Leaf level Ash Dieback Disease Detection and Online Severity Estimation with UAVs

Bates, Elizabeth; Popović, Marija; Marsh, Conor; Clark, Ronald; Kovac, Mirko; Kocer, Basaran Bahadir

DOI
10.1109/ACCESS.2025.3541980

Publication date
2025

Document Version
Final published version

Published in
IEEE Access

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 11 January 2025, accepted 10 February 2025, date of publication 13 February 2025, date of current version 4 April 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3541980

RESEARCH ARTICLE

Leaf Level Ash Dieback Disease Detection and Online Severity Estimation With UAVs

ELIZABETH BATES[1], MARIJA POPOVIĆ[2], CONOR MARSH[3], RONALD CLARK[4], MIRKO KOVAC[1,5,6], (Member, IEEE), AND BASARAN BAHADIR KOCER[1,7]

[1]Aerial Robotics Laboratory, Department of Aeronautics, Imperial College London, SW7 2AZ London, U.K.
[2]MAVLab, Faculty of Aerospace Engineering, TU Delft, 2629 HS Delft, The Netherlands
[3]Department of Mechanical, Aerospace, and Civil Engineering, The University of Manchester, M13 9PL Manchester, U.K.
[4]Department of Computer Science, University of Oxford, OX2 6NN Oxford, U.K.
[5]Laboratory of Sustainability Robotics, Swiss Federal Laboratories for Materials Science and Technology, EMPA, 8600 Dübendorf, Switzerland
[6]Laboratory of Sustainability Robotics, École Polytechnique Fédérale de Lausanne, 1005 Lausanne, Switzerland
[7]School of Civil, Aerospace and Design Engineering, University of Bristol, BS8 1TR Bristol, U.K.

Corresponding author: Mirko Kovac (m.kovac@imperial.ac.uk)

The authors would like to thank all the Forest Research (FR) staff who provided feedback to their research. FR is the research agency of the Forestry Commission and Great Britain's principal organisation for forestry and tree-related research. This work was partially supported by funding from the Engineering and Physical Sciences Research Council (award no. EP/N018494/1, EP/R026173/1, EP/R009953/1, EP/S031464/1, EP/W001136/1), the SERI funded ERC Proteus consolidator grant (grant no: MB22.00066), the EU H2020 AeroTwin project (grant ID 810321), the Empa-funded DroneHub project, and the Empa-Imperial College London research partnership. Mirko Kovac was supported by the Royal Society Wolfson fellowship (RSWF/R1/18003). They would like to thank the other members of the Aerial Robotics Lab, Imperial College London, and the Laboratory of Sustainability Robotics, Empa/EPFL for their support and numerous stimulating discussions on this topic.

ABSTRACT Ash dieback, caused by the fungal pathogen Hymenoscyphus fraxineus, is devastating ash tree populations across U.K. and Europe, with projections indicating that up to 80% of ash trees may die as a result of the disease. The extensive loss of this keystone species threatens biodiversity and may lead to significant habitat degradation. Since no cure exists, early detection and removal of infected trees are critical to slowing the spread of the disease. Traditional identification methods rely on visual assessments of canopy loss, which are inefficient and impractical for large-scale monitoring. Leveraging advancements in computer vision and deep learning, our key objective is to develop a tool to detect ash dieback symptoms at the leaf level, classifying leaves into three categories: healthy, early-stage infection, and mid-stage infection. Since there is no known available dataset for ash dieback at the leaf level, we generated a new synthetic dataset and trained a YOLOv5 single-stage object detection model. The final model achieves mean Average Precision (mAP) scores of above 90% for each category. Evaluations on real ash tree leaf footage captured using uncrewed aerial vehicles (UAVs) show strong alignment between the model's detections and expert annotations. Our tool demonstrates the potential of integrating advanced computer vision techniques into tree health monitoring platforms. In the near future, this can provide conservationists and researchers with a novel, efficient means of early disease identification.

INDEX TERMS Environmental sensing, leaf disease severity estimation, ash dieback, computer vision.

I. INTRODUCTION

Trees play a vital role in maintaining the well-being of both humans and the environments they inhabit. When tree species are threatened and their populations decline, the consequences can be far-reaching, severely impacting ecosystem functions and the numerous species that depend on the trees [1]. This work focuses specifically on ash dieback (also known as chalara), a disease prevalent throughout Europe and the UK. Detecting ash dieback effectively in its early stages is challenging, as symptoms first appear on the leaves. Consequently, the most common method of detection, whether using computer vision techniques or by eye, is to assess tree canopy reduction once the disease has caused visible leaf loss. Previous works have accomplished this by using high-quality aerial imagery [2]. The use of uncrewed aerial vehicles (UAVs) enables close proximity to trees for gathering leaf imagery, providing an efficient and

The associate editor coordinating the review of this manuscript and approving it for publication was Aysegul Ucar.

cost-effective means of collecting high-resolution data [3]. However, a significant challenge in developing real-time deep learning-based leaf disease detection systems, such as those for ash dieback, lies in the availability of training data. Acquiring and annotating training data is costly due to the difficulties of accessing large-scale forests with complex terrains, the need for post-processing data, and the time and effort required for manual human labelling.

Recent work has explored synthetic data generation to overcome the issues of training data scarcity, as well as the high cost and time associated with data collection [4], [5]. In many application domains, including autonomous driving [6] and object pose estimation [7], synthetic data has facilitated model training and deployment. However, despite its potential, synthetic data generation for leaf disease detection remains an underexplored research area due to several unique challenges. First, the simulation environment must accurately capture the biological complexity of the disease in terms of leaf colour, texture, and shape. Second, sophisticated modelling and simulation techniques are needed to model the variability of symptoms, which depend on factors such as infection severity and environmental conditions. Finally, the research community lacks extensive publicly available real-world data to use as a reference for synthetic data creation.

To address these issues, we introduce a new synthetic dataset and deep learning-based methodology for ash dieback detection. Our goal is to leverage recent advances in synthetic data generation to fill the critical gap in training data for this problem. In our detection framework, we demonstrate how synthetic data can be exploited for accurate real-time ash dieback detection, including assessments of leaf health and disease severity. The leaf-level disease detection platform developed in [8] shows the ability to recognise and learn visually complex textures like bacterial spots and mildew, indicating that a multiclass infection severity model is a promising research direction. In this work, we propose the generation and use of a synthetic dataset and the You Only Look Once (YOLO) algorithm to create a valuable model for professionals in wildlife conservation, forestry, or environmental sensing robotics. To the best of our knowledge, it is the first work on a publicly available ash dieback leaf disease dataset generated with expert inputs and feedback by mimicking and diversifying the available samples in Unity from scratch.

II. RELATED WORK

A. ASH DIEBACK SYMPTOMS

Ash dieback causes a rapid decline in the health of ash trees, particularly in young trees and saplings. Key symptoms include necrotic patches on leaves, brown midribs, leaf discolouration, wilting, and tip dieback of branches. As the disease progresses, it extends along stems and branches, leading to canopy dieback. Dark, diamond-shaped lesions may appear on the trunk at branch junctions, eventually disrupting the tree's fluid supply or weakening the trunk through drying and cracking. The best time to identify ash dieback is from June to September to avoid confusion with seasonal changes [9]. There are many symptoms that can indicate the presence of chalara, with the initial visual indicators appearing on the leaves. Necrotic patches develop on the leaves and spread, whilst the midrib of the leaves slowly turns brown, and the natural green colour of the leaves shifts to yellow and brown [10]. These patches can also appear individually on the leaves, resembling brown spots that often extend into the leaf rachises (leaf stalks) [11].

B. CROWN-LOSS DETECTION

Recent research has explored methods for identifying ash trees infected with chalara by detecting crown loss, a common indicator of late-stage disease. High-quality aerial satellite imagery [12], [13] and real-time crown loss estimation models using UAV imagery and object detection have been developed [14], [15]. Another similar work with aerial imagery investigated structure from motion data to identify ash dieback at the crown loss level [16]. Other studies have investigated chalara detection using hyperspectral image clustering [17], [18] and combining aerial hyperspectral imaging and LiDAR for emerald ash borer detection [18]. LiDAR is also increasingly used in forest health monitoring for 3D information collection and individual tree segmentation [19]. However, to the best of our knowledge, there is currently no open leaf-level ash dieback dataset or severity level estimation framework available in this context.

