Tree health assessment from UAV images: Improving object detection and classification using hard negative mining and semi-supervised autoencoder

Hela Jemaa[1,2], Wassim Bouachir[1], Brigitte Leblon[3], Armand LaRocque[4], Ata Haddadi[5] and Nizar Bouguila[2]
[1]*Data Science Laboratory, University of Québec (TÉLUQ), Montréal, Québec*
[2]*Concordia University, Montréal, Québec*
[3]*Lakehead University, Thunder Bay, Ontario*
[4]*University of New Brunswick, Fredericton, New Brunswick*
[5]*A&L Canada Laboratories, London, Ontario*

***Abstract*—Orchard tree inventory has been an essential step to obtain up-to-date information for effective tree treatments and crop insurance purposes. Inventorying trees is often performed manually through fieldwork surveys, which are generally time-consuming, costly, and subject to errors. Motivated by the latest advances in UAV imagery and deep learning, we propose a new framework for individual tree detection and health assessment. We adopt a divide-and-conquer approach to address the problem of orchard trees' health assessment in two stages. First, we build a tree detection model based on a hard negative mining strategy to improve object detection. In the second stage, we address the health classification problem using a new convolutional autoencoder architecture mainly designed to extract relevant features. The performed experiments demonstrate the robustness of the proposed framework for orchard tree health assessment from UAV images. In particular, our framework achieves an F1-score of 86.24% for tree detection and an overall accuracy of 98.06% for tree health assessment. Moreover, our work could be generalized for a wide range of UAV applications involving a detection/classification process.**

***Keywords*-Hard Negative Mining (HNM); Autoencoder; semi-supervised learning; UAV; YOLO; DeepForest.**

I. Introduction

The agriculture industry has always been concerned about tree diseases due to their significant and negative impact on crop quality and production. Impacts of crop stress, such as diseases and pests, range from minor side effects to severe losses of entire yields, resulting in significant costs for agricultural businesses. Traditional tree inventorying methods mainly rely on human expertise and are often performed manually, which is labor-intensive, time-consuming, costly, and subject to errors. Recent advances in remote sensing have produced new instruments, providing new alternatives to conventional approaches, such as satellites, airplanes, and unmanned aerial vehicles (UAVs). On the other hand, deep neural networks (DNNs) [1] have considerably advanced the state-of-the-art in a variety of visual recognition tasks. Motivated by the latest advances in deep learning-based computer vision systems, we propose a new framework to automatically detect trees and assess their health. The proposed framework adopts a divide-and-conquer approach through two main stages. The first stage addresses the tree localization problem using visual object detection, while the second deals with the tree health assessment through image patch classification.

In the field of object **detection**, extensive studies [2]–[7] have been devoted to aerial imagery applications. Most of the proposed methods have adopted approaches based on fine-tuning networks, pre-trained on large-scale image datasets (e.g., ImageNet [8] and MSCOCO [9]) for detection in the UAV domain. While such fine-tuning-based approaches achieved interesting results, we argue that UAV images exhibit particular challenges compared to conventional object detection tasks. First, UAV images often have a large field of view, with more complex background regions, which may substantially disturb object detection. Second, objects of interest are often not uniformly distributed with respect to background regions, which may cause an imbalance between positive and negative examples. The data imbalance problem can also be observed between easy and hard negative examples, as with UAV images, a large part of the background shows regular patterns and can be analyzed easily for detection. We believe that existing deep learning algorithms are not mainly designed for these situations [10], as they mostly assign the same weight to all the examples, so that during training, easy examples may dominate the total loss, reducing training efficiency.

In order to improve the network's robustness against complex backgrounds, we propose a dedicated detection procedure based on a hard negative mining strategy. This allows the model to focus on hard examples during the training phase, which helps reduce detection errors caused by complex backgrounds.

Following the detection stage, we formulate the health status assessment of each detected tree as a **classification** task of tree image patches. With visual classification, a key challenge is to find the relevant feature representation that best describes the data in the feature space. Traditional autoencoders [11]–[13] have been studied and used for feature extraction tasks. However, because standard autoencoders generally do not use label information, the extracted repre-

sentation may be limited in handling discriminative tasks. To address the feature relevance problem, we instead propose a semi-supervised discriminant convolutional autoencoder, where the encoder part is trained on both labeled and unlabeled data to learn a compact and relevant representation of input data. This approach allows the classification network to exploit both labeled and unlabeled data to learn a more robust feature representation for classification.

