Mapping Disaster Risk from Aerial Imagery

An OpenAI Carribean Challenge

Deepak Raghavan Balaji, Amit Kumar Jena, Sai Sudhamsa Potru, Abhinayana Madhu, Dr. Kuldeep Chaurasia

***Abstract*—In regions that are prone to natural disasters, it is important that the buildings follow certain construction standards to avoid demolition. One of the factors that predict the risk of damage is the roof material. In this paper, we outline a method to classify buildings on the basis of roof material from aerial drone imagery. The proposed method is an ensemble model of network architectures resnet152, resnext101, efficentnet b5, efficientnet b6 and efficientnet b7 each of which are trained with hyperparameters that are tuned using cyclical learning rate approach . We have obtained a log loss value as low as 0.4373. Therefore, the proposed method can be used to perform an accurate classification of roof material using aerial drone imagery.**

***Index Terms*—deep learning; convoluted neural networks;image classification; resnet; efficient-net; ensemble learning; aerial-imagery; disaster-risk-management; drone imagery; transfer learning.**

I. Introduction

IN areas like the Caribbean with high risk of natural disasters like earthquakes, floods and hurricanes there is a risk of immense damage of life and property if the buildings do not follow modern construction standards. While the buildings can be retrofit to follow these standards, it requires officers to inspect them manually which is a time-consuming, labour-intensive and a very expensive process. Roof material is a particularly relevant characteristic that determines risk, and it is also a predictor of other risk factors like building material. Modern developments in computer vision and machine learning can enable us to accurately classify the buildings according to their roof material which is very useful to building inspectors, to make inspections faster and cheaper, and lets them target resources where they create the most impact. To do this, we have collected a set of aerial drone images of a few regions in the Caribbean and trained several deep convolutional neural networks and compared the results of various models [1].

We have experimented with several Convolutional Neural Network(CNN) architectures for image classification. Some of the models we used were Alexnet, GoogleNet, ResNet, Inception and EfficientNet. The most accurate models were chosen and the final predictions were made by using an Ensemble model to aggregate the six best models.

All the above mentioned models were previously trained on ImageNet dataset. We have used Transfer Learning to train our data on top these pre-trained models.

II. Related Work

Deep learning was explicitly proposed in 2006 by Hinton et al [2]. The basic motivation of deep learning is to have many layers of neural network that model the learning process similar to that of the human brain. In recent years Convolutional Neural Networks have been used extensively in the field of Computer Vision. Computer vision has exploded in the recent years when Alexnet [3] won the ImageNet Competition in 2012. There have been several models since, which have made object detection, classification and other applications of Computer Vision more accurate.

There have been several papers written on building identification and rooftop extraction from aerial high resolution imagery which is the first process in our work. A 2016 paper called *Identification of Village Building via Google Earth Images and Supervised Machine Learning Methods* [4] by Guo et al. proposes a very efficient method using CNNs to identify buildings from Google Earth images. The authors of this paper also experimented with another supervised machine learning method called adaptive boosting (AdaBoost) which lost to the CNN by a small margin.

An IEEE paper published in 2015 by li et al. [5] suggests a novel method to extract rooftops from remote sensing imagery by using higher order conditional random fields(HCRF). They have combined the high-level information, obtained by by unsupervised image presegmentation, along with the low level pixel information.

The dataset that we have used already has building footprints for extracting the buildings. Therefore, we did not perform rooftop segmentation.

The core work that we have done in this project is to classify images once we have the extracted building images. Many papers have been written previously on classifying geospacial data, although none have aimed towards classifying buildings for disaster risk prediction. One such papers was written in 2015 by Luus et al. [6] for segregation of land into 21 predefined classes on the basis of their usage. In this paper a multiview deep learning approach is proposed. Another paper published in 2017 by Fu et al. uses a multiscale Fully Convolutional neural network for land classification from remote sensing imagery. They have introduced Atrous convolution to increase the density of the output class maps. There methods were performed on datasets with insufficient labelling which is a common drawback in remote sensing datasets. An efficient classification method despite this drawback was proposed in a 2017 IEEE paper by G. Scott et al. [7]. The method proposed in this paper uses data augmentation specifically tailored for remote sensing images and Transfer Learning with CaffeNet, GoogleNet and ResNet. The two papers mentioned above used the UC Merced data set to train their models.

A recent paper by Narloch et al. [8] used deep convolutional neural network (DCNN) to predict the compressive strength of cement-stabilized rammed earth (CSRE) trained and tested on SEM (scanning electron microscope) image database. This is similar to the problem the current paper is addressing, which is to predict the demolition-risk using rooftop images.