C. USE OF UAVS

UAVs are increasingly used for environmental applications such as agricultural sensing and plant health monitoring [20], [21]. Advances in image resolution, data collection, deep learning, and GPU capabilities have made UAV-based identification, classification and segmentation tasks feasible and efficient [22]. Applications include seed planting [23], invasive plant monitoring [24], and wetland management [25]. UAVs can be equipped with various sensors to collect specific types of data [26] and can perform leaf-level information gathering and sample collection via perching [8], [27], [28]. Prior work has demonstrated UAV-based forest topology sensing, leaf sample collection, leaf infection detection, and crown loss estimation [8], [29]. However, most aerial imaging studies related to this problem have focused on ash dieback symptoms at the tree level [3], [14]. In this work, we extend the applicability of UAV-based sensing to the detection and classification of ash dieback, specifically at the leaf level.

D. OBJECT DETECTION

Detecting leaves with ash dieback symptoms using UAVs is an object detection task. Traditional methods relied on hand-crafted feature extraction [30], [31], [32], but convolutional neural networks (CNNs) have yielded more successful models [33], [34]. CNNs learn multi-level spatial features and outperform traditional methods [35], [36]. There are two main CNN-based approaches for object detection:

two-stage models (e.g. R-CNN) and single-stage models (e.g. YOLO, SSD) [37], [38], [39]. Single-stage models are faster, while two-stage models tend to have higher accuracy [38], [40]. Both have been applied to object detection in UAV-based footage with reasonable speed and high accuracy [8], [41]. Detecting chalara early at the leaf-level is ideal for minimising its spread. Existing work on leaf detection uses computer vision [42] and combines segmentation and classification with CNNs [43]. CNN-based models achieve high accuracy for leaf disease detection [8], [44], [45]. However, these methods are trained on datasets focused on crops (e.g. PlantVillage [46], DiaMOS [47], BRACOL [47], [48]) and lack specific ash tree leaves or chalara symptoms. A small dataset of Austrian broad leaf trees includes 25 healthy ash leaf images [49], but no extensive ash dieback dataset currently exists. In this work, we propose a new synthetic dataset to address this gap.

E. SYNTHETIC DATASETS

Generating synthetic data can overcome the limitations of real datasets, such as time-intensive collection and manual human labelling [4]. Techniques like ray and path tracing have improved realism [50], and software like Unity and Blender enable synthetic image creation. Combining synthetic and real data has been shown to generally boost model accuracy [6], [14], [51]. In some applications, significantly more synthetic data relative to real data is needed to achieve the same training results as with real data due to the lack of variation in synthetic datasets generated using traditional methods [5].

We propose a new synthetic dataset and a deep learning-based methodology that can detect and recognise ash tree leaves at various stages of chalara infection. Figure 1 shows an overview of our approach. Our framework takes images and video as input, allowing aerial footage of ash foliage to be analysed in real-time and with the class of ash dieback infection displayed. Classes of ash dieback severity have previously only been distinguished by canopy loss [52]; by the time that crown loss is evident on ash trees, the severity of the chalara infection is likely past its early stage. Our dataset is available from Zenodo[1] and the associated detection framework is available from GitHub.[2]

In this work, our model detects multiple classes of leaf-level ash dieback infections, presenting a novel chalara risk detection tool that could be used alongside existing canopy loss estimation platforms. Since our goal is to identify infections in real-time, we leverage a single-stage YOLO detection model to ensure fast inference times. It is hoped that this work could be helpful for conservationists and future research in this area, with the goal of efficient ash dieback detection and consequent action to slow the spread across the UK and all over the world.

[1]https://zenodo.org/records/13964157

[2]https://github.com/AerialRoboticsGroup/Leaf-level-Ash-Dieback-Disease-Detection-and-Online-Severity-Estimation

III. OVERVIEW AND APPROACH

A reliable dataset of leaf diseases is essential for accurate detection using deep learning-based models. To address this, we propose a new dataset for ash dieback detection. In contrast to previous work on real and synthetic datasets summarised in the former section, our dataset considers ash leaves at various infection levels. In addition, we exploit synthetic data to facilitate model training. A key aspect of our approach is a method for synthetic data generation based on expert input, which enables training an accurate detection model for this problem.

Knowing the chalara severity during detection is important for assessing the disease risk and informing the appropriate treatment procedure for the tree. To this end, we define three classes for disease detection: (i) healthy, (ii) early-stage, (iii) mid-stage foliar infection, as shown in Table 1 together with late-stage infection indicators. We developed this taxonomy based on the literature, as well as images and explanations from experts at Forestry Research. In particular, early-stage infection is indicated by dark speckles and early vein structure invasion, and mid-stage infection demonstrates significant discolouration and signs of rachis infection [11].

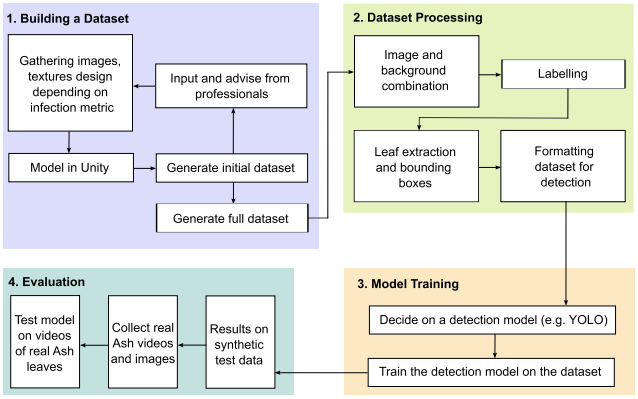

FIGURE 1. Our framework for ash dieback detection. First, we generate a dataset for ash dieback to train a deep learning-based detection model (YOLO). We supplement real data with synthetically generated disease images. We conduct ablation studies to validate our model choices for accurate disease detection and finally deploy it on real UAV imagery.

Fig. 1 summarises the organisation of the approach and each block is indicated with colours and addressed in separate sections. Building the dataset block is explained in Section III-A, dataset processing is explained in Section III-B, the model training block is given in Section III-C1 and the evaluation block is explained in Section IV.

A. BUILDING A DATASET

To create this dataset, we designed a synthetic forest environment with the Unity simulator with ash trees modelled at various stages of infection. To simulate as accurately as possible, high-quality tree textures at each stage were required. These can be extracted, colour corrected to reduce brightness and then overlaid to the ends of branches.

TABLE 1. Infection classes for leaf disease detection.

Disease level	Leaf description
Healthy	• Fully green, solid colour • Clear oval shape leaf • Green leaf midrib/veins
Early-stage infection	• Mainly green, some yellow parts (not solid colour) • Perhaps brown midrib/veins • Some brown spots present
Mid-stage infection	• Some green on leaf; at least a third of the leaf is brown • Brown veins/midrib • Shape possibly no longer oval • Partial leaf
Late-stage infection	• Almost all brown • Wilted, pointing downward • Not oval-shaped, starkly different to healthy leaves

The leaves were placed in a generally flat orientation, perpendicular to the main trunk of the tree, in order to copy the look of a sapling tree. We simulated a sapling tree as this was sufficient for generating images of the ash leaves, while a larger, more complex tree might obstruct the view of certain leaves.

FIGURE 2. Young ash tree with healthy leaves, synthetically generated in unity.

A preliminary image of the generated tree can be seen in Figure 2, with healthy ash leaves added. The textures used were that of healthy, early-stage infection and mid-stage infection, as seen in Figure 3. Late-stage infection, like that described in Table 1, differs significantly from the other infection levels in the shape, colour, and orientation of the leaves. Consequently, this class of infection could not be accurately modelled in a way that aligned with the literature or the images provided by the Forestry Research team. Therefore, we decided to limit the classes to only the first 3 levels: Healthy, early-stage and mid-stage infection, as these generated images closely resembled real ash leaves. After evaluating the types of real datasets available, we considered two options: either retaining the background or removing it to isolate the leaves. We decided to retain the background, like the images in the DiaMOS dataset [47], to emulate the environment that real aerial footage might capture.

FIGURE 3. Three selected examples of leaf textures used to create the leaves in Unity, including healthy (left) [49], early-stage infection (middle) and mid-stage infection (right) leaves (both acquired from email correspondence with Forest Research experts).

