Computers and Electronics in Agriculture 162 (2019) 53–69

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

A framework for the management of agricultural resources with automated aerial imagery detection

Karla Saldana Ochoa*,[1], Zifeng Guo[1]

ETH Zürich, Institute of Technology in Architecture, Chair for Computer Aided Architectural Design, Switzerland

ARTICLE INFO

Keywords:
Trees detection
Street segmentation
Agriculture
Machine learning
CNN
UAV

ABSTRACT

The acquisition of data through remote sensing represents a significant advantage in agriculture, as it allows researchers to perform faster and cheaper inspections over large areas. Currently, extensive researches have been done on technical solutions that can benefit simultaneously from both: vast amounts of raw data (big data) extracted from satellite images and Unmanned Aerial Vehicle (UAV) and novel algorithms in Machine Learning for image processing. In this experiment, we provide an approach that fulfills the necessities of rapid food security, assessment, planning, exploitation, and management of agricultural resources by introducing a pipeline for the automatic localization and classification of four types of fruit trees (coconut, banana, mango, and papaya) and the segmentation of roads in the Kingdom of Tonga, using high-resolution aerial imagery (0.04 m).

We used two supervised deep convolutional neural network (CNN): the first, to localize and classify trees (localization) and the second, to mask the streets from the aerial imagery for transportation purposes (semantic segmentation). Additionally, we propose auxiliary methods to determine the density of groupings of each of these trees species, based on the detection results from the localization task and render it in Density Maps that allow comprehending the condition of the agriculture site quickly. Ultimately, we introduce a method to optimize the harvesting of fruits, based on specific sceneries, such as maximum time, path length, and location of warehouses and security points.

1. Introduction

Located in the Pacific Ocean, the Kingdom of Tonga extends over an area of 362,000 km^2. With a population of 107.122 inhabitants in 2016, 58.4% of its population depends on agriculture and forestry as a primary source of income and a key driver for economic growth. Its most prominent agricultural products are bananas, coconuts, coffee beans, vanilla beans, and roots such as cassava, sweet potato, and taro[2] (Halavatau and Halavatau, 2001).

Most of the countries in the Pacific region are exposed to high-risk disasters including cyclones, earthquakes, tsunami, storm surge, volcanic eruptions, landslides, and droughts, e.g., Tonga is affected by more than one tropical cyclone every four years. Theses recurrent disasters cause damage and losses to agriculture, food security and local economy. In the last years, according to the 2015 Report of the Secretary-General on the Implementation of the International Strategy for Disaster Reduction; disasters worldwide cost around USD 1.5 trillion in economic damage. The frequency and severity of natural disasters are increasing, revealing an urgent need to strengthen the resilience of food assessments and security (FAO, 2015).

To understand how local agriculture and food security were affected by a natural disaster, aerial imagery from the site and the succeeding mapping and classification of data are required. The field of Remote Sensing over the past decades has robustly investigated faster methods to collect, produce, classify, and map earth observation data. In recent years, the use of Unmanned Aerial Vehicles (UAV) to collect data has increased rapidly, mainly for their inexpensive hardware and rapidly deploy for the collection of imagery. In parallel, the development of new technics to detect objects in optical remote sensing imagery were actively explored[3] by several scholars. In 1991 an automatic tree detection and delineation from digital imagery was performed by Pinz (1991) who proposed a Vision Expert System using aerial imagery. He

* Corresponding author: Building HIB, Floor E 15, Stefano-Franscini-Platz 1, CH-8093 Zurich, Switzerland.
E-mail address: saldana@arch.ethz.ch (K. Saldana Ochoa).

[1] The two authors contributed equally to this work.

[2] The processing of coconuts into copra and dried coconut was once the only significant industry and only commercial export.

