

Classification of Bark Beetle-Induced Forest Tree Mortality Using Deep Learning

Rudraksh Kapil^{*†§}, Seyed Mojtaba Marvasti-Zadeh^{†§}, Devin Goodsman[‡], Nilanjan Ray^{*}, and Nadir Erbilgin[†]

^{*}Department of Computing Science, University of Alberta

[†]Department of Renewable Resources, University of Alberta

[‡]Canadian Forest Service, Natural Resources Canada, Edmonton, Alberta

Emails: {rkapil, seyedmoj, nray1, erbilgin}@ualberta.ca, devin.goodsman@nrcan-rncan.gc.ca

Abstract—Bark beetle outbreaks can dramatically impact forest ecosystems and services around the world. For the development of effective forest policies and management plans, the early detection of infested trees is essential. Despite the visual symptoms of bark beetle infestation, this task remains challenging, considering overlapping tree crowns and non-homogeneity in crown foliage discoloration. In this work, a deep learning-based method is proposed to effectively classify different stages of bark beetle attacks at the individual tree level. The proposed method uses RetinaNet architecture (exploiting a robust feature extraction backbone pre-trained for tree crown detection) to train a shallow subnetwork for classifying the different attack stages of images captured by unmanned aerial vehicles (UAVs). Moreover, various data augmentation strategies are examined to address the class imbalance problem, and consequently, the affine transformation is selected to be the most effective one for this purpose. Experimental evaluations demonstrate the effectiveness of the proposed method by achieving an average accuracy of 98.95%, considerably outperforming the baseline method by $\sim 10\%$. The code and results are publicly available at <https://github.com/rudrakshkapil09/BarkBeetle-Damage-Classification-DL>.

I. INTRODUCTION

Bark beetle outbreaks significantly impact forests worldwide, thereby disrupting the functioning and properties of natural ecosystems. As a result of various factors (e.g., population density, tree moisture & condition, beetle & host tree species), a successful bark beetle attack gradually reveals itself by affecting various parts of the host tree [1]. Over time, the crown of an infested tree begins to fade – there is a gradual change in foliage color from a healthy green to yellow, red, and finally a leafless (i.e., needle-less) gray. These are referred to as different attack stages). The rate of discoloration depends on the progress of bark-beetle induced fungal infection that interrupts nutrient and water flow through the phloem and xylem, as well as environmental conditions such as soil moisture content [2]. The fading process is linked to the ecology of bark beetles (see Fig. 1), in which female bark beetles bore tunnels (called oviposition galleries) in the phloem to lay their eggs, and the larvae hatch and excavate additional larval galleries to feed on phloem tissue. If colonization is successful, the tree ultimately dies, and the next generation of beetles disperses from the parental tree in search of new hosts [3]. The detection of infested trees by *Dendroctonus mexicanus*

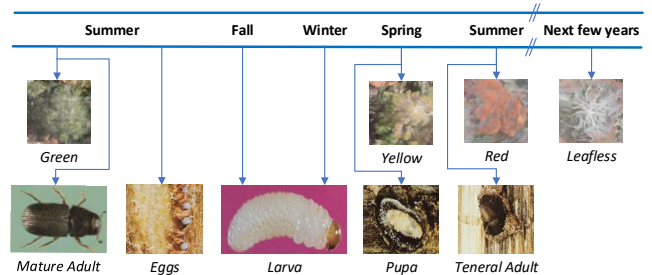


Fig. 1. Typical life cycle of bark beetles and their effect on host tree foliage over time. Beetle images have been adapted from [3].

is studied in this paper, which is among the most damaging insects to pine forests in Mexico.

Emerging bark beetles disperse in a number of ways in search of new hosts, with the majority partaking in short-range dispersal [4]. They fly below the forest canopy and attack suitable host trees within a few hundred meters. Hence, identifying attacked trees will help determine the next likely location of infestations and guide beetle management activities (e.g., sanitation, removal, or disposal) to prevent infestations from further spreading [5]. Automated systems can be designed to detect and analyze bark beetle infestations using remote sensing and machine learning (ML), avoiding labor- and cost-intensive efforts of traditionally employed ocular assessments. Although satellite and aircraft platforms are widely used at the landscape level, recent research has focused on leveraging UAVs for data collection due to their advantages at the individual tree level (e.g., higher spatial and temporal resolution). Besides, classical ML-based approaches (e.g., random forests (RF) or support vector machines (SVM)) require feature selection which demands prior experience and extensive effort to achieve satisfactory results. Thus, exploiting deep learning (DL)-based models is of interest due to their capacity for “learning” powerful representations and exhibiting good generalization through the discovery of intricate, underlying data patterns.