We created an initial dataset consisting of 50 leaf images from each class. These images were generated by photographing leaves at a single focal point on the tree, iteratively varying the angle and lighting. The image processing, detailed further in Section III-B1, required the images to be simple enough for clear separation of the foreground (leaves) and background. However, the initial images contained too many overlapping leaves, making it challenging for computer vision techniques to distinguish individual leaves. To address this issue, we enhanced the depth of field to focus on the leaves and altered the perspective to capture the leaves from a closer range. Each image now features a single stem with approximately 3-6 fully visible leaves as the main focus. This adjustment was applied to the leaves of each class, and examples from the final dataset are shown in Figure 4. Samples from this dataset were shared with Forestry Research experts for feedback and class confirmation. The final dataset comprises 15,000 images, evenly distributed with 5,000 images for each infection class.

FIGURE 4. Three sample images of the full dataset from Unity, healthy on the left, early-stage infection in the centre and mid-stage infection on the right.

B. DATASET PROCESSING

The dataset creation involved several stages: pre-processing, augmentation, labelling, and conversion into a format compatible with the YOLO model. For many object detection models, such as YOLOv5, the required format includes each image and a corresponding annotation text file. This file contains a numerical value representing the class of the object, along with the coordinates of the bounding box around the object, in the following order: x_centre, y_centre, width, and height. Initially, all images in the dataset were labelled. Given the nature of dataset generation, this process was straightforward: the first 5,000 images represent healthy leaves, the next 5,000 early-stage infections, and the final 5,000 mid-stage infections. Image IDs, their corresponding labels, and their location paths were stored in a data frame to facilitate random shuffling and splitting into training, validation, and test sets. The split was 70% training (10,500 images), 15% validation (2,250 images), and 15% testing (2,250 images). Given the relatively large dataset size, allocating 70% for training ensures computational efficiency while maintaining representativeness across the training, validation, and test sets.

1) IMAGE PROCESSING

To generate the bounding box information required for the YOLO format, the leaves in the images had to be extracted and enclosed within bounding boxes. A bounding box is a simple rectangle that closely encompasses the object of interest. The initial goal was to place bounding boxes around all the fully visible leaves, which typically resulted in 3-6 bounding boxes per image. The first step was to separate the main leaf structures from the background, producing a binary image with a black background and white leaves. The original RGB images were processed using k-means clustering to retain only pixels with intensities close to the darkest colour present. The images were then converted to the LAB colour space, excluding the 'luminosity' (L) component and focusing only on the A and B channels. Minimal frame blurring was then applied, followed by Otsu's thresholding method on both channels. This technique minimises intra-class intensity variance, effectively converting the images into binary format [53]. Closed contours in the mask were filled, resulting in clearly separated masks. We employed various image manipulation strategies to ensure that individual leaves formed distinct contours. In addition, we used morphological transformations, such as dilation and erosion. Dilation involves convolving the image with a kernel to enlarge the main contours, while erosion thins the contours by computing a local minimum over the kernel area. Although these methods worked for some images, the variety of angles and overlapping leaves in many dataset images often prevented proper isolation of individual leaves, as shown in Figure 5.

We also used the watershed algorithm to separate the leaves, with the intention of estimating the centre of the main

FIGURE 5. Example of 3 iterations of erosion (right) on the binary mask of an image from the dataset (left).

leaves by applying a distance transform [54]. An example of this applied to one of the images can be seen in Figure 6. However, it is evident that each leaf is not completely separated. Relying solely on these strategies would have led to incorrect segmentation across the dataset, resulting in inaccurate bounding boxes.

FIGURE 6. Example of the watershed algorithm applied to an image from the dataset (left). The outcome of the distance transform (middle) and the resulting image segmentation (right) can be seen.

Following these insights, we adjusted our approach so that instead of treating individual leaves as the primary objects, we applied a single bounding box to the main cluster of leaves present. The dataset was designed such that all leaves in each image shared the same label, allowing the model to potentially learn the shape and compound, pinnate pattern of ash leaves as a group. One significant advantage of this strategy was that it avoided the need to manually draw boxes around each leaf, a process that would have been extremely time-consuming given the scale of the dataset. However, grouping all leaves as a single large object introduced challenges, such as increased visual variation between images and a more complex nature for the detected 'object'. This variation was less pronounced when individual leaves were annotated. After training the model with these larger bounding boxes, testing on real video footage showed that the model often produced bounding boxes much larger than the actual leaf clusters, sometimes occupying most of the frame. This issue likely arose because the model learned to associate large bounding boxes with correct predictions, as this was the pattern it had been trained on. As a result, the dataset was reconstructed to reduce the size of the initial bounding box annotations, aiming for more accurate leaf detection.

2) BACKGROUND ADDITION TO DATASET

To address the issue of overly large bounding boxes, we modified the initial dataset so that the objects of interest

no longer occupy the majority of the image. First, a series of suitable background images were captured from a location with an abundance of ash trees. Next, a collection of images featuring ash foliage was taken to replicate the environment that an aerial robot might encounter when identifying ash dieback. Each image in the dataset was then resized, padded, and randomly placed onto one of the background images. The initial backgrounds of the smaller images were removed during this augmentation process. Because the original images already contained varying angles, no additional rotation was applied, only random translation. However, initial tests showed poor model performance when the original backgrounds were left intact. This was possibly due to the model detecting the sharp, square shape around the smaller masked images, as shown in Figure 7. The new dataset is no longer entirely synthetic, as the backgrounds are now real images and will partially appear within the bounding boxes. Using real, suitable backgrounds is likely more beneficial than generating synthetic ones, as it provides training images that more closely resemble future aerial footage.

FIGURE 7. Example of augmented dataset images with the initial background (left) and added synthetic sample (right).

3) BOUNDING BOXES AND ANNOTATION

A binary mask of the reduced and translated original image, created after separating the foreground and background, was used to identify the leaves and correctly position a bounding box. The contours of this mask were sorted by size, with the largest contour corresponding to the extracted leaves. A bounding box was then placed around this contour, and the coordinates were converted to the YOLO format. However, a challenge with this approach was that some images contained leaves that were not entirely in the frame, resulting in many images featuring partial leaves alongside complete ones. The implications of this are discussed in Section IV-C. After this step, all new images in the dataset were sorted into either the training, validation, or test folders. Each image was accompanied by a text file of the same name (image ID) containing class labels and bounding box information, preparing the dataset for the YOLO training process.

C. MODEL TRAINING

Our pipeline is agnostic to the model used for leaf detection. As a demonstrative use-case, we employ the YOLOv5 for both training and evaluation against the synthetic test data. We found that single-stage detector models, such as those using the YOLO algorithm, deliver high accuracy and precision with faster inference times. While two-stage models often outperform single-stage ones, the lower computational requirements of a model like YOLO make it more suitable for mobile or smaller devices. This is particularly valuable for conservation work, where real-time detection could enhance fieldwork efficiency. Our former work also used the tiny YOLOv3 model [8], training on a dataset of 6,000 images and achieving high mAP scores on their test data. This influenced the decision to use a YOLO model in this study, along with the opportunity to explore a more recent version and a larger architecture.