III. Study Area and Data Resources

A. Maps

The dataset consists of high resolution aerial drone images of 7 different regions (Borde rural and Borde Soacha, Mixco 1, Mixco 3 and Ebenezer, Castries, Dennery and Gros Islet) across 3 different countries (Colombia, Guatemala, St. Lucia), with resolutions of these images varying from 3.6 to 4.5 centimetres. The images in figures 1, 2 and 3 show the maps of the study area.

Due to the high level of resolution covering a large area mass, the size of the images are rather humongous. The smallest image out of the seven , that of mixco 3, is of size 1.6 Gigabytes and the rest scales up to 10 Gigabytes in size. The resolution, pixel dimension and size of each image is shown in table I Out of the seven TIFF images present, the labels of rooftops within two of those images (Castries and Gros Islet) are unverified. So those images have not been included within the training as well as test data.

Fig. 1: Map of borde-soacha and borde-rural in Colombia

Fig. 2: Map of mixco-3,mixco-1 and ebenezer in Guatamela

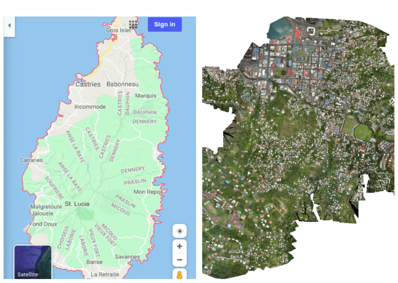


Fig. 3: Map of Castries in Saint Lucia

TABLE I: Details of the Dataset

Region	Resolution	Pixel Dimension	Size
$Borde\ Rural$	$4\ cm$	$5318 * 31315$	$4.5\ GB$
$Borde\ Soacha$	$4.25\ cm$	$40159 * 45650$	$4.5\ GB$
$Mixco\ 1\ and\ Ebenezer$	$4.3\ cm$	$27604 * 26641$	$2.0\ GB$
$Mixco\ 3$	$3.8\ cm$	$26066 * 19271$	$1.6\ GB$
$Dennery$	$4.2\ cm$	$21184 * 41534$	$2.5\ GB$

B. Other Data Resources

There is a folder for each country and a sub-folder for each region. These sub-folders contain the tiff image of that region along with a a thumbnail of the map, one imagery JSON file, one train GEOJSON file and one test GEOJSON file. The thumbnail provides a overview of the actual map with much lesser resolution, so as to help visualizing the image on a computer screen, since the TIFF images cannot be loaded by a default image viewer due to it's large size and resolution. There is an imagery data file which contains the co-ordinates for the outline of the town in the tiff image. The training GEOJSON file contains the unique building ID, building footprint(Geo-spatial coordinates of each building in the training set), roof material, and verified field. The test GEOJSON file contains the unique building id and building footprint(Geo-spatial coordinates of each building in the test set).

IV. Methodology

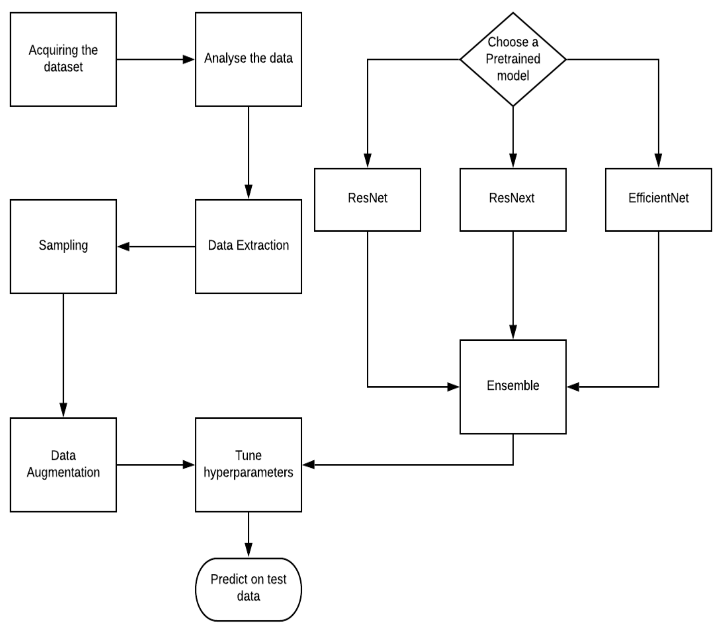


Fig. 4: Methodology

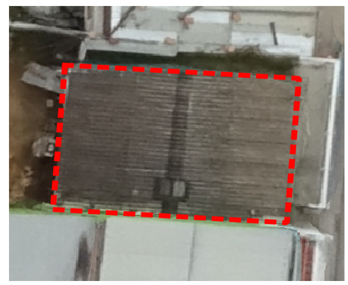
A. Data Pre-Processing

1) Data Extraction: Images of building rooftops were extracted from the high resolution aerial image of each town using the geojson file provided alongside it. The file consists of the building footprint in polygonal coordinates which were parsed using geopandas python package [9] and the exact rooftop image was extracted from the the big .tiff image using Rasterio python package.

