# Tree species detection model for agroforestry

Anonymous CVPR submission

Paper ID ****

## 1. Introduction

Trees are a large carbon pool for our planet. They play a vital role in offseting anthropogenic carbon emissions and attenuating the effects of climate change. We can grow this pool through afforestation and deferred deforestation, oftentimes supported through carbon offset payments by sectors emitting carbon to the landowners growing these forests.

One high-potential opportunity for afforestation is through the transition of pastureland operations to silvopasture operations. Silvopasture is simply the agricultural practice of combining tree cropping with livestock management.

In order for a payment system like this to work, we need a highly accurate estimate of the amount of carbon stored in trees. Fortunately, decades of research has been invested into tree growth projections and today, allometric equations are readily available for many tree species, which can accurately estimate above and below-ground tree biomass as a function of tree height, tree diameter, species, and local climate.

This research develops a novel dataset of tree photographs and trains a deep learning model to identify tree species from images in the silvopasture setting. These tools will be combined with a broader suite of tools being developed to accurately estimate tree carbon from phone-based measurements to enable carbon payments on agricultural operations transitioning to silvopasture. In the future, these phone-based measurements will be used to augment and improve the models developed in this paper.

### 1.1. Problem statement

In this paper I compile a novel dataset of high-fidelity tree photographs, taken in profile perspective. The dataset contains classified images of the 7 most common trees used in silvopasture in the southeastern US, where pilot projects for this silvopasture transition are underway. I then train a deep learning model to predict tree species from profile-vew tree photographs

### 1.2. Related work

Image classification is a common problem solved using convolutional neural networks. [[DESCRIPTIONS OF IMAGENET, ALEXNET, VGGNET, GOOGLENET, AND RESNET]] [5, 6, 4].

[[PAPER]] Expanded the possibilities of using pre-trained models in the described frameworks for use in more targeted settings. [[SUMMARY OF APPROACH]]. [[SUMMARY OF FINDINGS]].

In Feng et al.'s work on long-tailed object detection, they explore training classification models on highly-similar objects, a similar challenge to that conducted in this paper [2]. [[SUMMARY OF APPROACH]]. [[SUMMARY OF FINDINGS]]

Carpentier et al. use a self-collected dataset to a very similar problem: tree detection through bark [1]. [[SUMMARY OF APPROACH]]. [[SUMMARY OF FINDINGS]].

Fricker et al. also attempts to classify trees, however from an aerial perspective instead of the profile perspective taken in this paper [3]. [[SUMMARY OF APPROACH]]. [[SUMMARY OF FINDINGS]]

## 2. Dataset

This paper develops a novel dataset of high-fidelity tree photographs, taken in profile perspective. The species included are the seven most common species of tree used in silvopasture in the southeastern US: Black Locust, Black Walnut, Honey Locust, Loblolly Pine, Northern Red Oak, Pecan, Chinese Chestnut.

I compile these photographs from images scraped from the internet and augmented with reflections and random croppings of each image. All images are center-cropped and scaled for use in a deep learning model. Lastly, the dataset is filtered, using a pre-trained deep learning model, to only those images with high likelihood of being trees.

### 2.1. Image scraping

I use three data sources to generate a dataset of tree images: the Harvard Arboretum Plant Image database [[CITE]], the Arbor Day Foundation website landing page for each tree [[CITE]], and the results of Microsoft Bing image searches for each tree species of interest. I also evaluated BarkNet, a dataset of close-up photographs of tree

| | N |
|---|---|
| Harvard Arboretum | |
| Arbor Day Foundation | |
| Bing image search | |
| Total | |

Table 1. Number of images scraped from each data source

| | N |
|---|---|
| Black Locust | |
| Black Walnut | |
| Honey Locust | |
| Loblolly Pine | |
| Northern Red Oak | |
| Pecan | |
| Chinese Chestnut | |
| Total | |

Table 2. Number of images scraped for each tree species

bark supporting a tree-bark identification model, however, this dataset did not contain any of the species of interest in this project.

I developed website-specific python codes to scrape and download images of each tree species from each source. I downloaded all available images from the Harvard Arboretum and Arbor Day sources for each of the species of interest. And for the results of my Bing Image searches, I downloaded all of the first [[300 now, will be 1000]] images that were not protected from python code web access. I use Bing image search instead of Google image search because of recent changes in Google's website layout that make web scraping more difficult.

This results in a preliminary dataset of [[XXX]] images, detailed below.