1) YOU ONLY LOOK ONCE (YOLO) MODEL

The YOLO algorithm is widely recognised in the fields of computer vision and machine learning as a fast and accurate single-stage detector. It divides the input image into an $S \times S$ grid and approaches object detection more like a regression task than a classification task [39]. Each grid cell predicts the object presence and bounding box properties. If the predicted bounding box values exceed a specified Intersection over Union (IoU) threshold, they are displayed [55]. The object score and class probability are calculated using the binary cross-entropy loss with logits, a loss function from PyTorch. This loss function plays a crucial role in teaching the model to accurately identify ash leaves at each infection stage, the equation is provided in (1).

$$\begin{aligned}
&\lambda_{\text{coord}} \sum_{i=0}^{s^2} \sum_{j=0}^{B} M_{ij}^{\text{object}} \left[\left(x_i - \hat{x}_i\right)^2 + \left(y_i - \hat{y}_i\right)^2 \right] \\
&+ \lambda_{\text{coord}} \sum_{i=0}^{s^2} \\
&\sum_{j=0}^{B} M_{ij}^{\text{object}} \left[\left(\sqrt{w_i} - \sqrt{\hat{w}_i}\right)^2 + \left(\sqrt{h_i} - \sqrt{\hat{h}_i}\right)^2 \right] \\
&+ \sum_{i=0}^{s^2} \sum_{j=0}^{B} M_{ij}^{\text{object}} \left(C_i - \hat{C}_i\right)^2 \\
&+ \lambda_{\text{noobject}} \sum_{i=0}^{s^2} \sum_{j=0}^{B} M_{ij}^{\text{noobject}} \left(C_i - \hat{C}_i\right)^2 \\
&+ \sum_{i=0}^{s^2} M_i^{\text{object}} \sum_{\text{c } \epsilon \text{ classes}} \left(p_i(c) - \hat{p}_i(c)\right)^2 \qquad (1)
\end{aligned}$$

where B is the number of predicted bounding boxes in each grid cell, s is the number of cells in the image, c is the class prediction for each grid cell and $p(c)$ represents the confidence probability. λ_{coord} and $\lambda_{\text{noobject}}$ are the weights to decide the importance of localisation against recognition during the training. In this equation, any box j of cell i, x_{ij} and y_{ij} represent the coordinates of the centre of the anchor box. The variable h gives height, w gives the width of the box, M denotes if an object appears in a grid cell and

C provides the confidence score [55]. The architecture for the YOLOv5 consists of a backbone, neck, and head. The backbone contains a CNN that aggregates image features at various granularities, the neck consists of many layers that mix image features and combine them for the prediction phase, and the head uses the features to begin the class and bounding box prediction stage.

The main activation function used in the YOLOv5 architecture is the Sigmoid Linear Unit (SiLU) function [56], also known as the swish function, which nonlinearly interpolates between the linear function and the ReLU function [57].

In terms of progression, the YOLO models have improved significantly from the initial one. YOLOv2 performed better due to the introduction of batch normalisation, as well as an updated backbone architecture. It previously used GoogLeNet [58], but replaced it with DarkNet-19 [59]. YOLOv3 updated the network for feature extraction again, increasing the 19 convolution layers in YOLOv2 to a 53-layered network, called Darknet-53 [60]. This network outperforms the previously more accurate ResNet-152 network, which is substantially slower and does not use the GPU as efficiently as Darknet-53. YOLOv4 was a bigger step up, utilising methods like ''bag of freebies'', regularisation, class label smoothing and self-adversarial training to name a few. YOLOv5 was released soon after v4 and it seems the main difference is that the anchor box selection process is integrated into the model. Since 2022, many more YOLO models have been released including the recent YOLOv11 released in late September 2024. Changes since YOLOv5 include architectural adjustments like the addition of an 'EfficientRep Backbone' in YOLOv6 and Extended Efficient Layer Aggregation Networks (E-ELAN) in YOLOv7 to increase model performance. The most recent versions continue to improve incrementally with enhancements like the ability to reduce model parameters and FLOPs and the migration to an anchor-free model such that the centre of the object is predicted, rather than the offset [61]. YOLOv5 was used in this work since it was the most simple to implement given the structure of our dataset and its performance on the benchmarking MS COCO Object Detection Dataset is sufficiently high [62] to justify the trade off in using a more recent model. However, this methodology with the synthetic dataset is agnostic to the object detection model, including any version of YOLO.

2) TRAINING IN CONTEXT

There are various sizes of YOLOv5 architectures, depending on the speeds, input image sizes, and the number of model parameters. In this work, the YOLOv5m model was used so as to maintain quick inference times and reasonable training times. Our specific set of model hyperparameters is shown in Table 2 and most of the default parameters are used in this work. The larger YOLOv5l was tested too, but this took much longer to train and produced mAP values that were very similar to the smaller model. The training was conducted using a T4 GPU, the batch size was 16, and the model was trained for 50 epochs. This number of epochs seemed appropriate since the values of precision and mAP began to plateau around the 40 epoch mark. After training, the model weights were saved as an ONNX file so they could be used locally in a software environment.

TABLE 2. Main hyperparameters table for the YOLO model training.

Hyperparameters	Values
Learning rate (α_i)	0.01
Batch size	16
Epochs	50
Weight decay (η_e)	0.0005
IoU training threshold (η_i)	0.2

IV. RESULTS, EVALUATION, AND DISCUSSION

With our work, we can consider two types of testing. The mAP, precision and recall values are good indicators of model performance, but this is strictly measuring how well the 15% test portion of the synthetic data is being predicted. Since the dataset is entirely synthetic in terms of the generated infected leaves, it is critical that we test the model on real images of ash tree leaves in each of the three classes.

TABLE 3. Results table for the synthetic data test.

Class	Precision	Recall	mAP at 0.5	mAP at 0.5:0.95
All	0.973	0.979	0.988	0.963
Healthy	0.998	0.998	0.995	0.995
Early-Stage	0.996	0.996	0.995	0.974
Mid-Stage	0.924	0.941	0.975	0.922

A. SYNTHETIC TEST RESULTS

The model produced high mAP values on average over all the classes, at 0.963 when averaged over different IoU thresholds from 0.5 to 0.95. The values seen in Table 3 are the results from the best epoch; throughout the training, the weights of the current best model are saved iteratively. Precision is useful to know in this context since it measures how many of the predictions made by the model were correct. Recall, however, is more about measuring how well the model can detect positive samples. The average precision is the area under the precision and recall curve, and so the mAP is the average of AP. These three metrics improved over the 50 epochs of training. The loss values for class, object and bounding box sharply decrease in the first 5 epochs and continue to decrease at a slower rate. Class loss and object loss plateau more so than object loss and are also lower throughout the training process, indicating that binary cross entropy loss is consistently the largest contributor to the total loss. The confusion matrix seen in Figure 8 indicates the

model was able to consistently identify the right labels for both healthy and low-level infection leaves, and with slightly less success for the mid-level infection leaves. Whilst this confusion matrix is encouraging, it is only demonstrating the success of the model on the synthetic dataset and it is likely that this strong correlation would not be observed on a test set of real leaves.

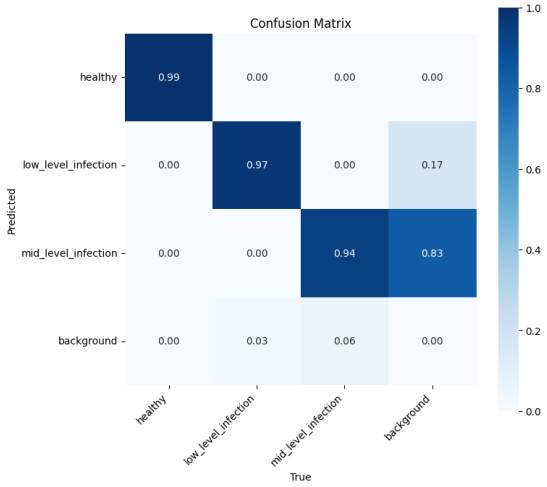

FIGURE 8. Confusion matrix of the model using the test dataset.

B. REAL TEST RESULTS

In order to evaluate the ability of the model to recognise leaves and make predictions on real videos of leaves, we acquired real video footage of ash leaves, e.g. shot from an aerial robot in close proximity to tree foliage. The details on the aerial robot vision pipeline can be seen in [14]. To test our proposed framework, we integrated them into a DJI Tello UAV, which is a palm-sized aerial platform weighing less than 250 g. We employed a ground computer to provide internet connectivity to the UAV, enabling remote data streaming. Online data streaming and inference were conducted on the ground computer and we achieved near real-time processing at approximately 20 frames per second.

We performed the experiments in an area with ash trees, mainly saplings as these were within reach, recording multiple videos in the field were taken. A variety of videos were taken, some with just a few leaves, and some more complicated ones with lots of foliage, to test how the detector responds. Videos containing healthy, early-stage, and mid-stage leaves were also taken, in order to see whether all three classes can be identified. When these videos are run through the developed model, the results are encouraging, there are many detections and they seem appropriate according to the metric described in Table 1, though these are observational opinions. Random stills from various videos were taken, and these were sent to the Forestry Research experts for their opinion on the infection stage of the leaves in each frame. The detection results of the model on these same frames were found, and the results can be seen in Figure 9. The detector agrees with the expert opinion on most of the frames, and those that did not align were only a single class apart. However, our results in the confusion matrix illustrated in Fig. 8 are inconclusive. Since the current bottleneck in this research is the lack of real dataset availability, there is a need to test and validate the proposed approach with a real dataset.