[3] This process determines whether a given aerial or satellite image contains one or more objects belonging to the class of interest and locate the position of each predicted object in the image (Cheng and Han, 2016).

https://doi.org/10.1016/j.compag.2019.03.028
Received 12 October 2018; Received in revised form 18 February 2019; Accepted 27 March 2019

was able to locate the center of trees crown and estimate their radius using local brightness maxima. In 1995 Gougeon (1995), launched a rule-based algorithm, that followed the valleys of shadows between tree crow in a ground sampled distance from digital aerial imagery. Hung et al. (2006) proposed a vision-based shadow algorithm for tree crowns to detect and classify imagery from UAV, using color and texture information to segment regions of interest. Hassaan et al. (2016) presented an algorithm to count trees in urban environments using image processing techniques for vegetation segmentation and tree counting. By applying a k-means clustering algorithm and setting threshold values to green clusters centers, the algorithm was able to segment out the green portion out of any image without any noise.

Today, the development of machine learning approaches provides researchers with a conceptual alternative to solve problems in the mentioned domains without predefining the rules for a specific task. Instead, models can learn the underlying features emerging from a large amount of data. One of the most prominent approaches comes from the field of image processing and computer vision named Convolutional Neural Network (CNN). The algorithm is based on an end-to-end learning process, from raw data to semantic labels, which is an essential advantage in comparison with previous state-of-the-art methods (Nogueira et al., 2017). This model outperforms all the other approaches in tasks like image classification, object recognition and localization, and pixel-wise semantic labeling. The early implementation of CNN by LeCun et al. (1998) achieved 99.2% of accuracy in handwriting digits recognition and led the boost of CNN based image processing in the following 20 years. In recent years, large online image repositories such as ImageNet (Deng et al., 2009), and high-performance computing platforms like GPU acceleration, have contributed significantly to the success of using CNN in a large-scale image and video recognition. Competitions and challenges like the ImageNet Challenge (Russakovsky et al., 2015) and Visual Object Classes Challenge (Everingham et al., 2015) attract many researchers and as a result, state-of-art CNN models such as AlexNet (Krizhevsky et al., 2012) and VGG-Net (Simonyan & Zisserman, 2014) respectively – both available online.

Moreover, researchers can directly use or train these models on their dataset with no need to design its architecture, e.g., YOLO model (Redmon et al., 2016) achieved an excellent performance on recognition and made the real-time object localization possible. In the meantime, Long et al. (2015) with their novel model FCN achieved 20% relative improvement in pixel-wise semantic segmentation in the PASCAL VOC challenge. Also, SegNet proposed by Badrinarayanan et al. (2015) also achieved competitive performance as it is designed to be efficient both in terms of memory and computational time during prediction – It is also significantly smaller in the number of trainable parameters than other competing architectures.

The use of deep learning[4] in Remote Sensing has grown exponentially since it can effectively encode spectral and spatial information based on the data itself. During the last years, considerable efforts have been made to develop various methods for the detection of different types of objects in satellite and aerial images with CNN, such as road, vegetation, tree, water, buildings, cars, etc. – In the Conclusion section we address quantitative measures to support the effectiveness of the proposed approach compared to existing approaches: Chen et al., 2014; Luus et al., 2015; Lu et al. 2017; Kussul et al., 2017; Mortensen et al., 2016; Sørensen et al., 2017; Milioto et al. 2017.

In this paper, we aim to provide an approach that fulfills the necessities of rapid food security, assessment, planning, exploitation, and management of agricultural resources; we propose a framework to efficiently localize and classify four types of tropical fruits (coconut, banana, mango, and papaya). We pursue the latter by a method to automatically identify and segment roads, so that fastest and safest ways to transport crops to adjacent warehouses or security points can be detected.

To do so, we used two supervised deep CNNs; the first CNN model performs the task of object localization, to localize and classify the type of trees. The locations of the trees are not only used to control agricultural resources, but also in scenarios of natural disasters they can be compared with the previous state to have a better understanding on how local agriculture and food security were affected. This information can directly inform and accelerate subsequent relief efforts. Additionally, we propose a method to determine the density of each of these trees to improve productivity, based on the detection results of the first CNN and presented as Density Maps to quickly comprehend the condition of the agricultural site.

The second CNN model performs a semantic segmentation, that masks the streets from the aerial imagery to help identify local transportation infrastructure and, in the scenario of natural disasters, evaluates the damage, proposing a proper plan to distribute aid across affected areas. Ultimately, we introduce a method to optimize the harvesting process, based in specific sceneries, such as maximum time, path length, and location of the warehouse and security points.