### 2.2. Dataset augmentation

I augment the dataset with a reflection of each image about its vertical axis. I do not include horizontal and diagonal reflections because all tree images classified with this model are expected to be up-right. I also augment the dataset with [[3]] croppings of each image at [[3]] different relative sizes, [[10%, 25%, and 50%]]. The centroid of each cropping is selected at random from the set of all centroids that allow the full crop region to lie within the photograph.

This results in a dataset of [[XXX]] images.

In a future iteration of this work, I will also consider augmenting my dataset with hue and value-shifted images, and fooling images [[CITE]].

### 2.3. Image cropping and scaling

I center and scale all images to the same shape and size for use in a deep learning model. I first upscale all images to the maximum image size in the dataset. I next center-crop each image, taking the largest square within the image with centroid at the image center. And finally I downscale all images to [[32x32]] pixels to improve the speed and efficiency of my model.

### 2.4. Dataset filtering

Lastly, I filter the dataset to only include, with high likelihood, profile-perspective photographs of trees. I choose to filter my dataset because each of my data sources may include drawings of trees or images loosely related to trees (e.g., a table made from the wood of a black walnut tree). The former is more of a concern in the Harvard Arboretum and Arbor Day Foundation datasets, and both the former and latter are concerns in the results of Bing image searches. My dataset also includes random croppings of all images, which may not include any component of a tree and I choose to filter these croppings too.

I filter my dataset by running a pre-trained deep neural net classifier on my images and only keeping those images whose prediction probability for the tree class exceeded a threshold of [[TBD]]. I choose to use the [[ResNET]] model for this filtering, and I scale all images to [[3xYxY]] pixels to use the model on my images. I only require scaling images because both my dataset and the dataset used to train ResNET use images with equal height and width.

I include a histogram of tree class predictions for all images in my database below. This filtering removes [[XX]] images from my dataset and results in a final dataset of [[XX]] images.

## 3. Technical approach

In this paper I aim to train a model with the lowest Top-1 misclassification error. This metric best aligns with ultimate objective of this model, which is to make the best tree species prediction from an image taken on a phone while on a silvopasture operation. For robustness, I also evaluate Top-5 misclassification error, precision, recall and F2 scores of predictions across the models I construct.

In selecting the highest-performing model, I consider three categories of models: baseline models, deep convolutional neet net models, and transfer learning models.

I first train four baseline models. These models serve as a standard against which I can aim to improve classification error through the use of deep convolutional neural networks. The models I consider are K-nearest neighbors, multivariate logistic regression, support vector machine classification, and a simple, fully connected neural net. I optimize the hyperparameters associated with each of these models

| | Top-1 | Top-5 | $F_2$ |
|---|---|---|---|
| KNN | | | |
| Logistic | | | |
| SVM | | | |
| FC Net | | | |

Table 3. Preliminary performance results of baseline models

through a grid search approach.

I next optimize three deep convolutional neural net models. These models are the target group of models from which I intended to generate tree classifications from images. I consider structures similar to three of the most recognizeable high-performing CNNs: VGGNet, GoogLeNET, and ResNET [5, 6, 4]. I optimize the hyperparameters associated with each of these models through a grid search approach.

Lastly, I implement transfer-learning techniques to train the final fully connected layers of the three high-performing models mentioned above: VGGNet, GoogLeNET, and ResNET. In these models, I rescale my tree image dataset to be compatible with each model's inputs, utilize the pretrained convolutional layers in each model, and train only the final fully connected layers, maintaining the same dimensions of the original table.

My model selection is based on the highest performing model among those described in this section.

## 4. Preliminary results

Describe quantitative evaluation results obtained thus far. Results/Evaluation: The report should contain a quantitative evaluation of the baseline method(s) described in the method section. The quantitative evaluation results should be in the form of figures or tables. All evaluation results should contain metrics other than loss values.

After tuning each of the baseline models, I find that [[MODEL]] results in the lowest Top-1 misclassification error. This performance is [[XXX]] percentage points lower than predictions made at random. The next highest-performing model was [[MODEL]], and the lowest-performing model was [[MODEL]]. Results are included below.

## References

[1] M. Carpentier, P. Giguère, and J. Gaudreault. Tree species identification from bark images using convolutional neural networks, 2018. 1

[2] C. Feng, Y. Zhong, and W. Huang. Exploring classification equilibrium in long-tailed object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3417–3426, October 2021. 1

[3] G. A. Fricker, J. D. Ventura, J. A. Wolf, M. P. North, F. W. Davis, and J. Franklin. A convolutional neural network classifier identifies tree species in mixed-conifer forest from hyperspectral imagery. *Remote Sensing*, 11(19), 2019. 1

[4] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015. 1, 3

[5] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition, 2014. 1, 3

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions, 2014. 1, 3