By using the hard negative mining approach and the semi-supervised autoencoder discussed above, we propose an effective framework to automatically detect orchard trees and perform health assessment. The main contributions of this paper are summarized as follows:

- We propose a novel framework for automatic tree detection and health assessment from UAV images, according to a divide-and-conquer approach. The proposed framework could be generalized for a wide range of other UAV applications involving a detection/classification process.
- We adopt a hard negative mining approach to address the problem of negative hard-easy examples imbalance.
- We present a semi-supervised convolutional autoencoder to address the tree health assessment problem, taking advantage of both labeled and unlabeled data, to ensure the relevance of feature representation for the health classification task.

II. Related works

A. Tree health assessment

In this section, we present related works devoted to tree health assessment using UAV images and computer vision. The study in [14] proposes a framework for the detection and quantification of Eucalyptus Longhorned Borers (ELB) damages in eucalyptus stands. Treetops were calculated using the local maxima filter of a sliding window algorithm. Afterward, large-scale mean-shift segmentation was performed to extract the crowns and classify them using random forest. The work of [15] presents a two-step solution. First, the authors use a candidate selection technique to find the potential regions corresponding to trees. In the second phase, a convolutional neural network (CNN) architecture is used to predict the fir tree damage stage in each candidate region. In [16], the authors propose an approach for species classification and assessment of the vital status of forest stands, by using automated individual tree crowns delineation. They use preprocessing techniques for tree segmentation. They then employ an ensemble algorithm, known as error-correcting output codes (ECOC) to carry out pixel-by-pixel classification of images by spectral features. Vegetation indices were explored in [17] to train and validate a support vector machine (SVM) model to classify each tree pixel into one of the two categories: healthy and stressed. Based on UAV-based hyperspectral images, [18] introduces a spectral-spatial classification framework combining an SVM with an edge-preserving filter (EPF) to automatically extract tree crowns damaged by Dendrolimus tabulaeformis.

B. Hard negative mining

Object detectors often face the issue of data imbalance, where training datasets include a large number of negative examples. Generally, most of the negative data samples are easily identified by the detector (easy negative examples) while only a few are difficult (hard negative examples).

To mitigate this issue, hard negative mining (HNM) can be adopted. Various HNM approaches [19]–[21] involve iteratively bootstrapping a small set of negative examples, by selecting those that trigger a false positive alarm in the detector. For example, [22] presented a training process of a state-of-the-art face detector by exploiting the idea of hard negative mining and iteratively updating the Faster R-CNN-based face detector with hard negatives harvested from a large set of background examples. Their method outperforms state-of-the-art detectors on the Face Detection Data Set and Benchmark (FDDB). Similarly, an improved version of faster R-CNN is proposed in [23], by using hard negative sample mining for object detection. Likewise, [24] used the bootstrapping of hard negatives to improve the performance of CNN-based detectors. The authors pre-trained Faster R-CNN to mine hard negatives, before retraining the model. The work of [25] presented a cascaded Boosted Forest, which performs effective hard negative mining and sample reweighting, to classify the region proposals generated by RPN. The A-Fast-RCNN method, described in [26], adopts a different approach for generating hard negative samples, by using occlusion and spatial deformations through an adversarial process. Another approach to apply HNM using Single Shot multi-box Detector (SSD) is proposed in [27], where the authors use medium priors, anchor boxes with 20% to 50% overlap with ground truth boxes, to enhance object detector performance. The proposed framework updates the loss function so that it considers the anchor boxes with partial and marginal overlap.

In our method, we propose a HNM approach for the tree detection stage, where the mined hard negative samples are used to introduce a new class. Then, we retrain the object detector using the true positive and false positive examples to enhance the discrimination power of the model.

C. Semi-supervised autoencoder

Autoencoders and their extensions [12], [13], [28] have been important in various fields such as computer vision, and natural language processing (NLP). A standard Autoencoder is an unsupervised model made up of an encoder network that maps the input data to a latent space representation, and a decoder network mapping the latent space representation back to the original input data. The goal is to learn a compact representation capturing the most important features, by

minimizing the reconstruction loss. However, representations extracted by conventional autoencoders may not be useful for discriminative tasks [29], as label information is not used by unsupervised autoencoders.

