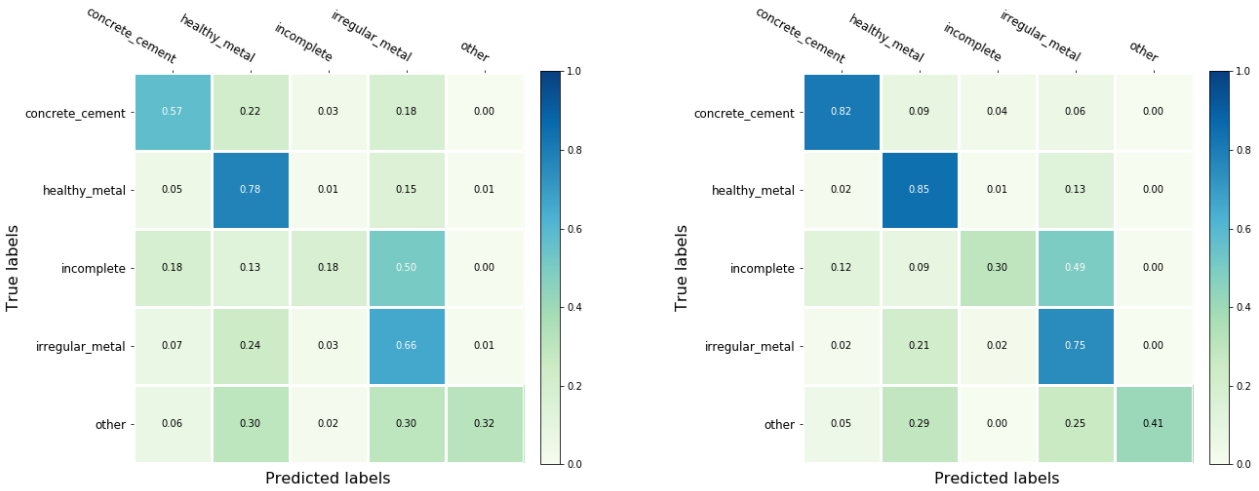
Method	Avg. Log-loss	Avg. Accuracy	Avg. Macro-F1
Pixel-based features + SVM	0.82	0.68	0.49
Pixel-based features + RF	0.80	0.68	0.38
ResNet50 + SVM	0.57	0.76	0.63
ResNet50 + RF	0.70	0.75	0.45

Tabelle 3: Performance metrics of best-performing pixel-based and suggested processing routines. (Note, that a lower log-loss indicates a better result)

The significant improvement in performance from pixel-based to deep features also becomes visible in Fig. 6. It shows the confusion matrices for both classifiers using a random split. When visually assessing the result, the class imbalances should be kept in mind: only 4.5% of the roofs carry the groundtruth label ïncomplete", 1.3% öther". It becomes apparent that both classifiers struggle to identify the least represented classes correctly.

4.3 Removing Noisy Labels

The problem of noisy data was tried to be tackled by selecting clean subsets of training data to improve the accuracy of the model. However in terms of log-loss on the test set, this proved to be counterproductive. It is likely that due to more homogeneous training sets the model became bad at handling noise (wrong or contentious labels) and was punished heavily for "wrong"classifications with high confidence.



(a) Pixel-based features classified by SVM

(b) ResNet50 features classified by SVM

Abbildung 6: Confusion Matrices for pixel-based vs. deep features on random split of the provided data (same split for both).

5 Discussion

5.1 Solving the Real-Life Problem

At first glance, the presented accuracy and confusion measures from section 4.2 might not seem very impressive. It is therefore worth looking at what the system fails to achieve, why that might be and what the implications for the real-life problem at hand are.

1. Ïncomplete"roofs: The recall on ïncomplete"roofs is very low at only 30%. As described in section 2.3, the boundary between ïrregularänd ïncomplete"roofs is hard to define. Both types of roofs carry a higher than normal risk to be destroyed during natural hazards. With respect to that the misclassification of ïncompleteäs ïrregular"might be excusable. The truly problematic cases when considering risk assessment are those 21% that are labeled as "concreteör "healthy metal"when they are actually heavily damaged.

2. Ïrregular metal"roofs: These roofs are classified as "healthy metal"with 21% probability, which is bad for risk assessment. One cause for these errors could be the inconsistent labeling of rusty, but intact looking roofs as described in section 2.3. Interestingly, roofs of ïrregular metaläre very unlikely to be labeled as ïncompleteäs opposed to vice versa. One explanation for this might be that there are eight times more training examples for ïrregular"than for ïncomplete", allowing for a better recall.

3. Öther"roofs: It is unsurprising that this class has a low recall, since it is the most diverse class and has very few training examples. It is reassuring that öther"roofs, which are intact, are almost never labeled as ïncomplete".

When taking these aspects into account, many misclassifications actually seem excusable or at least not so harmful to the actual goal of the project. It should also be considered that the proposed method tried to minimize the log-loss metric. A real-life classification system should have a high recall of high-risk roofs, which might require changes to the pipeline.

5.2 Outlook

Further investigations should look at what the misclassified roofs have in common to improve relevant features. It might be worth to re-annotate parts of the data with more consistent labeling policies.

Since image statistics were surprisingly successful for some classes they might be combined with deep features in order to further enhance the classification result.

In order to extract features that are more relevant to distinguish roofs, the last layers of the pretrained networks should be retrained to better fit the problem. Retraining was attempted during the project but could not be brought to satisfying results in the given time frame.

6 Appendix

The used code is publicly accessible on GitHub, mostly in the form of Jupyter Notebooks (see `https://github.com/jo-ruether/ipeo_carribean`).

Literatur

[1] D. Chen, L. Zhang, J. Li, and R. Liu, "Urban building roof segmentation from airborne lidar point clouds," *International Journal of Remote Sensing*, vol. 33, no. 20, pp. 6497–6515, 2012.

[2] K. Soman, "Rooftop detection using aerial drone imagery," in *Proceedings of the ACM India Joint International Conference on Data Science and Management of Data*, CoDS-COMAD '19, (New York, NY, USA), p. 281–284, Association for Computing Machinery, 2019.

[3] D. Yudin, A. Naumov, A. Dolzhenko, and E. Patrakova, "Software for roof defects recognition on aerial photographs," *Journal of Physics: Conference Series*, vol. 1015, p. 032152, may 2018.

[4] E. A. B. Matthew A. Shreve, "Image segmentation system for verification of property roof damage, us patent 20170352100a1," 2017.

[5] D. Marmanis, M. Datcu, T. Esch, and U. Stilla, "Deep learning earth observation classification using imagenet pretrained networks," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, pp. 1–5, 12 2015.

[6] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," *CoRR*, vol. abs/1512.03385, 2015.

[7] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *CoRR*, vol. abs/1512.00567, 2015.

[8] G. Huang, Z. Liu, and K. Q. Weinberger, "Densely connected convolutional networks," *CoRR*, vol. abs/1608.06993, 2016.

[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014.

[10] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International journal of computer vision*, vol. 115, no. 3, pp. 211–252, 2015.