2. Data

2.1. Data for the first CNN: Object Localization model

For this experiment, we used UAVs high-resolution imagery over satellite images, the latter is easily affected by cloudy environments. Also, freely available satellite images have lower resolution than UAV imagery. The imagery was captured in October 2017 and was made available in early 2018 as part of an Open AI Challenge coordinated by WeRobotics, Pacific Flying Labs, OpenAerialMap and the World Bank UAVs for Disaster Resilience Program. We participated in this challenge that aim to crowdsource the development of automated solutions for the analysis of aerial imagery; with specific focus on humanitarian, development and environmental projects.

A total of 80 km^2 of high resolution (under 10 cm) aerial imagery was obtained from the Kingdom of Tonga, covering four areas of interest (with a combination of rural and urban areas). The first three covered 10 km^2 each, and the latest covered 50 km^2. The spatial resolution of the optical imagery is 4 cm or 8 cm depending on the Area of Interest.

We created the training data by selecting the imagery from the 50 km^2 area with 8 cm of precision and further used it in the first supervised CNN. We obtained labeled imagery through the Humanitarian OpenStreetMap community, where experts label every type of tree from this aerial imagery with these tree classes: coconut, banana, mango, and papaya.

To prepare the training data for the first CNN model, we split the original full-size aerial imagery into square patches with predefined resolution ($256 \times 256 \times 3$). In order to increase the sample of training data, we used data augmentation techniques, including random horizontal and vertical flipping and random rotations having a result of 27,293 labeled images. The patches are intentionally overlapped until half of the subdivision resolution – because some trees may have been split and will not be recognized correctly – securing that at least one patch can entirely cover each tree. The patches are labeled by vectors that contain position, size and type of tree (Fig. 1); this will be further explained in Section 3.1.

[4] Deep learning is a branch of machine learning that refers to multi-layered interconnected neural networks that can learn features and classifiers at once, i.e., a unique network may be able to learn features and classifiers (in different layers) and adjust the parameters, at running time, based on accuracy, giving more importance to one layer than another depending on the problem. End-to-end feature learning (e.g., from image pixels to semantic labels) is the significant advantage of deep learning when compared to previous state-of-the-art methods (Nogueira et al., 2017).

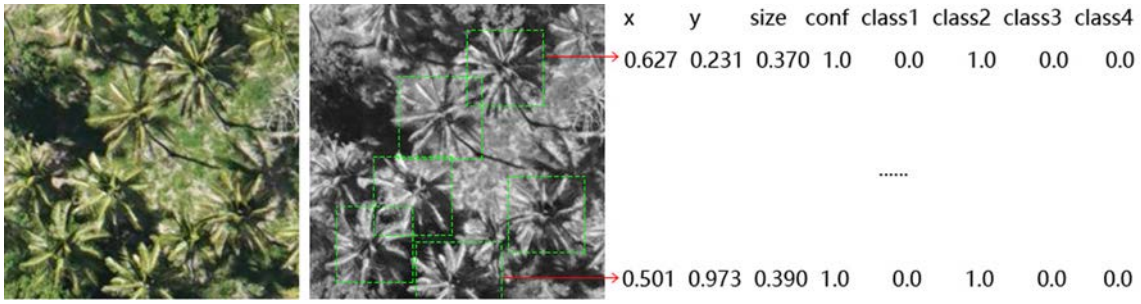


Fig. 1. One patch is exemplifying how the training data was prepared, to be fed to the first CNN.

2.2. Data for the second CNN: Semantic segmentation model

The training data used in the second CNN is from the ISPRS commission II/4 benchmark on Urban Classification and 3D Building Reconstruction and Semantic Labeling These data correspond to the urban area of Potsdam, Germany, and consists of high-resolution True Ortho Photo and their respective Digital Surface Models. This data has been classified manually into six land cover classes: impervious surfaces, buildings, low vegetation, trees, cars, and background (Fig. 2). We split the imagery into square patches of 256 × 256 × 3 without overlapping, achieving a training data of 20,102 images.