There are some novel semi-supervised autoencoders that are able to learn from both labeled and unlabeled data [30]–[32]. The basic idea is to use the autoencoder to learn an efficient representation of the input data, and then use the learned representation for training on a specific task using labeled data. Semi-supervised autoencoders have been applied in various domains, including computer vision and NLP. For example, the work in [30] introduces a semi-supervised version of the Variational Autoencoder (VAE) model for image classification. The authors use a VAE to learn a latent representation of the input images, and then use a classifier to predict the label of the image based on the latent representation. The model is trained on both labeled and unlabeled data, with the objective of maximizing the log-likelihood of the labeled data and the marginal likelihood of the unlabeled data. The authors demonstrate that their model outperforms existing semi-supervised methods on the MNIST dataset. Similarly, a dual-objective framework is presented in [32] for feature extraction in fault diagnosis. The work of [31] proposes a semi-supervised variant of the Generative Adversarial Network (GAN) model for image classification. A GAN is used to learn a generator network and a discriminator network. The network is used to generate images and the discriminator network is used to predict the label of the image. The model is trained on both labeled and unlabeled data, with the objective of minimizing the cross-entropy loss of the labeled data and the Wasserstein distance of the unlabeled data.

For our tree health classification stage, we adopt a semi-supervised autoencoder using both labeled and unlabeled data to provide relevant features that best determine the tree damage status. Our proposed autoencoder-based solution uses a dual-objective framework, where the autoencoder and the classifier are jointly trained to optimize both reconstruction and classification objectives.

III. Proposed Method

A. Motivation and overview

The objective of this work is to develop an automated framework for tree health assessment using UAV RGB images. The proposed framework, shown in Figure 1, adopts a divide-and-conquer strategy to solve two sub-problems: tree detection and tree health classification.

The task of tree detection is challenging due to the presence of complex backgrounds that may distract object detectors. As depicted in Figure 2, the color, shape, and texture of some objects belonging to the background (yellow rectangles) are visually similar to the target tree objects (blue rectangles), leading to false positives. To overcome this challenge, we adopted a hard negative mining approach

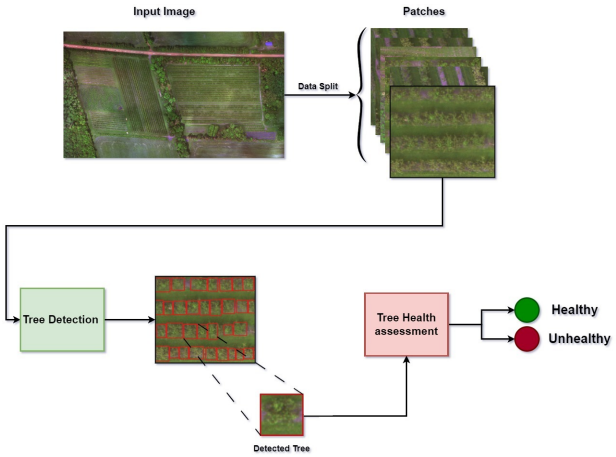

Figure 1. Overview of the proposed framework

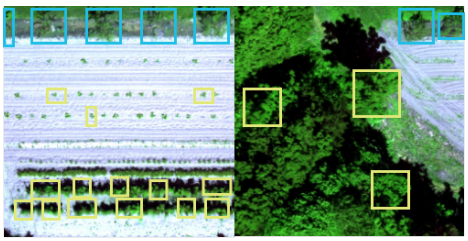

Figure 2. Detection results of a baseline model without the use of hard negative mining approach. Yellow rectangles are false detections (FP) and blue rectangles are correct detections (TP). When using a hard negative mining strategy, most of the false detections are not detected

to improve the detector's ability to discriminate between trees and background regions. By iteratively introducing false positives as hard negative examples during training, the detector can learn to better differentiate between the objects of interest and background regions.

For tree health assessment, we aim to extract the most relevant low-dimensional features to improve the performance of tree damage classification. To deal with this issue, we proposed a semi-supervised autoencoder that goes through unsupervised training and supervised training to take advantage of both labeled and unlabelled data. This semi-supervised training is performed to optimize both the reconstruction loss and the binary cross-entropy loss, which helps extract the most relevant features discriminating between the two classes.

B. Individual tree detection

The proposed method for tree detection, shown in Figure 3, involves training a baseline object detector with our manually annotated tree dataset. This baseline model is then evaluated to identify hard negative examples, which generally correspond to the areas causing false positives. The identified examples are used to introduce a second class. Once hard negative samples are harvested, we include them

in the training set and perform fine-tuning of the baseline tree detector. The motivation behind using false positive (FP) detections as a new class is that FP is generally a source of noise in the training data, which may cause inaccurate detection. By including these samples and retraining the object detector, the model learns to distinguish between true positives and false positives. We use the focal loss [33] as an objective function during fine-tuning of the object detector to address the issue of class imbalance between the target class (tree) and the hard negative class.

