

Real-time Scene Understanding for UAV Imagery based on Deep Convolutional Neural Networks

Clay Sheppard and Maryam Rahneemoonfar
School of Engineering and Computing Sciences, Texas A&M University-Corpus Christi

Abstract

Real-time scene understanding is important for many applications of Unmanned Aerial Vehicles (UAVs) such as reconnaissance, surveillance, mapping, and infrastructure inspection. With the recent growth of computation power, it is feasible to use Deep Learning for real-time applications. Deep Convolutional Neural Networks (CNNs) have emerged as a powerful model for classifying image content, and are widely considered in the computer vision community to be the de facto standard approach for most problems. Current Deep learning approaches for image classification and object detection are designed and evaluated on lab setting human-centric photographs taken horizontally from a height of 1-2 meters. UAV images are taken vertically in high altitude; therefore the objects of interest are relatively small with a skewed vantage point which creates a real challenge in detection and classification of such images. Here we present a deep convolutional approach for classification of Aerial imagery taken by UAV. We applied our network on optical imagery taken with UAV RS-16 from Port Mansfield, TX. Experimental results in comparison with ground-truth show 93.6 % accuracy for UAV image classification.

1. Introduction

Unmanned Aerial Vehicles (UAVs) have seen unprecedented level of growth in recent years. UAVs are increasingly used for surveillance, fire detection, reconnaissance, mapping, cartography, landslide monitoring, inspection, traffic monitoring, search and rescue, to name a few application domains [1-3]. It is important for many of the aforementioned applications to perceive the scene in real-time. If a high altitude UAV can perform scene understanding by predicting the nature of landscape at any particular location, then a lower altitude UAV or a UGV (unmanned ground vehicle) can be sent to the desired location for further detailed analysis for a specific application. The main purpose of the scene understanding using high altitude UAV is to remove the overhead of finding the desired location for the lower

altitude UAV or UGVs for their specific purposes. Current computer vision algorithms and datasets are designed and evaluated on lab setting human centric photographs taken horizontally with a close distance to the object. For UAV imagery taken vertically in high altitude (10m to 100m) the objects of interest are relatively small and with less features, questioning the sustainability of current methods. For example an aerial image of building has only the top view of the building and except the roof, no distinguishable features are there. On the other hand corresponding terrestrial image of the same building has many features like door, windows, and walls which makes it easier for recognition even by human.

Several approaches are utilized for object detection in aerial images. In 1988, Huertas and Nevatia [4] proposed a technique to detect buildings according to the rectangular components and shadow information. Based on the similar concept, Sirmacek and Unsalan [5] presented an approach to detect buildings using invariant color features, edge, and shadow information. Cote and Saeedi [6], introduced a method using Harris corner detector for building detection. Manno-Kovacs and Sziranyi [7] proposed a framework based on region orientation with several steps to achieve building detection. The main drawback of aforementioned methods is that they are not suitable for real time applications. Moreover, traditional approaches cannot learn new features automatically. Feature engineering is required to decide explicitly what features to learn. With the recent advances in GPU technology, deep learning has emerged as a feasible solution to real time applications. Deep learning consists of simultaneous learning of hierarchical models from multiple levels of representation that helps to identify input data [7, 8]. However limited work is performed for scene understanding using UAV images.

In this paper, a deep convolutional neural network framework is implemented to achieve a fast and accurate result for scene understanding on Aerial images taken with UAV. The network architecture comprises a series of convolution and pooling layers followed by fully connected layers. We applied our network on optical imagery taken with UAV RS-16 from Port Mansfield, TX. The proposed architecture is able to predict the labels for

the images captured by UAVs in real time. We applied the proposed network on 3864 images and achieved the accuracy of 93.6% on test dataset. We also evaluated the classwise accuracy for all four classes. In addition to real-time class prediction with high-accuracy, our deep learning approach has automatic feature learning and the ability to handle larger datasets.

The rest of the paper is organized as follows. Proposed methodology is explained in section 2. Experimental results are presented in section 3. Finally, the discussions and conclusions are drawn in section 4.

2. Deep network architecture

Convolutional Neural Networks (CNN) comprises various convolutional and pooling (subsampling) layers that resemble human visual system. The network architecture that was developed in this work is shown in Figure 1. As we can see in Figure 1, the first layer of the network is input layer containing the input image. The convolutional layer COV1 takes the 3 bands (RGB) in the input image and produces 6 different feature maps using a 5X5 kernel function. The convolution layer is followed by 2X2 max pooling layer (MP1 layer) with stride 2. 2X2 Max pooling reduces the dimensions of the image by taking the maximum value in a window at every depth slice in the feature map by 2 along both width and height. A stride of 2 indicates that the window is moved two pixels at a time. This condenses the information by reducing the features by half. Reducing the dimensions of the image reduces computation time and allows the model to fit into GPU memory [9]. After first pooling layer we applied another layer of convolution (COV2 layer). This convolutional layer takes 6 output features from the previous layer as input and maps them to 12 feature maps using a 5X5 kernel function. The convolution layer is again followed by a 2X2 max pooling layer (MP2 layer) with stride 2. This max pooling layer is followed by a fully connected layer (FC1 layer). As can be seen in Figure 3 the size of this fully connected layer is 8. We used dropout to prevent the network from over-fitting at this stage. Moreover, it helps network to learn fast. According to this technique,