3. Procedure

3.1. Classification and location of trees

This CNN model is trained with the training data described in the subchapter *Data for the first CNN: Object Localization model*. This model is able to classify and locate different trees species. The CNN takes one square RGB image of 256 × 256 × 3 as input and provides the corresponding prediction. At the end of the prediction process, the localization results are assembled. If the distance between two or more recognized trees – of the same species – is less than a predefined threshold, the latter are considered as one, and their locations are averaged.

The architecture of CNN is based on a modified YOLO model. As introduced by Redmon et al., YOLO works with a prediction grid, and each cell of the grid is responsible for recognizing one object. Objects are predicted as one or more bounding boxes with a confidence value and a one-hot vector that represents the type of the object; in this experiment, the species of the trees. The confidence value reflects the probability of the cell containing an object and how accurate the bounding box is. Bounding boxes with confidence values larger than a user-defined threshold are kept and are rendered as a result. In our case, the prediction grid is 5 × 5, where each cell predicts one bounding box and four classes. A bounding box is represented by four values: x, y, the radius of the object and confidence value. Since trees seen from above are mainly circular, the width and height of the bounding box are simplified by radius. Therefore, the output is a tensor or a three-dimensional matrix of 5 × 5 × 8. We set the threshold for the confidence values to 0.8. We overlap the patches in order to avoid missing a tree localization when several trees are found in one cell. The process of cells activation is illustrated in Fig. 3.

The overall architecture of the model can be illustrated in Fig. 4. The initial convolutional layers of the network extract features from the image while the fully connected layers predict output probabilities and coordinates. The network has 24 convolutional layers followed by two fully connected layers (Redmon et al., 2016)

The model adopts sum-squared error as the basis of the loss function, however, as Redmon et al. mentioned, sum-square loss weights the localization, the classification, and the confidence errors equally and destabilize the model, which is not ideal for our task. In order to overcome this issue, two modifications of the loss function are introduced. First, an additional coefficient is multiplied to the confidence error and second, the ground truth of confidence value (which is either 0 or 1) is used as the coefficient of the localization and the classification errors. Therefore, the confidence value gains higher priority in training and increases the accuracy of the model in detecting the existence of trees. Moreover, the penalty of localization error only happens when the ground truth tree exists. Let C and C′, B and B′ and T and T′ be the confidence values, the bounding boxes (x, y coordinates, and size) and the species of trees of the ground truth and the prediction respectively, λ be the coefficient for confidence error (in our practice is set to 5) and N be the number of grid cells (which is 25 in our case). Then the loss function can be written as follow:

$$\sum_{i=0}^{N} C_i(B_i - B_i^{'})^2 + \sum_{i=0}^{N} C_i(T_i - T_i^{'})^2 + \lambda \sum_{i=0}^{N} (C_i - C_i^{'})^2$$

Before training the model, we split the dataset in a 60–5–35 ratio for training, validation, and testing respectively. The output of the first CNN model retrieved the location and class of trees in pixel space. In chapter 3. We will further discuss the performance and results of the model.

3.1.1. Density and heat maps

Density maps can provide valuable insight into natural scenarios such as agriculture because they can communicate the characteristics of geo-data, e.g., the concentration of trees in space. In order to determine the density of the detected trees, we map their locations back into their geo-coordinates. We employ the Gaussian Kernel to determine the density of each class of trees: let p be the positions of all the retrieved trees, N be the number of all trees, the density at a given location p' can be calculated as:

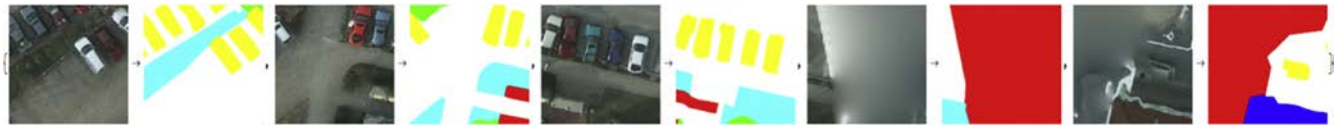


Fig. 2. An example of how the training data was prepared, to be fed to the second CNN.

Fig. 3. The first image shows one patch been subdivided in a grid of 5 × 5, the second image exemplifies the process of the cells being activated, and the third image shows the local coordinate of the founded trees.