C. DISCUSSION

The results from the synthetic test set are encouraging, with high mAP, precision and recall values across all the classes. The lowest mAP value is 0.922 for mid-stage detection. It is clear that the model detects healthy leaves the best, and then the early-stage, followed by the mid-stage. This could be for many reasons; the healthy leaves are slightly brighter than the other classes, potentially making them clearer or more recognisable. There is also less visual variation in the texture of healthy ash leaves, whilst the symptoms of early and mid-stage infected leaves can differ more in colour, texture and placement. We also observe that some of the backgrounds of the initial leaf images for the mid-stage infection were not always entirely removed, along with partial leaves being removed, see Figure 10. We anticipate that this had an effect during the training and was one of the causes for the lower precision, recall, and mAP values for this particular class.

Testing the model on videos of real ash leaves was necessary to understand whether the synthetic dataset can provide enough detail and information. This is also important to verify that our pipeline is a useful tool for tackling the ash dieback problem in real-world scenarios. The small sample of frames used to cross-check the model's prediction with that of the expert is not comprehensive enough to be conclusive about the model's performance on real footage, though it does demonstrate some positive results and that the synthetic dataset was close enough visually that the model functions with real data inputs. There is a level of subjectivity worth considering about the experts' opinions on the severity of the leaf infections in Figure 9; it was expressed in their feedback on these images that some were borderline in their categories and difficult to label. Images 3, 5 and 11, for example, are all identified as mid-stage by the detector, perhaps due to the lighter green foliage making the infection spots more prominent in contrast. Distinguishing an obvious difference in the image, for example, image 9 which was labelled mid-stage by the experts is not a clear-cut task even by eye. Future research into applying hyperspectral imaging to provide more information on the appearance and patterns of infections at different levels could help tackle such issues.

The model tended to pick up at most 5 or 6 leaves in a frame, and usually only 1. Not all identifications are correct, but the ones that are untrue usually only differ by one class; healthy and low-level infection classes seem to be mislabelled most frequently. This is likely partially due to these two classes being the most present in the collected videos. Interestingly, the model most often detects single leaves and sets of 2 or 3 leaves grouped together, despite the annotations

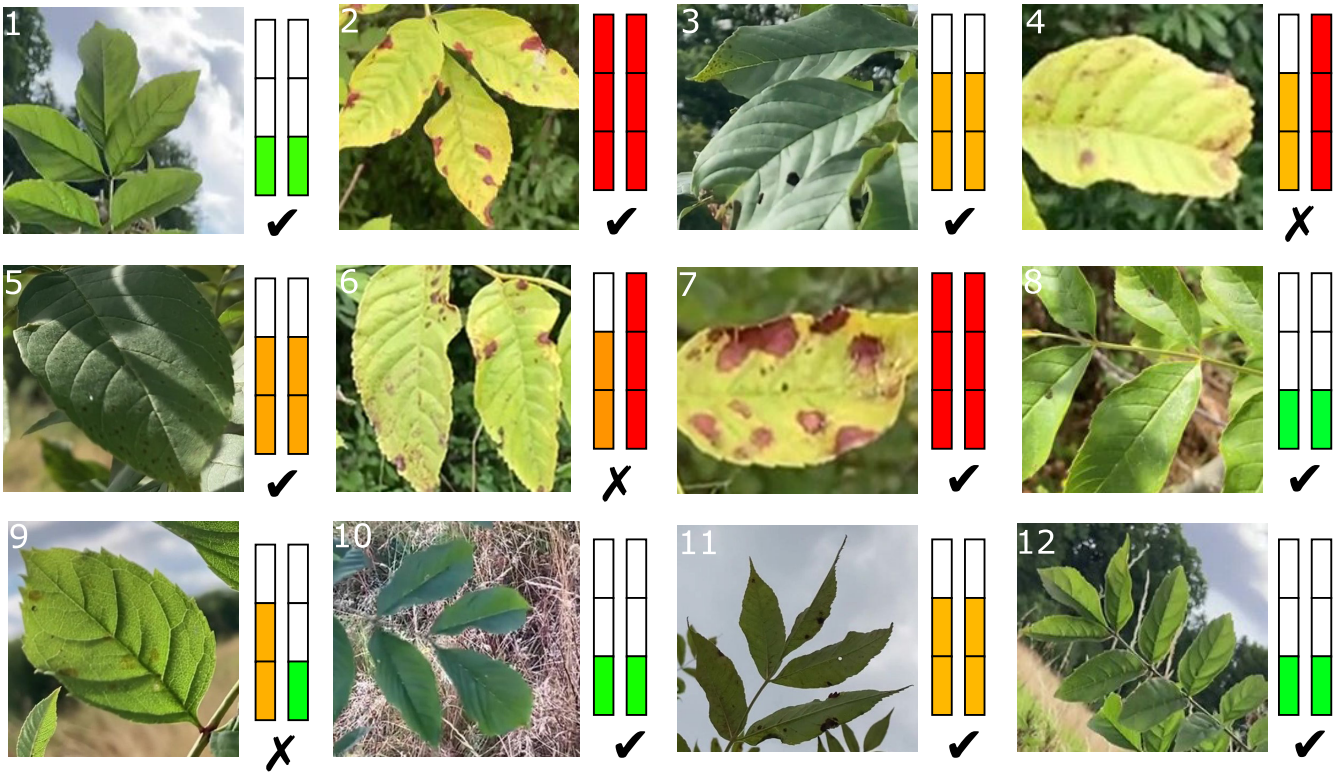

FIGURE 9. Comparison of expert opinions of leaf infection stage vs the model's prediction. The left bar shows the expert's assessment, right bar shows the model's prediction. The bar indicates [healthy, early-stage, mid-stage] of the disease.

FIGURE 10. One of the lower quality images from the dataset, including partial leaves and some of the background unremoved (left). Incorrect prediction of mid-stage infection leaves from a distance (right).

of the bounding boxes during training being exclusively around groups of approximately 5 leaves. We hypothesise that these single leaves are frequently identified since the model learned during training that these features were critical to a correct prediction. The initial aim was to make a leaf-level infection detector; the model's ability to predict both single leaves and small groups of leaves is a positive outcome that remains useful when detecting ash dieback from aerial level footage. The model also seems to frequently place boxes around partial leaves or leaves on the edges of the frames. This could be because when the background is removed from each of the generated images, the foreground often contains both full and partial leaves within the frame. The bounding box is placed around the main foreground contour, which frequently includes these partial leaves. It is likely the model has also learned to associate these with each class label.

A limitation of this model is that it often incorrectly classifies objects as mid-stage infection leaves when the video input is zoomed out, and the leaves and foliage are much smaller, like in Figure 10. Additionally, we observed that occasionally dead leaves or those in the late-stage of infection were also detected as mid-stage. These misclassifications may be due to the similar texture of the mid-stage infection leaves and that of dark, small or late-stage infected leaves. In terms of the detection of other leaf species, it was unusual for the model to predict the healthy or early-stage classes, however, some instances of mid-stage infections were predicted. The addition of more textural samples of mid-stage infection leaves to the generated dataset would likely improve the model's performance, along with the addition of real images to train on.

The model could place multiple bounding boxes at a time. These were often correct, and the placement and size were frequently true to the leaves being detected. The detector also worked in different lighting conditions and tested on videos from both a very sunny and a more overcast day. The variety of lighting and angles provided in the generated dataset meant that leaves could be detected from different viewpoints and in shady areas as well as bright ones. The next step in regard to testing the performance would be to collect a larger sample of frames for expert opinion comparison. To fully assess the

generalisability of the model, our future work aims to test it on a real dataset created with support from field researchers.

D. COMPARISON DISCUSSION

The PlantVillage is a popular dataset for leaf disease detection, comprising of roughly 54,000 high-quality images that capture a range of health states and diseases across various crop species [46], [47]. Despite its extensive scope and quality, the dataset is unsuitable for our research as it does not include ash leaves. Furthermore, the images in PlantVillage feature individual leaves set against plain, uniform backgrounds. This poses a challenge for real-world applications where the backdrop is often complex, involving varied colours, textures, and overlapping leaves.