2) Smote: The test data needs to be classified into five different classes of roof material. The training data is imbalanced, because in the real-world not all categories are equally found. In our dataset, there are many mare data entries for *healthy_metal* than there are for *other*. The distribution of the training data into different categories is shown in table II. As is evident from table II, there is an imbalance in the training data. This can cause the model to over represent some classes and under represent others. Also, the cost of misclassifying an abnormal class as normal is very high. For example, it is very important to not classify *incomplete* as *healthy_metal*, even though there is more training data available for *healthy_metal*. For this reason we use sampling. Synthetic Minority Oversampling Technique was proposed in 2002 by Chawla et al. [10] as an efficient method to classify minority classes accurately by introducing a bias towards the minority class.

3) Data Augmentation: Data Augmentation is an efficient technique for improving the diversity of training data by performing transformations on the images. Data Augmentation techniques include cropping, padding , flipping, zooming etc. For the rooftop images, the transformation performed were vertical flip, zoom and increasing the brightness of the image. The fastai framework was used to perform data augmentation and training predefined CNN models on our data [11]. The performance of the model improved by a great degree by using data augmentation.
Mixup is a regularization technique proposed by Zhang et al. in this 2017 article [12]. It uses data augmentation to prevent overfitting the training data. In mixup, instead of providing raw images we take a linear combination of two images and pass it as input to the model. Mixup increases the accuracy of the model.

TABLE II: Distribution of the training data

Category	*Sample_image*	*Count*
concrete_cement		1518
healthy_metal		14817
incomplete		669
irregular_metal		5241
other		308

B. Deep Learning Methodology

Convolutional Neural Networks are the best choice in computer vision or image classification tasks. Many CNN architectures have been experimented in this project, and the most accurate models were chosen upon which ensemble learning was applied to get the best outcome.

Most neural networks are trained with 32-bit floating point precision(FP32), which means that all the inputs, outputs and the weights are stored in FP32 format. The arithmetic operations performed on them also take FP32 numbers as input and output FP32 numbers. New hardware, however, has enhanced ALUs that operate faster on lower precision data types. We have trained our models using 16-bit floating point(FP16) precision [13].

1) Hyper parameters: Setting the optimal hyper-parameters is essential for training a deep learning model. Optimal hyper-parameters reduce the training time as well as improving the performance of the model. A 2018 paper by Leslie.N Smith outlines an efficient method for reaching an optimal balance point for learning rate, by examining the training and validation loss for subtle clues of underfitting and overfitting. The method suggested is called cyclical learning rate. The learning rate was determined for each model using this method. [14] The momentum and weight decay was set as constant in all models at 0.9 and 0.01 respectively. Batch size is 32 for all models.

The model was trained for a few epochs on the images with size 128x128, then the images are resized to 256x256 and the model was trained for few more epochs on these. After this, the pretrained model is unfreezed and trained for 10 epochs. No of epochs: For image size 128x128: 10 epochs, After resizing images to 256x256: 15 epochs and After unfreezing the model: 10 epochs

2) Residual Neural Networks: To get an accurate prediction of the roof material, the CNN must consist of a lot of layers. Depth of neural networks have proved to be a crucial factor for achieving better performance in classification tasks. But learning better networks is not as easy as stacking more layers. Stacking more layers provides us with the vanishing/exploding gradients issue [15]. This problem has been addressed through the normalized initialization and intermediate normalization layers, which enable deeper networks to start converging to the Stochastic Gradient descent using back-propagation. But this method exposes the *degradation* problem where, as the network depth increases, accuracy gets saturated and decreases rapidly. To address this problem we use a deep residual neural network(resnet) [16] proposed by microsoft research in 2016, where skip connections are established which enables us to make more layers , thereby increasing the accuracy of the model. In resnet, we can go upto 200 layers before which the accuracy gets saturated. The best resnet model for our purpose was the pretrained resnet152 model of the fastai python framework.