The process of mining hard negatives and fine-tuning is performed iteratively, by continually refining the training set, which gradually improves the tree detection accuracy. The detection method steps are outlined in Algorithm 1.

Algorithm 1 Hard negative mining algorithm for object detection

Require: Training dataset with manual annotation
Ensure: Improved model for object detection
1: Train a baseline detector using the annotated dataset
2: Perform qualitative and quantitative evaluation
3: **while** Performance is unsatisfactory **do**
4: Identify false positive detections
5: Define new negative class using false positives
6: Add false positives to the training dataset
7: Fine-tune model using focal loss and updated training dataset
8: Evaluate
9: **end while**

C. Tree health assessment

To deal with tree health classification, we introduce a semi-supervised autoencoder. The proposed network architecture, shown in Figure 4, is similar to that of a traditional autoencoder, except for the training process and the loss function. The training process could be separated into unsupervised training and supervised training. Thanks to supervised training, a semi-supervised autoencoder makes full use of label information to provide a more appropriate representation than traditional autoencoders. The loss function (Eq. 1) is a combination of reconstruction loss and binary cross-entropy loss.

$$L_{total} = L_{recons}(x, \hat{x}) + L_{binary}(y, \hat{y}), \tag{1}$$

where $L_{recons}(x, \hat{x}) = ||x - \hat{x}||_2^2$ is the reconstruction loss, x is the input tree image, and $\hat{x}$ is the reconstructed image from the autoencoder. $L_{binary}(y, \hat{y}) = -y \log(\hat{y}) - (1 - y) \log(1 - \hat{y})$ is the binary cross entropy, where y is the true binary label, and $\hat{y}$ is the predicted probability of the positive class.

The training process is made up of two types of training:

1) Unsupervised training: During this step, the architecture and the loss function of our semi-supervised autoencoder are the same as that of the traditional autoencoder. Its architecture consists of an encoder that maps the input data into a lower-dimensional latent space representation and a decoder that maps the encoding back into the original input space.

2) Supervised training: In the supervised training, the architecture of the encoder is the same as the traditional autoencoder with an altered head designed for binary classification. The optimal values of parameters obtained through the unsupervised training process are set as the initial values of the encoder during the supervised training process.

IV. Experiments

A. Dataset

Images were collected over two apple orchards in Souris, Prince Edward Island, Canada (Lat. 46.44633N, Long. 62.08151W). Our dataset consists of UAV images of four orchards containing both healthy and damaged trees. To prepare the data for object detection models, the orthomosaic is split into small patches of 515x512 pixels using a regular grid. These patches are then divided into three subsets: training, validation, and testing using 3-fold cross-validation to ensure an unbiased evaluation of object detection models.

Trees have been annotated by indicating bounding box locations, as well as their health status (healthy or damaged) based on fieldwork inventories. The total number of trees in the dataset is approximately 2,828, out of which 2,240 are healthy, and 588 are unhealthy. The tree images were divided into training, validation, and testing sets using stratified 10-fold cross-validation, ensuring that the ratio of healthy and damaged trees remains consistent across subsets.

B. Experimental setup

To evaluate the performance of our framework for tree detection, we use the following metrics.

- Precision P_{d} (Eq. 2) is the percentage of correct detections among all the detected trees.

$$P_d = \frac{\mathrm{TP}_d}{\mathrm{TP}_d + \mathrm{FP}_d} \tag{2}$$

- Recall R_{d} (Eq. 3) is the percentage of correctly detected trees over the total number of trees in the ground truth.

$$R_d = \frac{\mathrm{TP}_d}{\mathrm{TP}_d + \mathrm{FN}_d} \tag{3}$$

- $F1\text{-score}_d$ (Eq. 4) is the harmonic average of precision and recall.

$$F1\text{-score}_d = 2 * \frac{\mathrm{P}_d * \mathrm{R}_d}{\mathrm{P}_d + \mathrm{R}_d} \tag{4}$$

In equations 2, 3, and 4, the subscript *d* denotes detection, TP_d is the number of true positives (i.e. correctly detected trees), FP_d is the number of false positives (i.e. regions