The DiaMOS plant dataset comprises approximately 3,000 leaf images and 500 fruit images from pear plants. These images display a range of symptoms, including spot, curl, and slug infestation, in addition to healthy leaves. However, it is important to highlight the dataset's class imbalance: over 2,000 images pertain to slug infestation, while leaf curl is represented by a meagre 54 images [47]. Training a model on this dataset would disproportionately excel at detecting slug infestation at the expense of other conditions. To address this issue, the original DiaMOS study employed data augmentation techniques [47]. Another distinctive feature of this dataset is that it incorporates different severity levels, gauged by the infected leaf area percentage. Uniquely, the images are captured in situ, encompassing varied lighting conditions, angles, and complex backgrounds. This contextual diversity provides an edge for real-world applicability, as the model becomes acclimated to conditions closely resembling actual pear leaf environments.

The BRACOL dataset centres on the Coffea arabica plant, featuring close to 2,000 leaf images that depict diseases such as brown leaf spot, leaf rust, and leaf miner [47], [48]. While smaller in scope compared to datasets like PlantVillage, BRACOL's focus on a single plant species lends it specificity for targeted disease detection. However, its utility is curtailed by the choice of a white, homogeneous background for all leaf images, a limitation it shares with the PlantVillage dataset. This design decision hampers the model's performance in naturalistic settings where leaves are viewed against diverse backgrounds. Other datasets like the plant pathology dataset and the Citrus dataset both specialise in fruit tree leaf infections but diverge in their imaging approaches. Specifically, the plant pathology dataset captures leaves in their natural habitat, enhancing its ecological validity for real-world applications [63]. In contrast, the Citrus dataset, akin to PlantVillage and BRACOL, opts for monochromatic backgrounds, potentially limiting its generalisability [47], [64].

Apart from the expansive PlantVillage dataset, most available datasets are notably smaller in size. This limitation is understandable given the considerable time and expertise required to amass and accurately label thousands of leaf images. However, the availability of large, high-quality datasets is instrumental for training robust models. Moreover, the specificity of the data–in terms of both the leaf species and the targeted disease–is critical. While many leaf diseases exhibit overlapping symptoms, models trained on precise and specific datasets are likely to yield the most reliable and accurate results.

The TensorFlow 'plant_leaves' dataset comprises of 4,502 high-quality images spanning 12 distinct species [65]. Each image features a centrally-placed leaf or leaflet against a dark, uniform background. The dataset is rich in diversity, capturing a range of healthy leaves as well as various disease symptoms like spots and necrotic sections, which are also indicative of ash dieback. Future research on ash dieback could leverage this general leaf dataset as an initial training ground for a potential generative model. Because of having similar features, it is also possible to make use of the 'plant_doc' dataset [66], a comprehensive collection of 2,598 images that cover 13 plant species and 17 disease categories, for our comparison studies.

The creation and annotation of extensive datasets for machine learning are both labour-intensive and potentially costly endeavours, particularly when expert labelling is required for specialised data. Synthetic data generation presents a viable alternative to alleviate these challenges by enabling the creation of large, customised datasets without manual labeling [4]. However, for visual tasks like disease detection, additional metadata such as bounding boxes may still be necessary for effective training. Automating this aspect can mitigate some costs but also presents challenges. The primary obstacle in the usage of synthetic data lies in generating photorealistic models and textures that accurately represent both healthy and diseased leaves, a task that currently defies easy automation.

Even though there are many ways of generating photorealistic synthetic samples and synthetic datasets are a valuable training resource, it remains essential to validate model performance on real-world data in addition to synthetic test sets.

Recent studies indicate that combining real and artificial image data can significantly improve model accuracy [6], [51], while the exclusive use of synthetic data also holds promise [4]. For instance, [5] explored this approach in the context of urban and traffic scenes, finding that the inclusion of synthetic data alongside real images enhanced the model's performance. Moreover, their results suggest that a substantially larger volume of synthetic data is needed to achieve comparable performance when real data is not part of the training set. Given the limited availability of ash tree dieback-specific images, our new synthetic dataset presents a valuable asset for advancing research in this domain.

It would be beneficial to collect a substantial amount of real data to add to the generated synthetic dataset; many studies demonstrated that a combination of real and generated data yields good results [6], [51]. Various augmentation techniques could even be applied to the real data collected

to further expand the number of images to train on. Alternatively, a smaller amount of pictures could be taken to collect more textures for the leaf generation process in Unity. This would have the effect of diversifying the dataset further, which would be expected to improve the prediction performance, especially for the more complex mid-stage infection class. Taking more images of late-stage infected leaves and changing the modelling method in Unity could also make incorporating this class into the detector possible. This extends the applicability of our tool.

The Forestry Research experts consulted during this work provided insight into symptoms and developmental patterns. It was explained that a large proportion of symptomatic leaves are prematurely shed at various stages of infection. Future work could perhaps explore this point further, and whether ground evaluation to identify fallen infected leaves as well as those present in the canopy could help identify trees in the early stages of ash dieback.

V. CONCLUSION

In this work, we introduce a synthetic image dataset to train a detector for recognising and identifying ash tree leaves at various stages of infection. The multiclass dataset of infected ash leaves is novel in that it is the only dataset of its kind with accompanying annotations and a large number of images, sufficient for training a deep convolutional neural network. The model built is capable of being used on real videos of ash leaves despite being trained on solely synthetic data. The detector enables assessing the infection severity of ash leaves that are suffering from chalara, a tool that does not currently exist to the best of the authors' knowledge. We evaluated this model using real video images. Although our study is not extensive, it shows promising results and validates the value of our contribution in the area of tree health monitoring using aerial robotics.

A. FUTURE WORK

The results of this work are encouraging considering the small, but still noticeable gap, between synthetically generated leaf images and real ones. There are many avenues worth exploring in future work to improve the performance of this tool on real aerial footage of ash dieback leaf infections. In future work, we intend to close this gap between the synthetic leaf images and real ones using generative adversarial networks (GANs) or diffusion models. We expect this to improve the results of the model when retrained on more realistic images. We are also exploring potential ways [67] to segment the diseased leaves at a finer scale.

ACKNOWLEDGMENT

The authors would like to thank all the Forest Research (FR) staff who provided feedback to their research. FR is the research agency of the Forestry Commission and Great Britain's principal organisation for forestry and tree-related research.

REFERENCES

[1] R. J. Mitchell et al., "Ash dieback in the U.K.: A review of the ecological and conservation implications and potential management options," *Biol. Conservation*, vol. 175, pp. 95–109, Jul. 2014.

[2] M. Gašparović, I. Pilaš, D. Klobučar, and I. Gašparović, "Monitoring ash dieback in Europe—An unrevealed perspective for remote sensing?" *Remote Sens.*, vol. 15, no. 5, p. 1178, Feb. 2023.

[3] S. K. Valicharla, X. Li, J. Greenleaf, R. Turcotte, C. Hayes, and Y.-L. Park, "Precision detection and assessment of ash death and decline caused by the emerald ash borer using drones and deep learning," *Plants*, vol. 12, no. 4, p. 798, Feb. 2023.

[4] D. Schraml, "Physically based synthetic image generation for machine learning: A review of pertinent literature," *Proc. SPIE*, vol. 11144, pp. 108–120, Sep. 2019.

[5] V. Seib, B. Lange, and S. Wirtz, "Mixing real and synthetic data to enhance neural network training—A review of current approaches," 2020, *arXiv:2007.08781*.

[6] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3234–3243.

[7] J. Tremblay, T. To, and S. Birchfield, "Falling things: A synthetic dataset for 3D object detection and pose estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 2119–21193.

[8] B. B. Kocer, B. Ho, X. Zhu, P. Zheng, A. Farinha, F. Xiao, B. Stephens, F. Wiesemüller, L. Orr, and M. Kovac, "Forest drones for environmental sensing and nature conservation," in *Proc. Aerial Robotic Syst. Physically Interacting With Environ. (AIRPHARO)*, Oct. 2021, pp. 1–8.