Although DL has achieved great success in a wide range of computer vision applications, a few recent studies have attempted to use DL-based models to classify infested trees from UAV-captured images [7], [8], [9]. Due to the lack of sufficient samples, these works either train customized shallow networks (i.w., six convolutional layers in [7], [9]) or apply

[§]Equal contribution.

Fig. 2. An overview of the proposed method. First, the backbone and feature pyramid network (FPN) are initialized using the DeepForest model [6] trained for tree crown detection. Following that, the network is modified and trained to classify the stages of bark beetle attack.

transfer learning to models pre-trained on ImageNet [10] (i.e., classification accuracy of infested trees in a temperate forest is investigated in [8] by training two shallow CNNs (with three pre-trained models has produced lower results potentially and six convolutional layers) and applying transfer learning to the frozen weights that were previously trained for generic pre-trained DenseNet [13]. Despite the availability of object classification purposes (e.g., cats vs. dogs). multi-spectral images from a DJI Mavic 2 Pro & 30 RTK, the best

This paper proposes a DL-based method that exploits results of this method are obtained using only RGB bands modified RetinaNet architecture (pre-trained for tree crown detection) to classify four infestation stages of trees attacked by bark beetles (see Fig. 2). The simple and efficient CNN models of AlexNet [14], SqueezeNet [15], VGGNet [11], ResNet [12], and DenseNet [13]. Using a DJI Mavic 2 Pro & 30 RTK and performs well in aerial imagery applications with dense DJI Phantom 4 Quadcopter, this method uses RGB images to targets. Moreover, the proposed method considers various detect treetops in a traditional manner and classify healthy and augmented strategies to alleviate the problem of limited gray-attacked trees.

number of samples and investigates their effects on network performance. Empirical evaluations demonstrate that the proposed method outperforms the baseline and classical methods considerably.

II. RELATED WORK

This section briefly describes DL-based approaches that seek to detect infested trees by different bark beetle species from UAV captured images (i.e., individual tree level). First, the potential of deep neural networks (DNNs) to detect bark beetle outbreaks in forests is studied in [7]. It employs a two-stage method consisting of a classical image processing based crown detection and a six-layer convolutional neural network (CNN) for predicting red- and gray-attacked trees by four-eyed bark beetles *Polygraphus proximus* Blandford, Coleoptera, Curculionidae. This method uses RGB images captured by a DJI Phantom 3 Pro quadcopter, and the performance is compared with six well-known CNN models (e.g., VGGNet [11], ResNet [12], and DenseNet [13]). After that, the

III. PROPOSED METHOD

Even though this task seems like a simple color classification, ill-defined attack labels and imbalanced datasets make it more challenging than it appears. For instance, the distribution and some challenging samples are visualized in Fig. 3, in which green and leafless (needle-less) classes overlap with other classes. Also, Fig. 4 shows the RGB color space histograms for each class that reveals similarities between the yellow and red attack stages due to the gradual nature of foliage discoloration.

Our proposed method is based on the RetinaNet architecture [16], which has been successfully used for other remote sensing applications (e.g., [17]) owing to its ability to detect dense targets from data with highly imbalanced classes. The proposed RetinaNet-based architecture includes the backbone network (i.e., ResNet [12]), feature pyramid network (FPN), classification subnetwork, and focal loss. Although

Fig. 3. RGB color space distribution of bark beetle dataset images. The borders of the highlighted challenging samples indicate their true labels. Fig. 4. Histograms showing RGB color space distribution of the different attack stages with leaves.

the backbone network seeks to extract multi-scale features, the FPN combines semantically low-resolution features with low-level, high-resolution ones. The classification subnetwork then predicts the category of bark beetle attacks (i.e., green (healthy) tree, yellow/red-attacked tree, or leafless) using focal loss. This loss function helps to simultaneously handle the inherent similarity of attack classes and limited-sized data by focusing on hard samples and avoiding easy negatives.

