# Detecting Coconut Trees from Aerial Photographs

Amber Thomas and Tynan Challenor


## Abstract

- Natural disasters in the tropics can threaten food security [2]
- As a result, the World Bank issued a challenge to the global AI community to develop computer vision algorithms to count and locate standing food trees, like coconut trees, from aerial photographs [3]
- With an RCNN using Resnet101 base classifier pre-trained on the kitti object detection dataset and tuned using an in situ aerial photograph taken over the Island of Tonga, we trained a detection model that returned 94% AP using the PASCAL VOC evaluation metric [4], [5]


## Faster R-CNN Resnet101

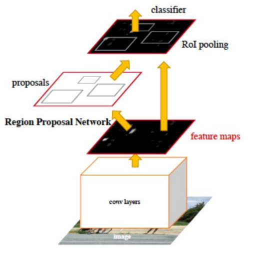


**Figure 1:** Faster R-CNN architecture—a region proposal network and a classification network [1].

Attributes of the Faster R-CNN [1], [7]:

Region Proposal Network

- Passes image through a CNN, generating a feature map
- A sliding window generates anchor boxes containing:
  - Likelihood that object is present score
  - 4 coordinates of the anchor box

The features from the proposed regions are then passed on for classification, which entails:

- Pooling the ROIs layer to pass onto fully connected layers, softmax classifier, and another linear regression for layer bounding boxes

Resnet101 [6]:

Loss Function:

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{cls}}\sum_i L_{cls}(p_i, p_i^*) + \lambda\frac{1}{N_{reg}}\sum_i p_i^* L_{reg}(t_i, t_i^*).$$

- $p_i$ is the predicted probability that i's anchor is an object
- $t_i$ is the vector containing the four corners of a proposed bounding box
- Classification loss was binary log loss
- Regression loss for the predicted bounding boxes was smooth L1

## Dataset and Data Cleaning

Our data set was a single 25,006 by 17,761 aerial photograph with a spatial resolution of 9cm, covering a roughly 50km2 region. It came with georeferenced map points of coconut trees. We padded and broke up the image into more manageable 300 by 300 segments, yielding 5040 sub images and 11,300 labelled coconut trees in total. We then randomly shuffled these sub images to separate train (74%, 3,832 images), dev (12%, 604 images), and test (12%, 604 images) sets. In addition we augmented the data by randomly flipping the images half the time. During our first round of testing and analysis, we found that there was a high incidence of unlabeled trees. After a round of clean up, we correctly labeled over a thousand more trees.

## Results


**Image 2.** Correctly bounded trees.


**Image 3.** Incorrectly bounded banana tree.

The algorithm learned quickly to identify coconut trees with 94% AP:

- Green boxes are those predicted by the algorithm, black ones are the ground truth
- Not all data were labeled correctly, yet the algorithm still learned to distinguish coconut trees; see the top right tree in image 2
- The algorithm sometimes mistook banana trees for coconut trees; see image 3
- After 8.87K iterations, loss was 0.049

| Model Attribute | Specification |
|---|---|
| Batch Sizes | 1 (SGD) |
| Regularization | L2 |
| Initialization | Variance scaling initializer |
| IOU Threshold | 0.6 |
| Optimizer | Momentum |
| Starting $\alpha$ | 0.0001 |
| Learning rate decay | Hand adjusted |
| Data Augmentation | Random horizontal flip |

## Error Analysis

In 160 dev images (multiple trees per image), the model made errors:

- 10 times it bounded a banana tree—they have similar fronds
- 7 times it did not bound a coconut tree that was present
- 4 times it bounded the shadow of a tree

The Training data were not well labeled, some banana trees were mislabeled coconut trees. A more extensive analysis is needed to determine the best way to decrease loss further. Some possibilities include:

- Removing positive labels from banana trees
- Training longer will help with:
  - Not labeling shadows and correctly -labeling coconut trees

Many unbounded coconut trees were wilted, the fronds were not like spokes

- More training examples with wilted fronds

## Discussion

We needed a detection algorithm in order to locate and count coconut trees. We initially trained on a Single Shot Detector with an inception base network in the hopes of developing a quick-to-train model without compromising on accuracy. However, after being unable to achieve the desired accuracy scores we decided to change architectures to a Faster-RCNN, which performs better than SSD's when detecting smaller to mid sized objects.

One advantage to lightweight networks such as mobilenet is that they can be run on mobile devices. This could be a boon for disaster relief efforts. To train such a network we would likely need more training data with more reliable labeling.

Nonetheless, our object detector using a faster R-CNN accomplishes the challenge set out by the World Bank, an example of deep learning's applicability.

## References


[1] J. Xu, "Deep Learning for Object Detection: A Comprehensive Review," Towards Data Science, 11-Sep-2017. [Online]. Available: https://towardsdatascience.com/deep-learning-for-object-detection-a-comprehensive-review-73930816d8d9. [Accessed: 20-Mar-2018].

[2] A. Fritz, Tropical Cyclone Gita is a monster Category 4, and its hammering Tonga, Washington Post, 12-Feb-2018.

[3] K. Leetaru, Using AI For Good: A New Data Challenge To Use AI To Triage Natural Disaster Aerial Imagery, Forbes. [Online]. Available: https://www.forbes.com/sites/kalevleetaru/2018/01/20/using-ai-for-good-a-new-data-challenge-to-use-ai- to-triage-natural-disaster-aerial-imagery/. [Accessed: 23-Feb-2018].

[4] "Speed/accuracy trade-offs for modern convolutional object detectors." Huang J, Rathod V, Sun C, Zhu M, Korattikara A, Fathi A, Fischer I, Wojna Z, Song Y, Gudarrama S, Murphy K, CVPR 2017

[5] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 3354–3361.

[6] J. Xu "An Intuitive Guide to Deep Network Architectures," *Towards Data Science*, 14-Aug-2017 [Online]. Available:

[7] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *arXiv:1506.01497 [cs]*, Jun. 2015.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *arXiv:1512.03385 [cs]*, Dec. 2015.