[9] *Ash Dieback (Hymenoscyphus Fraxineus)*. Accessed: Jun. 7, 2024. [Online]. Available: https://www.forestresearch.gov.uk/tools-and-resources/fthr/pest-and-disease-resources/ash-dieback-hymenoscyphus-fraxineus/

[10] F. Research. *Forest Research Rapid Assessment*. Accessed: Jun. 7, 2024. [Online]. Available: https://www.daera-ni.gov.uk/sites/default/files/publications/dard/chalara-fraxinea.pdf

[11] Observatree. (2016). *Field Identification Guide*. Accessed: Jun. 7, 2024. [Online]. Available: https://www.observatree.org.uk/resource-library/field-identification-guides/

[12] A. N. Skurikhin, S. R. Garrity, N. G. McDowell, and D. M. Cai, "Automated tree crown detection and size estimation using multi-scale analysis of high-resolution satellite imagery," *Remote Sens. Lett.*, vol. 4, no. 5, pp. 465–474, May 2013.

[13] F. H. Wagner, M. P. Ferreira, A. Sanchez, M. C. M. Hirye, M. Zortea, E. Gloor, O. L. Phillips, C. R. de Souza Filho, Y. E. Shimabukuro, and L. E. O. C. Aragão, "Individual tree crown delineation in a highly diverse tropical forest using very high resolution satellite images," *ISPRS J. Photogramm. Remote Sens.*, vol. 145, pp. 362–377, Nov. 2018.

[14] B. Ho, B. B. Kocer, and M. Kovac, "Vision based crown loss estimation for individual trees with remote aerial robots," *ISPRS J. Photogramm. Remote Sens.*, vol. 188, pp. 75–88, Jun. 2022.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[16] W. Flynn, S. Grieve, A. Henshaw, H. Owen, R. Buggs, C. Metheringham, W. Plumb, J. Stocks, and E. Lines, "UAV-derived greenness and within-crown spatial patterning can detect ash dieback in individual trees," *Ecolog. Solutions Evidence*, vol. 5, no. 2, 2024, Art. no. e12343.

[17] A. H. Y. Chan, C. Barnes, T. Swinfield, and D. A. Coomes, "Monitoring ash dieback (*Hymenoscyphus fraxineus*) in British forests using hyperspectral remote sensing," *Remote Sens. Ecol. Conservation*, vol. 7, no. 2, pp. 306–320, Jun. 2021.

[18] S. L. Polk, A. H. Y. Chan, K. Cui, R. J. Plemmons, D. A. Coomes, and J. M. Murphy, "Unsupervised detection of ASH dieback disease (*Hymenoscyphus fraxineus*) using diffusion-based hyperspectral image clustering," in *Proc. IEEE Int. Geosci. Remote Sens. Symp.*, Jul. 2022, pp. 2287–2290.

[19] Q. Zhou, L. Yu, X. Zhang, Y. Liu, Z. Zhan, L. Ren, and Y. Luo, "Fusion of UAV hyperspectral imaging and LiDAR for the early detection of EAB stress in ash and a new EAB detection Index—NDVI(776,678)," *Remote Sens.*, vol. 14, no. 10, p. 2428, May 2022.

[20] B. B. Kocer, H. Stedman, P. Kulik, I. Caves, N. Van Zalk, V. M. Pawar, and M. Kovac, "Immersive view and interface design for teleoperated aerial manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2022, pp. 4919–4926.
[21] L. Romanello, T. Lan, M. Kovac, S. F. Armanini, and B. B. Kocer, "Exploring the potential of multi-modal sensing framework for forest ecology," 2024, *arXiv:2410.23033*.
[22] M. M. D. Oghaz, M. Razaak, H. Kerdegari, V. Argyriou, and P. Remagnino, "Scene and environment monitoring using aerial imagery and deep learning," in *Proc. 15th Int. Conf. Distrib. Comput. Sensor Syst. (DCOSS)*, May 2019, pp. 362–369.
[23] J. Castro, F. Morales-Rueda, D. Alcaraz-Segura, and S. Tabik, "Forest restoration is more than firing seeds from a drone," *Restoration Ecol.*, vol. 31, no. 1, Jan. 2023, Art. no. e13736.
[24] N. C. de Sá, P. Castro, S. Carvalho, E. Marchante, F. A. López-Núñez, and H. Marchante, "Mapping the flowering of an invasive plant using unmanned aerial vehicles: Is there potential for biocontrol monitoring?" *Frontiers Plant Sci.*, vol. 9, Mar. 2018.
[25] I. Dronova, C. Kislik, Z. Dinh, and M. Kelly, "A review of unoccupied aerial vehicle use in wetland applications: Emerging opportunities in approach, technology, and data," *Drones*, vol. 5, no. 2, p. 45, May 2021.
[26] J. Jiménez López and M. Mulero-Pázmány, "Drones for conservation in protected areas: Present and future," *Drones*, vol. 3, no. 1, p. 10, Jan. 2019.
[27] F. Hauf, B. B. Kocer, A. Slatter, H.-N. Nguyen, O. Pang, R. Clark, E. Johns, and M. Kovac, "Learning tethered perching for aerial robots," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2023, pp. 1298–1304.
[28] T. Lan, L. Romanello, M. Kovac, S. F. Armanini, and B. B. Kocer, "Aerial tensile perching and disentangling mechanism for long-term environmental monitoring," 2024, *arXiv:2403.01890*.
[29] R. Dainelli, P. Toscano, S. F. Di Gennaro, and A. Matese, "Recent advances in unmanned aerial vehicles forest remote sensing—A systematic review. Part II: Research applications," *Forests*, vol. 12, no. 4, p. 397, Mar. 2021.
[30] Y.-Q. Wang, "An analysis of the viola-jones face detection algorithm," *Image Process. Line*, vol. 4, pp. 128–148, Jun. 2014.
[31] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 886–893.
[32] T. Lindeberg, "Scale invariant feature transform," *Scholarpedia*, vol. 7, no. 5, 2012, Art. no. 10491, doi: 10.4249/scholarpedia.10491.
[33] B. Dey, M. M. Ul Haque, R. Khatun, and R. Ahmed, "Comparative performance of four CNN-based deep learning variants in detecting Hispa pest, two fungal diseases, and NPK deficiency symptoms of Rice (*Oryza sativa*)," *Comput. Electron. Agricult.*, vol. 202, Nov. 2022, Art. no. 107340.
[34] B. Dey, J. Ferdous, R. Ahmed, and J. Hossain, "Assessing deep convolutional neural network models and their comparative performance for automated medicinal plant identification from leaf images," *Heliyon*, vol. 10, no. 1, Jan. 2024, Art. no. e23655.
[35] P. Bharati and A. Pramanik, "Deep learning techniques—R-CNN to mask R-CNN: A survey," in *Computational Intelligence in Pattern Recognition*. Singapore: Springer, Aug. 2019, pp. 657–668.
[36] T. Kattenborn, J. Leitloff, F. Schiefer, and S. Hinz, "Review on convolutional neural networks (CNN) in vegetation remote sensing," *ISPRS J. Photogramm. Remote Sens.*, vol. 173, pp. 24–49, Mar. 2021.
[37] P. Soviany and R. T. Ionescu, "Optimizing the trade-off between single-stage and two-stage deep object detectors using image difficulty prediction," in *Proc. 20th Int. Symp. Symbolic Numeric Algorithms Sci. Comput. (SYNASC)*, Sep. 2018, pp. 209–214.
[38] F. Sultana, A. Sufian, and P. Dutta, "A review of object detection models based on convolutional neural network," in *Intelligent Computing: Image Processing Based Applications*. Singapore: Springer, 2020, pp. 1–16.
[39] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digit. Signal Process.*, vol. 126, Jun. 2022, Art. no. 103514.
[40] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.
[41] J. Lee, J. Wang, D. Crandall, S. Šabanovic, and G. Fox, "Real-time, cloud-based object detection for unmanned aerial vehicles," in *Proc. 1st IEEE Int. Conf. Robotic Comput. (IRC)*, Apr. 2017, pp. 36–43.
[42] A. Rastogi, R. Arora, and S. Sharma, "Leaf disease detection and grading using computer vision technology & fuzzy logic," in *Proc. 2nd Int. Conf. Signal Process. Integr. Netw. (SPIN)*, Feb. 2015, pp. 500–505.
[43] Y. A. Nanehkaran, D. Zhang, J. Chen, Y. Tian, and N. Al-Nabhan, "Recognition of plant leaf diseases based on computer vision," *J. Ambient Intell. Humanized Comput.*, vol. 15, pp. 1–18, Sep. 2020.
[44] S. V. Militante, B. D. Gerardo, and N. V. Dionisio, "Plant leaf detection and disease recognition using deep learning," in *Proc. IEEE Eurasia Conf. IoT, Commun. Eng. (ECICE)*, Oct. 2019, pp. 579–582.
[45] M. Günder, F. R. I. Yamati, J. Kierdorf, R. Roscher, A.-K. Mahlein, and C. Bauckhage, "Agricultural plant cataloging and establishment of a data framework from UAV-based crop images by computer vision," *GigaScience*, vol. 11, Jun. 2022.
[46] D. P. Hughes and M. Salathe, "An open access repository of images on plant health to enable the development of mobile disease diagnostics," 2015, *arXiv:1511.08060*.
[47] G. Fenu and F. M. Malloci, "DiaMOS plant: A dataset for diagnosis and monitoring plant disease," *Agronomy*, vol. 11, no. 11, p. 2107, Oct. 2021.
[48] R. A. Krohling, J. Esgario, and J. A. Ventura, "BRACOL—A Brazilian arabica coffee leaf images dataset to identification and quantification of coffee diseases and pests," *Mendeley Data*, vol. 1, Nov. 2019.
[49] S. Fiel and R. Sablatnig, "Leaf classification using local features," in *Proc. 34th Annu. Workshop Austrian Assoc. Pattern Recognit. (AAPR), Comput. Vis. Global Soc., WG Visual Comput. Austrian Comput. Soc.*, P. Blauensteiner, M. Lettner, and J. Stöttinger, Eds., Österreichische Computer Gesellschaft, 2010, pp. 123–130. [Online]. Available: http://hdl.handle.net/20.500.12708/53459
[50] A. Kolker, S. Oshchepkova, Z. Pershina, L. Dimitrov, V. Ivanov, A. Rashid, and M. Bdiwi, "The ray tracing based tool for generation artificial images and neural network training," in *Proc. 12th Int. Joint Conf. Knowl. Discovery, Knowl. Eng. Knowl. Manage.*, 2020, pp. 257–264.
[51] T. A. Le, A. G. Baydin, R. Zinkov, and F. Wood, "Using synthetic data to train neural networks is model-based reasoning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, May 2017, pp. 3514–3521.
[52] S. C. Council. *Ash Dieback Toolkit Ash Tree Assessment the Issue—The Tree Council*. Accessed: Jun. 7, 2024. [Online]. Available: https://treecouncil.org.uk/wp-content/uploads/2019/12/Suffolk-Canopy-Description.pdf
[53] S. L. Bangare, A. Dubal, P. S. Bangare, and S. Patil, "Reviewing Otsu's method for image thresholding," *Int. J. Appl. Eng. Res.*, vol. 10, no. 9, pp. 21777–21783, 2015.
[54] T. Strutz, "The distance transform and its computation," 2021, *arXiv:2106.03503*.
[55] L. G. Fahad, S. F. Tahir, U. Rasheed, H. Saqib, M. Hassan, and H. Alquhayz, "Fruits and vegetables freshness categorization using deep learning," *Comput., Mater. Continua*, vol. 71, no. 3, pp. 5083–5098, 2022.
[56] J. Doherty, B. Gardiner, E. Kerr, N. Siddique, and S. S. Manvi, "Comparative study of activation functions and their impact on the YOLOv5 object detection model," in *Proc. Int. Conf. Pattern Recognit. Artif. Intell.*, Jan. 2022, pp. 40–52.
[57] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions," 2017, *arXiv:1710.05941*.
[58] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[59] J. Redmon. (2013). *Darknet: Open Source Neural Networks in C*. [Online]. Available: http://pjreddie.com/darknet/
[60] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*.
[61] J. Nelson. (2024). *What is YOLO? The Ultimate Guide*. [Online]. Available: https://blog.roboflow.com/guide-to-yolo-models/
[62] WongKinYiu. (2024). *YOLOv9—Github Repository*. Accessed: Dec. 23, 2024. [Online]. Available: https://github.com/WongKinYiu/'yolov9
[63] R. Thapa, K. Zhang, N. Snavely, S. Belongie, and A. Khan, "The plant pathology challenge 2020 data set to classify foliar disease of apples," *Appl. Plant Sci.*, vol. 8, no. 9, Sep. 2020, Art. no. e11390.
[64] H. T. Rauf, B. A. Saleem, M. I. U. Lali, M. A. Khan, M. Sharif, and S. A. C. Bukhari, "A citrus fruits and leaves dataset for detection and classification of citrus diseases through machine learning," *Data Brief*, vol. 26, Oct. 2019, Art. no. 104340.
[65] S. S. Chouhan, A. Kaul, and U. P. Singh, "A database of leaf images: Practice towards plant conservation with plant pathology," *Mendeley Data*, vol. 4, Apr. 2020.
[66] D. Singh, N. Jain, P. Jain, P. Kayal, S. Kumawat, and N. Batra, "PlantDoc: A dataset for visual plant disease detection," in *Proc. 7th ACM IKDD CoDS 25th COMAD*, Jan. 2020, pp. 249–253.
[67] M. A. Shoaib, K. W. Lai, J. H. Chuah, Y. C. Hum, R. Ali, S. Dhanalakshmi, H. Wang, and X. Wu, "Comparative studies of deep learning segmentation models for left ventricle segmentation," *Frontiers Public Health*, vol. 10, Aug. 2022, Art. no. 981019.