As shown in Fig. 2, the cropped images of tree crowns are normalized according to the mean and standard deviation of the training set images and then fed into the backbone network. The computations are forward propagated through the bottleneck layers, as well as being combined with the layers in the feature pyramid network. Each level of the pyramid feeds its computation to a classification subnetwork, each of which consists of four convolutional layers. Then the network outputs a score for each attack stage. At last, one-hot encoding is done to get the class prediction for each individual tree. In contrast to previous studies that train either a shallow network or deep models pre-trained on ImageNet [10], we exploit a pre-trained deep model (i.e., DeepForest [6] for tree crown detection) and train the modified network for the classification of attack stages. As a result of appropriately initializing network weights with features relevant to tree crowns, the classification subnetwork can learn to differentiate different bark beetle attack stages. Meanwhile, several different data augmentation strategies are considered to address the imbalance in the dataset. Although it is generally assumed that data augmentation will result in higher performance, we will show that blindly utilizing these techniques can drastically affect classification results for this task.

IV. EMPIRICAL EVALUATIONS

We evaluate the proposed method using the dataset presented in [18] that utilized a hexacopter with a Tarot 650 Pro to capture multiple RGB video sequences of a forested region in Northern Mexico from a top-down perspective. Five flights in total were conducted at three different average heights above ground (60 m, 90 m, and 100 m) during three months (June, July, and August). The individual frames from each flight were combined into five different orthomosaics and the ground truth information for each tree's center and attack stage were made available (see [18] for more details). The proposed method is compared with the baseline method [18] as well as the most promising SVM, RF, and K-nearest neighbors (KNN) classifiers. Hyperparameter tuning was done in a grid search manner for each classifier.

A. Implementation Details

We cropped individual tree crowns from the five orthomosaics as square patches and then split them into five separate training, validation, and testing sets, as shown in Table I. For each flight, one model is trained and evaluation is performed for each individually and averaged. The proposed networks

¹Bounding box regression subnetwork has been removed considering the available tree crown collections.

TABLE I
DATASET DISTRIBUTION FOR EACH FLIGHT BY CLASS.

| Subsets of Samples | Jun60 | Jul90 | Jul100 | Aug90 | Aug100 |
|--------------------|-------|-------|--------|-------|--------|
| Green Trees | 68 | 81 | 103 | 141 | 98 |
| Yellow Trees | 34 | 19 | 28 | 45 | 49 |
| Red Trees | 24 | 26 | 48 | 52 | 48 |
| Leafless Trees | 25 | 28 | 26 | 33 | 25 |
| Train | 128 | 130 | 174 | 230 | 187 |
| Augmented Train | 232 | 276 | 352 | 480 | 332 |
| Validation | 7 | 7 | 10 | 13 | 11 |
| Test | 16 | 17 | 21 | 28 | 22 |

were trained using the AdamW optimizer for 150 epochs and a batch size of 2 (five models for flights). The training procedure was performed on a Nvidia GeForce RTX 3090 GPU, with each model taking approximately 1.5 hours to train. The dataset was augmented by generating minority class samples using i) random affine warps, ii) vertical/horizontal flips, iii) 90°/180°/270° rotations, iv) cropping by a factor of 0.85, v) color jittering with random brightness, contrast, & saturation, and vi) Gaussian blurring with kernel size of 3. Furthermore, early stopping was considered to avoid overfitting during the training procedure.

B. Experimental Results

The experimental comparison of the proposed method (including the best model with affine warping data augmentation) with the baseline and best performing models for classical ML methods is shown in Table II. According to the results, the proposed method considerably outperforms 89% (& 7.6%) with (& without) data augmentation in average accuracy) the cellular automaton baseline method. Also, the classical ML methods have achieved significantly lower accuracy than our method, which can be explained by the ill-defined separation between classes in the RGB color space, as shown in Fig. 3. The confusion matrices for the challenging flights are shown in Fig. 5. Accordingly, the proposed method has no misclassifications for four of the flights, and only one leafless image is incorrectly predicted as red in the 100m flight due to the considerable overlap from nearby red attack stage trees. Since classic ML methods rely on manual feature selection, applying them directly to raw data results in poor performance. However, the proposed DL-based method can

TABLE II
COMPARISON OF CLASSIFICATION ACCURACY FROM VARIOUS MODELS.

| | Model | Average Accuracy (%) |
|----------------|-----------------------------|----------------------|
| | SVM | 53.10% |
| | KNN | 53.10% |
| | RF | 40.24% |
| | Baseline [18] (Best result) | 89% |
| | Ours (with warping) | 98.95% |
| Ablation Study | Ours (without augmentation) | 97.69% |
| | Ours (with cropping) | 96.29% |
| | Ours (with flips) | 94.74% |
| | Ours (with blurring) | 92.23% |
| | Ours (with rotation) | 84.71% |
| | Ours (with color jittering) | 83.90% |