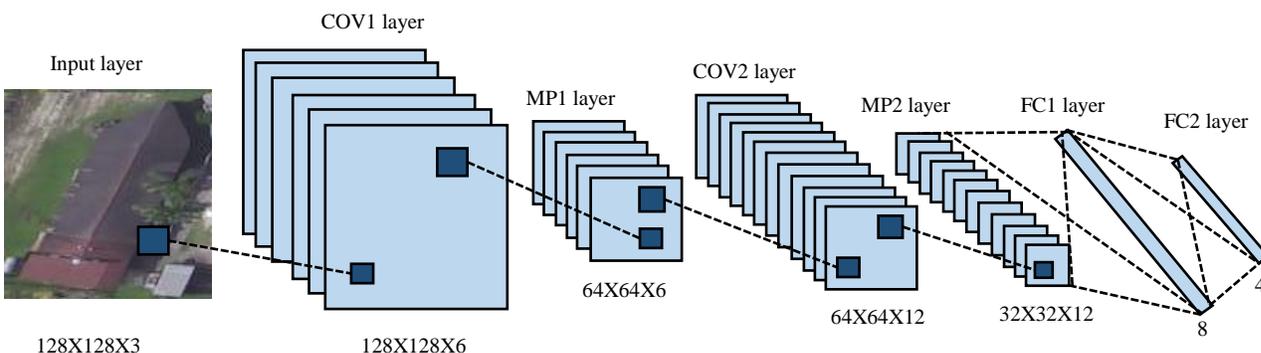


Figure 1: Network architecture developed for scene understanding

some units are randomly dropped along with their connections [10]. How many connections will be dropped off is decided by the percentage of dropout. In our research 50% of connections were randomly dropped off while training the network. Finally, the last fully connected layer (FC2) after the dropout layer gives the prediction for classification with the output size 4 because we have used four classes in this research namely, building, ground, road and tree. Softmax was used after the final output layer. Softmax activation is the normalized exponential probability of class observations represented as neuron activations. It is used for the output layer to ensure that the sum of the components of output vector is equal to 1. Batch normalization was performed after every convolution to remove internal covariate shift [11].

Out of 3880 images around 70% are used for training and 30% for testing. The network was trained for 60 epochs on 2800 images. To minimize the error an Adam optimizer is used [12] because it requires little tuning of hyperparameters. The learning rate for the Adam optimizer is set at a constant 1e-3. Cross entropy [13] is used as the cost function. Rectified linear function [14] is used as an activation function.

3. Experimental Results

3.1. Dataset

The dataset used in this work comprises images captured by an optical camera mounted on a UAV. The study area for this research is Port Mansfield, TX and data was captured on March 4-6, 2015. Figure 2 shows the google earth image of the study area along with the sample image captured by UAV in the inset. The UAV used in this work is Recon System™ (RS-16) Unmanned Aircraft System (UAS). The aircraft is a multi-payload, long endurance system capable of performing safe and successful civil missions in remote locations. Details of the aircraft, payload, and operations can be found in [15]. The resolution of the captured images is 1920 x 1080. The red circle in the Figure 2 shows the limit of our range



Figure 2: Location of the study area on Google earth along with the sample image captured by UAV in the inset

based on the C2 radio link between the GCS and the RS-16, and the blue box shows the airspace limitations of our COA. A sample image captured by UAV can also be seen in the inset. The network was implemented using TensorFlow running on an NVidia 980Ti GPU. For training, 2800 images were used. For testing, a different set of 1064 images was used.

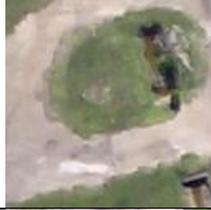
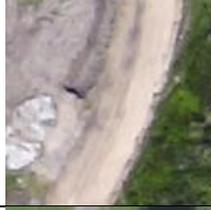
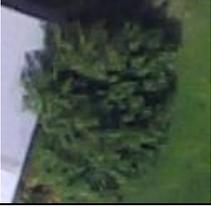
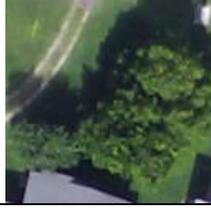
3.2. Network testing and validation

We tested and validated our algorithm on 1064 images. The network was trained with the dropout value of 50 which indicates that 50% connections were dropped off randomly from the fully connected part of the network while training. We achieved the high accuracy of 93.6% on 1064 testing dataset. Our method is also very fast. The execution time required for our deep architecture is 0.0012 second.