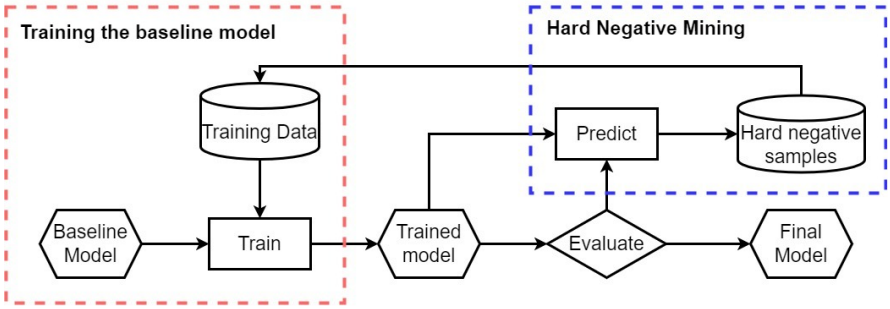

Figure 3. Tree Detection using hard negative mining approach

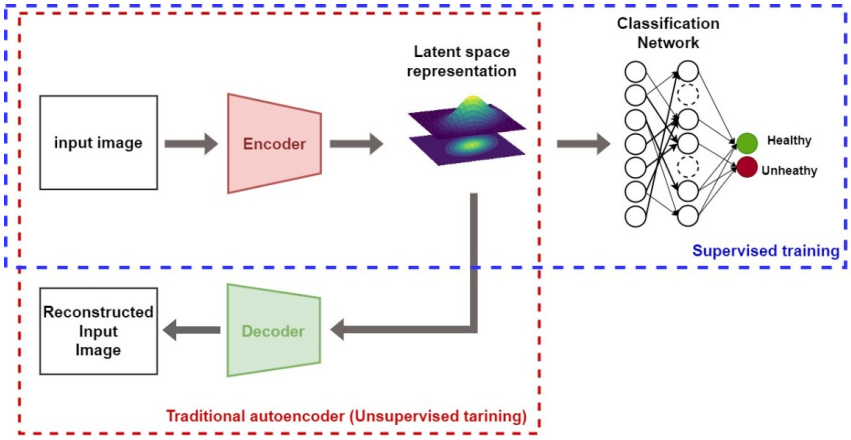

Figure 4. Semi-supervised autoencoder for tree health assessment. The red box illustrates the unsupervised learning using a traditional autoencoder, and the blue box represents the supervised training for tree health classification

incorrectly detected as trees), and FN_d denotes the number of false negatives (i.e. number of missed trees). On a test image, a detection is considered as correct if the Intersection Over Union (IOU) between the detected tree and the tree region in the ground truth is greater than 50%.

To evaluate the performance of our framework for tree health classification, we use the following metrics.

- Precision P_c (Eq. 5) is defined as the ratio of correct classifications for a given class to the total number of classifications made for that class.

$$P_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FP}_c} \tag{5}$$

- Recall R_c (Eq. 6) is defined as the ratio of correct classifications for a given class to the total number of instances that actually belong to that class.

$$R_c = \frac{\text{TP}_c}{\text{TP}_c + \text{FN}_c} \tag{6}$$

- $F1\text{-score}_c$ is the harmonic average of P_c and R_c of a given class.
- Accuracy (Eq. 7) is defined as the ratio of the correct classifications to the total number of tree instances classified.

$$Accuracy = \frac{\text{Number of correct classifications}}{\text{Total number of trees classified}} \tag{7}$$

In equations 5, 6, and 7, the subscript c denotes classification, TP_c is the number of correctly classified instances of a given class, FP_c is the number of instances incorrectly classified as belonging to a given class, and FN_c is the number of instances incorrectly classified as not belonging to a given class.

C. Detection results

The goal of this experiment is to evaluate the effectiveness of the proposed HNM approach for tree detection. We use the DeepForest object detector [34] as a baseline model. The choice of DeepForest is motivated by its domain-specific knowledge, as it is trained on a large dataset that includes images of different tree species, ages, and environmental conditions. We apply our HNM strategy on DeepForest using its prebuilt model trained on The National Ecological Observatory Network (NEON [35]) crowns dataset. Table I shows the detailed and overall cross-validation results for the proposed approach. From the table, we can see that the proposed detection method achieves an overall F1-score of 86.24%. We can also notice that the method is

Table I
DETAILED AND OVERALL CROSS-VALIDATION RESULTS OF THE TREE DETECTION STAGE IN TERMS OF PRECISION, RECALL, AND F1-SCORE.