ELIZABETH BATES received the M.Sc. degree in artificial intelligence from Imperial College London. She is an AI Researcher. She conducted her final year project with the co-authors in the Aerial Robotics Laboratory. She is currently a Researcher with the Alan Turing Institute, focusing on Reinforcement Learning and its application in the area of autonomous network defence.

MARIJA POPOVIĆ received the master's degree in engineering from the Integrated Mechanical and Electrical Engineering, University of Bath, in 2015, and the Ph.D. degree from the Autonomous Systems Laboratory, ETH Zurich, in 2019. She is currently an Assistant Professor with MAVLab, TU Delft. Before starting in TU Delft, she was a Junior Research Group Leader with the University of Bonn and Cluster of Excellence "PhenoRob," from 2021 to 2024, and a Postdoctoral Research Associate with the Smart Robotics Laboratory, Imperial College London, from 2020 to 2021. Her research centers around developing algorithms that enable robot autonomy, including adaptive informative path planning, robot learning, and computer vision.

CONOR MARSH is currently a Technical Specialist in synthetic environments and real-time simulations for mixed-reality platforms.

RONALD CLARK received the M.Sc. degree in information engineering from the University of the Witwatersrand and the D.Phil. degree from the University of Oxford. From 2017 to 2020, he was a member of the Dyson Robotics Laboratory. He then held an Imperial College Research Fellowship, before joining the University of Oxford, in 2022, as an Associate Professor. His research has received various accolades, including the Best Paper Honourable Mention at CVPR, in 2018.

MIRKO KOVAC (Member, IEEE) is currently the Director of the Aerial Robotics Laboratory, Imperial College London, and the Head of the Materials and Technology Centre of Robotics, Empa Materials Science Institute in Switzerland. His research focuses on the development of novel, biologically inspired flying robots for distributed sensing in air and water and on autonomous robotic construction for digital infrastructure systems. His particular specialization is in robot design, fluid-structure interaction, and multi-modal robot mobility. He is internationally known as an Emerging Leader in bio-inspired aerial robotics. He is the winner of multiple awards and the author of more than 50 articles on mobile robotics that have been published in major journals, including *Science*, IEEE TRANSACTIONS, and *Science Robotics*. He regularly advises industry, investment funds, and government on robotic research strategy and is the holder of the prestigious Royal Society Wolfson Fellowship.

BASARAN BAHADIR KOCER is currently a Lecturer with the University of Bristol. His research explores the integration of design, perception, and control in aerial robots, investigating their impact on the operation and sustainable solutions for the future. His research focuses on aerial physical interaction for applications, including environmental sensing and aerial manufacturing, to advance the eco-friendly and efficient use of aerial robots. Inspired by nature, he studies flexible and efficient learning schemes to develop designs, algorithms, and approaches that can adapt to diverse aerial robots and tasks, an experience that informed his work on the Aerial additive Manufacturing work conducted at Imperial College London. He believes the emphasis on continuous learning enables us to tackle complex and dynamic challenges in aerial systems and the environment.