Table 1 presents some difficult images for which our algorithm predicts accurately. As can be seen in Table 1, the network is able to predict well on some difficult images. One of the building images presented here is round shaped, and despite being not trained to predict round shaped buildings explicitly, the network was able to predict it well. In the ground images and one of the tree images mentioned in the table, there can be observed road like structures, but based on the dominant feature present in the image the network was able to predict it accurately. The road images presented in Table 1 have no proper

boundaries but still network could detect the roads.

Table 1. Correctly classified images and their label

Sample images		Predicted Label
		Building
		Ground
		Road
		Tree

3.3. Class wise Accuracy Assessment

In this section we calculated class wise statistics to analyze the performance of our method on individual classes. We computed precision and recall for all the classes as shown in Table 2. Precision corresponds to error of commission (inclusion); it estimates how many instances classified in a particular class are actually from that class. Recall corresponds to error of omission (exclusion); it estimates how many instances actually belong to a particular class are classified correctly [16].

Table 2. Class wise precision and recall

Class	Precision (%)	Recall (%)
Building	97.96	96.97
Ground	90.20	92.00
Road	90.80	87.78
Tree	95.83	98.57
Overall accuracy (%)	93.6	

From Table 3, it can be observed that all the classes have adequate precision and recall. Building and tree classes have high precision and recall as compared to ground and road. There are some false positives and false negatives observed in ground and road classes that are responsible for relatively lower precision and recall as compared to building and tree classes. Further investigations by looking at the images which are not classified correctly, revealed that few samples from ground are classified as road and vice versa. The reason is few samples in the road class are mud roads (dirt roads) that have similar texture with ground.

4. Conclusion

We presented a deep convolutional neural network framework for scene understanding on UAV based optical images. Traditional scene understanding methods require feature engineering to learn features explicitly. The main advantage of using Deep Learning is that, unlike traditional methods it automatically learns features. Semantic outputs are generated to classify various objects such as building, tree, ground and road without additional translation. Our network architecture comprises series of convolution and pooling layers followed by fully connected layers. We applied our network on optical imagery taken with UAV RS-16 from Port Mansfield, TX. The proposed architecture is able to predict the labels for the images captured by UAVs in real time. We reached the overall accuracy of 93.6% on the test dataset.

References

- [1] K. P. Valavanis, *Advances in unmanned aerial vehicles: state of the art and the road to autonomy* vol. 33: Springer Science & Business Media, 2008.
- [2] U. Niethammer, M. James, S. Rothmund, J. Travelletti, and M. Joswig, "UAV-based remote sensing of the Super-Sauze landslide: Evaluation and results," *Engineering Geology*, vol. 128, pp. 2-11, 2012.
- [3] S. Nebiker, A. Annen, M. Scherrer, and D. Oesch, "A light-weight multispectral sensor for micro UAV—Opportunities for very high resolution airborne remote sensing," *The international archives of the photogrammetry, remote sensing and spatial information sciences*, vol. 37, pp. 1193-1199, 2008.
- [4] A. Huertas and R. Nevatia, "Detecting buildings in aerial images," *Computer Vision, Graphics, and Image Processing*, vol. 41, pp. 131-152, 1988.
- [5] B. Sirmacek and C. Unsalan, "Building detection from aerial images using invariant color features and shadow information," in *Computer and Information Sciences, 2008. ISCIS'08. 23rd International Symposium on*, 2008, pp. 1-5.
- [6] M. Cote and P. Saeedi, "Automatic rooftop extraction in nadir aerial imagery of suburban regions using corners and variational level set evolution," *IEEE transactions on geoscience and remote sensing*, vol. 51, pp. 313-328, 2013.
- [7] A. Manno-Kovacs and T. Sziranyi, "Orientation-selective building detection in aerial images," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 108, pp. 94-112, 2015.
- [8] S.-h. Zhong, Y. Liu, and Y. Liu, "Bilinear deep learning for image classification," in *Proceedings of the 19th ACM international conference on Multimedia*, 2011, pp. 343-352.
- [9] V. Dumoulin and F. Visin, "A guide to convolution arithmetic for deep learning," *arXiv preprint arXiv:1603.07285*, 2016.
- [10] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *Journal of Machine Learning Research*, vol. 15, pp. 1929-1958, 2014.
- [11] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.
- [12] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [13] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of operations research*, vol. 134, pp. 19-67, 2005.
- [14] G. E. Dahl, T. N. Sainath, and G. E. Hinton, "Improving deep neural networks for LVCSR using rectified linear units and dropout," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8609-8613.
- [15] D. H. Bridges and D. Yoel, "Unmanned Aircraft Operations at Texas A&M University—Corpus Christi," in *AIAA Infotech@ Aerospace*, ed, 2016, p. 0745.
- [16] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern recognition*, vol. 37, pp. 1757-1771, 2004.