Folds	$P_{\text{d}}(\%)$	$R_{\text{d}}(\%)$	$F1\text{-score}_d(\%)$
Fold1	82.25	87.24	84.67
Fold2	87.57	88.06	87.82
Fold3	87.87	84.73	86.27
Average	85.85	86.67	86.24

stable across folds. This demonstrates the robustness of our model, which is able to perform well and consistently on different partitions. Further, in the ablation study section, we will demonstrate the importance of our HNM strategy in improving detection results.

D. Health assessment and comparison with state-of-the-art methods

We addressed the problem of tree health classification using a discriminative semi-supervised autoencoder and compared its performance with other widely used classifiers including ResNet, VGG, and DenseNet. We trained ResNet, VGG, and DenseNet on the same training set, using standard supervised learning. These classifiers were pretrained on the ImageNet dataset and fine-tuned on our tree dataset.

We evaluated the performance of all classifiers on the test set, using the defined classification metrics. Our results in Table II showed that the proposed semi-supervised autoencoder approach outperformed the other classifiers in all metrics, achieving an accuracy of 98.06%. The ResNet, VGG, and DenseNet classifiers achieved lower accuracy, precision, recall, and F1-score, with the best-performing model being ResNet with an accuracy of 90.85%. From the table, we can see also that the compared classifiers achieved unsatisfactory results for the unhealthy class compared to the healthy class.

The outperformance of the proposed semi-supervised autoencoder over VGG, ResNet, and DenseNet for tree health classification can be attributed to several factors. One of the key advantages of the semi-supervised approach is its ability to leverage both labeled and unlabeled data to learn robust and discriminative features. This is achieved by enforcing a reconstruction loss preserving the model input structure, while also learning informative features for the classification task. However, the compared models rely solely on labeled data and use more complex and computationally intensive architectures that may struggle to generalize to rare classes, leading to a low performance for the class unhealthy. Overall, our results demonstrate the effectiveness of the proposed discriminant semi-supervised autoencoder approach for tree health classification.

Table II
OVERALL CROSS-VALIDATION RESULTS OF OUR MODEL COMPARED TO OTHER DEEP LEARNING-BASED APPROACHES FOR HEALTH ASSESSMENT IN TERMS OF PRECISION, RECALL, F1-SCORE, AND ACCURACY. VALUES IN BOLD FONT CORRESPOND TO THE BEST-ACHIEVED RESULTS

Health Status	$P_{\text{c}}(\%)$	$R_{\text{c}}(\%)$	$F1\text{-score}_c(\%)$	Accuracy(%)
Resnet-101				
Healthy	96.52	93.68	95.07	90.85
Unhealthy	30.65	43.75	35.46	
DenseNet-121				
Healthy	96.78	92.56	94.61	90.07
Unhealthy	28.46	48.75	35.74	
VGG-16				
Healthy	96.02	93.05	94.41	89.75
Unhealthy	25.26	35	25.58	
Discriminant semi-supervised autoencoder				
Healthy	**98.05**	**99**	**98.25**	**98.06**
Unhealthy	**95.78**	**96.45**	**96.03**	

Table III
ABLATION STUDY RESULT OF THE DETECTION STEP. VALUES IN BOLD FONT CORRESPOND TO THE BEST RESULTS

Baseline Model	$P_{\text{d}}(\%)$	$R_{\text{d}}(\%)$	$F1\text{-score}_d(\%)$
Our detection model without HNM			
DeepForest	84.82	86.18	85.46
YOLO	79.40	88.05	82.64
Our detection model with HNM			
DeepForest	**85.85**	86.67	**86.24**
YOLO	82.01	**88.99**	84.81

V. ABLATION STUDY

- **Importance of mining hard examples for tree detection**: In order to evaluate the performance of the hard negative sampling strategy for tree detection, we reported the results of training a baseline object detector without hard negative samples by comparison to our HNM-based approach. Table III reports the overall 10-fold cross-validation results using two baseline models: DeepForest and YOLO-v5. The reported results show that both YOLO and DeepForest benefited from hard negative mining, with significant improvements in F1-score. However, DeepForest consistently outperformed YOLO in all experiments, achieving higher results. YOLO fine-tuned with the mined hard negatives achieved an F1-score of 84.81%, outperforming the YOLO baseline by 2.17%, which means that the detector learns to eliminate a number of false detections. Using the DeepForest model, the inclusion of hard negatives in training improves the performance compared to the baseline, with an improvement of 0.78% based on the F1-score.
- **Importance of unsupervised feature learning using the auto-encoder module for tree health assessment**: We also conducted an ablation study to investigate the contribution of unsupervised training using the autoencoder component of the proposed approach. In