An improvisation to the resnet models was provided by Facebook Ai research [17]. Their models, called resnext, have shown better results for image classification as compared to resnet models. They introduce a new dimension to the neural network called *cardinality*, in addition to depth and width. Usually in neural networks, linear transformations are performed on the present layer to obtain each neuron in the next layer. This paper proposes aggregating a set of transformations with the same topology, instead of performing linear transformations only. The size of this set of transformations is called *cardinality*. We have used pretrained resnext models Resnext101 64x4,which is a 101 layer deep model with cardinality 64, and resnext101 32x8, which has cardinality 32. [18].

3) EfficientNet: EfficientNet architecture was introduced by Tan et al, in the 2019 paper EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks [19]. As the name suggests, the paper demonstrates a method of scaling the three dimensions of a CNN, namely, depth, width and height. It is found that scaling any one dimension does not improve the accuracy linearly, instead the accuracy quickly saturates. Although resnets allow us to go deeper than normal, going deeper than 200 layers does not improve the efficiency of the CNN.

The solution to this problem lies in scaling multiple dimensions together. But scaling multiple dimensions arbitrarily results in sub-optimal accuracy and efficieny. The proposed scaling technique in the above article is the basis for efficient-net architectures.

Scaling doesnot change the layer operations of a network. So an optimal base network is chosen on top which optimal scaling parameters(α,β,γ,ϕ) are searched. By setting the value of $\alpha = 1.2, \beta = 1.1$ and $\gamma = 1.15$ and experimenting with the values of ϕ gives Efficientnet b1-b7.

For the roof material classification task mentioned in this paper, pretrained models of efficientnet implemented by [20] were used using the fastai framework. EfficientNet b5, b6 and b7 were used.

C. Ensemble Modelling

In addition to augmenting training data, we have performed test time augmentation on the test data and aggregated the prediction probabilities to get a more accurate prediction. [21] Ensemble is a method of classifying new data points by taking a set of classifiers and combining them in some aggregate function. Weighted average was used in the present classification task as an aggregate. The ensemble was performed on the outputs obtained from training the data on resnet152, resnext101 64x4, resnext 101 32x8, efficientnet b5, efficientnet b6 and efficientnet b7.

V. Results

The performance metric to evaluate the models' accuracy is the log loss function, which is an error metric. The log loss can be calculated by the following equation.

$$loss = \frac{-1}{N} . \sum_{i=1}^{N} \sum_{j=1}^{M} y_{ij} \log p_{ij} \quad (1)$$

A sample test image along with its predicted results are shown in figure 5.

The performance of the most accurate individual pretrained models on the validation set and the test set are shown in table III. The number of epochs which gave the best validation loss after unfreezing the model is also mentioned.

TABLE III: Performance of individual models

$Model$	$epochs$	$train\_loss$	$valid.\_loss$	$error\_rate$	$test\_loss$
Resnet152	2	0.465180	0.475009	0.182246	0.4723
Resnext101 64×4	4	0.591409	0.378783	0.139543	0.4898
EfficientNet b6	3	0.527126	0.520498	0.191325	0.4610
EfficientNet b7	8	0.584133	0.396385	0.147613	0.4526

The least log loss of **0.4373** was obtained from the ensemble of resnet152, resnext101 64x4, resnext 101 32x8, efficientnet b5, efficientnet b6 and efficientnet b7.

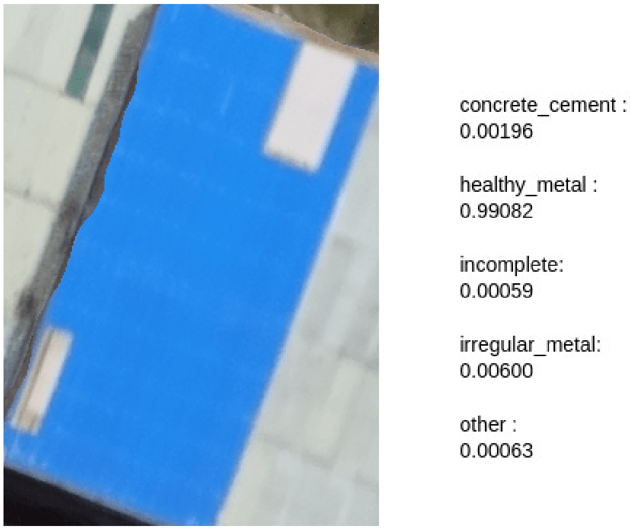


Fig. 5: Predictions on a test image

VI. Conclusion

Image Classification using deep learning techniques is a powerful alternative for manual inspecting in Disaster Risk Management. With the accuracy obtained using the proposed model, we can make a very good prediction about the strength of the buildings. We were introduced to this project through a competition conducted by Drivendata. With the help of this ensemble model, we have secured a global rank of 19 among 1425 participants in this competition. The methods employed in this project are based on the latest developments in image classification. The model can be improved with future developments in the field to make the predictions more accurate. Since modern neural networks tend to be overconfident about their predictions, better calibration techniques can be explored further to reduce the overall logloss. Training the model with Label Smoothing loss or using temperature scaling as propsed here [22] may improve the model's capability to predict more accurate probability values for each class.