Table IV
ABLATION STUDY OF THE CLASSIFICATION STEP. VALUES IN BOLD FONT CORRESPOND TO THE BEST RESULTS

Health Status	$P_c(\%)$	$R_c(\%)$	$F1\text{-score}_c(\%)$	Accuracy(%)
Our classification model without the autoencoder				
Healthy	97.57	96.56	97.06	95.35
Unhealthy	87.49	90.69	88.97	
Our classification model with the autoencoder				
Healthy	**98.05**	**99**	**98.25**	**98.06**
Unhealthy	**95.78**	**96.45**	**96.03**	

Table IV, we report the results of our classifier without and with the use of the autoencoder. We can see that the unsupervised pretraining of the autoencoder is a key component of the proposed approach, contributing to its superior performance. Training the autoencoder using unlabeled data, followed by supervised training on labeled data for the specific task of tree classification resulted in a 2.71% improvement in overall accuracy. The improvement can also be seen through the other metrics, precision, recall, and F1-score. This demonstrates the effectiveness of our approach to extract relevant features

Overall, the results of the ablation study highlight both the significance of adopting the hard negative mining (HNM) approach for object detection in complex backgrounds, as well as the effectiveness of using a discriminant semi-supervised autoencoder to extract relevant features for tree health classification.

VI. CONCLUSION

In this paper, an effective framework is proposed to address the problem of tree health assessment from UAV images. The first stage addresses the tree detection problem using a hard negative mining approach to improve tree detection performance. The second stage deals with tree health classification, where we propose a discriminative semi-supervised autoencoder as a binary classifier to identify damaged trees. Through our experiments, it has been shown that significant detection performance gains can be achieved by learning a baseline detection model with hard negative mining, and that the semi-supervised autoencoder allows the extraction of relevant features from damaged trees, which significantly improves classification performance.

Our future work aims to investigate the use of other bands such as red edge and near-infrared. These bands have been shown to provide valuable information about vegetation structure, to distinguish between trees and non-tree objects. Additionally, the use of vegetation indices such as the Normalized Difference Vegetation Index (NDVI) can provide insights into tree health and stress levels. We thus consider incorporating these modalities into our tree detection and health assessment approach for further performance improvement.

REFERENCES

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[2] Y. Long, Y. Gong, Z. Xiao, and Q. Liu, "Accurate object localization in remote sensing images based on convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 5, pp. 2486–2498, 2017.

[3] H. Jemaa, W. Bouachir, B. Leblon, and N. Bouguila, "Computer vision system for detecting orchard trees from uav images," *ISPRS-International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 43, pp. 661–668.

[4] G. Wang, X. Wang, B. Fan, and C. Pan, "Feature extraction by rotation-invariant matrix representation for object detection in aerial image," *IEEE Geoscience and Remote Sensing Letters*, vol. 14, no. 6, pp. 851–855, 2017.

[5] T. Moranduzzo and F. Melgani, "Detecting cars in uav images with a catalog-based approach," *IEEE Transactions on Geoscience and remote sensing*, vol. 52, no. 10, pp. 6356–6367, 2014.

[6] G. Cheng, J. Han, P. Zhou, and D. Xu, "Learning rotation-invariant and fisher discriminative convolutional neural networks for object detection," *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 265–278, 2018.

[7] Y. Lin, H. He, Z. Yin, and F. Chen, "Rotation-invariant object detection in remote sensing images based on radial-gradient angle," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 4, pp. 746–750, 2014.

[8] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[9] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*. Springer, 2014, pp. 740–755.

[10] G.-S. Xia, X. Bai, J. Ding, Z. Zhu, S. Belongie, J. Luo, M. Datcu, M. Pelillo, and L. Zhang, "Dota: A large-scale dataset for object detection in aerial images," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 3974–3983.

[11] Q. Meng, D. Catchpoole, D. Skillicom, and P. J. Kennedy, "Relational autoencoder for feature extraction," in *2017 International joint conference on neural networks (IJCNN)*. IEEE, 2017, pp. 364–371.

[12] K. Chen, J. Hu, and J. He, "A framework for automatically extracting overvoltage features based on sparse autoencoder," *IEEE Transactions on Smart Grid*, vol. 9, no. 2, pp. 594–604, 2016.

[13] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol, "Extracting and composing robust features with denoising autoencoders," in *Proceedings of the 25th international conference on Machine learning*, 2008, pp. 1096–1103.

[14] A. Duarte, L. Acevedo-Muñoz, C. I. Gonçalves, L. Mota, A. Sarmento, M. Silva, S. Fabres, N. Borralho, and C. Valente, "Detection of longhorned borer attack and assessment in eucalyptus plantations using uav imagery," *Remote Sensing*, vol. 12, no. 19, p. 3153, 2020.

[15] A. Safonova, S. Tabik, D. Alcaraz-Segura, A. Rubtsov, Y. Maglinets, and F. Herrera, "Detection of fir trees (abies sibirica) damaged by the bark beetle in unmanned aerial vehicle images with deep learning," *Remote sensing*, vol. 11, no. 6, p. 643, 2019.

[16] A. Safonova, Y. Hamad, E. Dmitriev, G. Georgiev, V. Trenkin, M. Georgieva, S. Dimitrov, and M. Iliev, "Individual tree crown delineation for the species classification and assessment of vital status of forest stands from uav images," *Drones*, vol. 5, no. 3, p. 77, 2021.

[17] I. Navrozidis, A. Haugommard, D. Kasampalis, T. Alexandridis, F. Castel, D. Moshou, G. Ovakoglou, X. E. Pantazi, A. A. Tamouridou, A. L. Lagopodi *et al.*, "Assessing olive trees health using vegetation indices and mundi web services for sentinel-2 images." in *HAICTA*, 2020, pp. 130–136.

[18] N. Zhang, Y. Wang, and X. Zhang, "Extraction of tree crowns damaged by dendrolimus tabulaeformis tsai et liu via spectral-spatial classification using uav-based hyperspectral images," *Plant Methods*, vol. 16, pp. 1–19, 2020.

[19] S. Jin, A. RoyChowdhury, H. Jiang, A. Singh, A. Prasad, D. Chakraborty, and E. Learned-Miller, "Unsupervised hard example mining from videos for improved object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 307–324.

[20] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 761–769.

[21] L. Zhang, Y. Wang, and Y. Huo, "Object detection in high-resolution remote sensing images based on a hard-example-mining network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 10, pp. 8768–8780, 2020.

[22] S. Wan, Z. Chen, T. Zhang, B. Zhang, and K.-k. Wong, "Bootstrapping face detection with hard negative examples," *arXiv preprint arXiv:1608.02236*, 2016.

[23] Y. Liu, "An improved faster r-cnn for object detection," in *2018 11th International Symposium on Computational Intelligence and Design (ISCID)*, vol. 2. IEEE, 2018, pp. 119–123.

[24] X. Sun, P. Wu, and S. C. Hoi, "Face detection using deep learning: An improved faster rcnn approach," *Neurocomputing*, vol. 299, pp. 42–50, 2018.

[25] L. Zhang, L. Lin, X. Liang, and K. He, "Is faster r-cnn doing well for pedestrian detection?" in *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*. Springer, 2016, pp. 443–457.

[26] X. Wang, A. Shrivastava, and A. Gupta, "A-fast-rcnn: Hard positive generation via adversary for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[27] N. Ravi and M. El-Sharkawy, "Improved single shot detector with enhanced hard negative mining approach," in *2022 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*. IEEE, 2022, pp. 25–30.

[28] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Proceedings of the 28th international conference on international conference on machine learning*, 2011, pp. 833–840.

[29] J. Snoek, R. P. Adams, and H. Larochelle, "Nonparametric guidance of autoencoder representations using label information," *Journal of Machine Learning Research*, 2012.

[30] D. P. Kingma, S. Mohamed, D. Jimenez Rezende, and M. Welling, "Semi-supervised learning with deep generative models," *Advances in neural information processing systems*, vol. 27, 2014.

[31] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training gans," *Advances in neural information processing systems*, vol. 29, 2016.

[32] X. Luo, X. Li, Z. Wang, and J. Liang, "Discriminant autoencoder for feature extraction in fault diagnosis," *Chemometrics and Intelligent Laboratory Systems*, vol. 192, p. 103814, 2019.

[33] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[34] B. G. Weinstein, S. Marconi, S. A. Bohlman, A. Zare, and E. P. White, "Cross-site learning in deep learning rgb tree crown detection," *Ecological Informatics*, vol. 56, p. 101061, 2020.

[35] B. G. Weinstein, S. Marconi, S. A. Bohlman, A. Zare, A. Singh, S. J. Graves, and E. P. White, "A remote sensing derived data set of 100 million individual tree crowns for the national ecological observatory network," *Elife*, vol. 10, p. e62922, 2021.