Acknowledgment

This project was supported by LeadingIndiaAI, an AI and deep learning Skilling and Research institution. We would like to thank our program coordinator Dr. Madhushi Verma, director of LeadingIndiaAI Dr. Deepak Garg, Dr. Vipul Mishra and Dr. Tapas Badal for providing us with the opportunity to work on a machine learning project and guiding us through the process.

The project was initiated through a competition supported by Driven Data [23]. We would like to thank Driven Data for providing us with this opportunity. The data was prepared by WeRobotics and World Bank Global Program for Resilient Housing.

We would also like to thank Bennett University for conducting this project and providing us with computational resources.

References

[1] *Open AI Caribbean Challenge: Mapping disaster risk from aerial imagery*, https://blog.werobotics.org/2019/10/08/open-ai-caribbean-challenge-mapping-disaster-risk-from-aerial-imagery/, Accessed: 2019-12-24.

[2] G. Hinton, S. Osindero, M. Welling, and Y.-W. Teh, "Unsupervised discovery of nonlinear structure using contrastive backpropagation," *Cognitive science*, vol. 30, no. 4, pp. 725–731, 2006.

[3] M. Z. Alom, T. M. Taha, C. Yakopcic, S. Westberg, P. Sidike, M. S. Nasrin, B. C. Van Esesn, A. A. S. Awwal, and V. K. Asari, "The history began from alexnet: A comprehensive survey on deep learning approaches," *arXiv preprint arXiv:1803.01164*, 2018.

[4] Z. Guo, X. Shao, Y. Xu, H. Miyazaki, W. Ohira, and R. Shibasaki, "Identification of village building via google earth images and supervised machine learning methods," *Remote Sensing*, vol. 8, no. 4, p. 271, 2016.

[5] E. Li, J. Femiani, S. Xu, X. Zhang, and P. Wonka, "Robust rooftop extraction from visible band images using higher order crf," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 53, no. 8, pp. 4483–4495, 2015.

[6] F. P. Luus, B. P. Salmon, F. Van den Bergh, and B. T. J. Maharaj, "Multiview deep learning for land-use classification," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 12, pp. 2448–2452, 2015.

[7] G. J. Scott, M. R. England, W. A. Starms, R. A. Marcum, and C. H. Davis, "Training deep convolutional neural networks for land–cover classification of high-resolution imagery," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 4, pp. 549–553, 2017.

[8] P. Narloch, A. Hassanat, A. S. Tarawneh, H. Anysz, J. Kotowski, and K. Almohammadi, "Predicting compressive strength of cement-stabilized rammed earth based on sem images using computer vision and deep learning," *Applied Sciences*, vol. 9, no. 23, p. 5131, 2019.

[9] *Geopandas: Making work with geospacial data easier*, http://geopandas.org/, Accessed: 2019-12-24.

[10] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "Smote: Synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.

[11] *Making neural nets uncool again*, https://www.fast.ai, Accessed: 2019-12-24.

[12] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," *arXiv preprint arXiv:1710.09412*, 2017.

[13] T. He, Z. Zhang, H. Zhang, Z. Zhang, J. Xie, and M. Li, "Bag of tricks for image classification with convolutional neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 558–567.

[14] L. N. Smith, "A disciplined approach to neural network hyper-parameters: Part 1–learning rate, batch size, momentum, and weight decay," *arXiv preprint arXiv:1803.09820*, 2018.

[15] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[17] S. Xie, R. Girshick, P. Dollár, Z. Tu, and K. He, "Aggregated residual transformations for deep neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1492–1500.

[18] *Resnext wsl*, https://pytorch.org/hub/facebookresearch_WSL-Images_resnext/, Accessed: 2019-12-24.

[19] M. Tan and Q. V. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," *arXiv preprint arXiv:1905.11946*, 2019.

[20] *Efficientnet-pytorch*, https://github.com/lukemelas/EfficientNet-PyTorch, Accessed: 2019-12-24.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE international conference on computer vision*, 2015, pp. 1026–1034.

[22] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, JMLR. org, 2017, pp. 1321–1330.

[23] *Open AI Caribbean Challenge: Mapping disaster risk from aerial imagery*, https://www.drivendata.org/competitions/58/disaster-response-roof-type/page/142/.