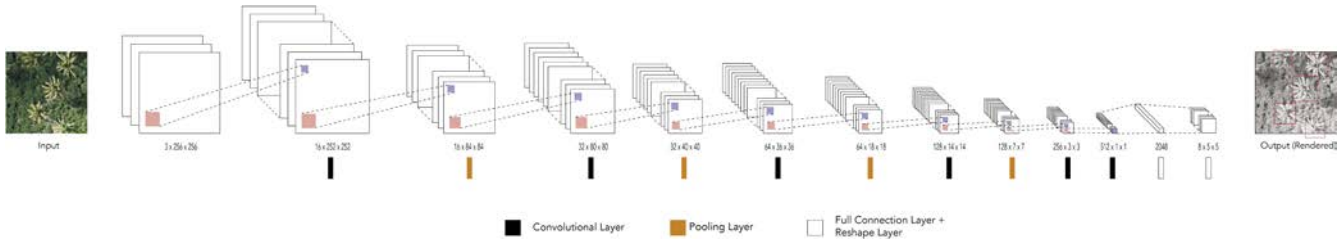


Fig. 4. The architecture of the YOLO model where the disposition of the convolutional, pooling and reshape layers are explained.

$$\sum_{i}^{N} \exp(-\sigma^2 ||p^{'} - p_i||^2)$$

We rendered this results as Density Maps for each species, and later an additional one for all species (Fig. 5). Out of 12,945 true positive trees located, 10,136 trees where coconut trees, representing 78% of the detected trees, 2340 bananas trees, 55 papaya trees, and 173 Mango trees.

3.2. Street detection

The street detection utilizes the method of pixel-wise semantic segmentation to extract the streets out of the original aerial imagery. The semantic segmentation is a process that takes an RGB image as the input and produces an equal-size image that is pixel-wised colored based on semantic labels as the output. All the pixels that have the same label are colored identically. For this task, we trained another CNN

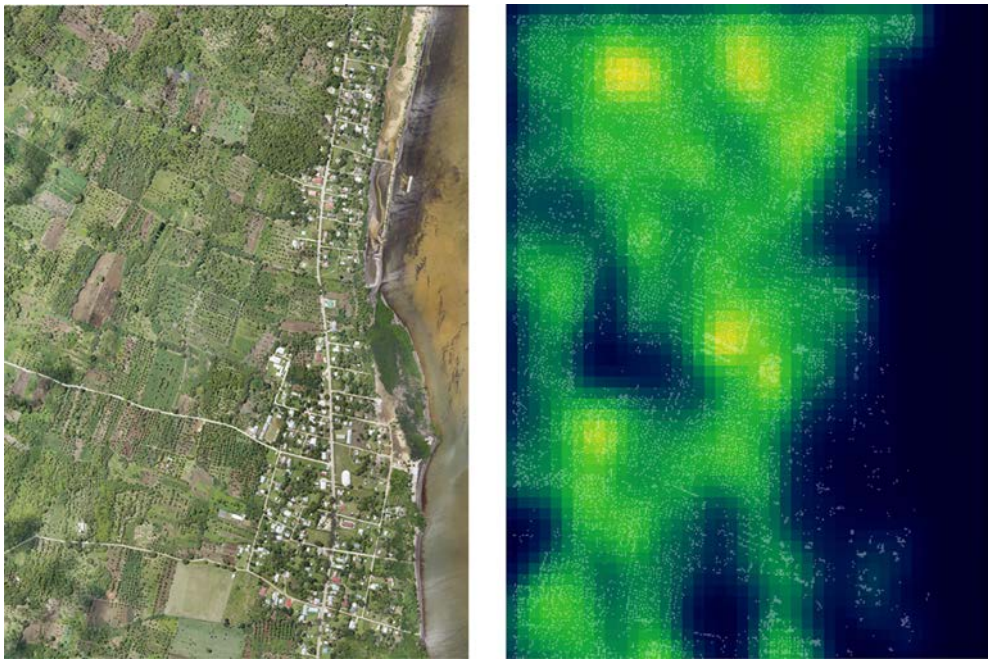


Fig. 5. The original UAV imagery, and the corresponding Density map displaying the localization of all species.

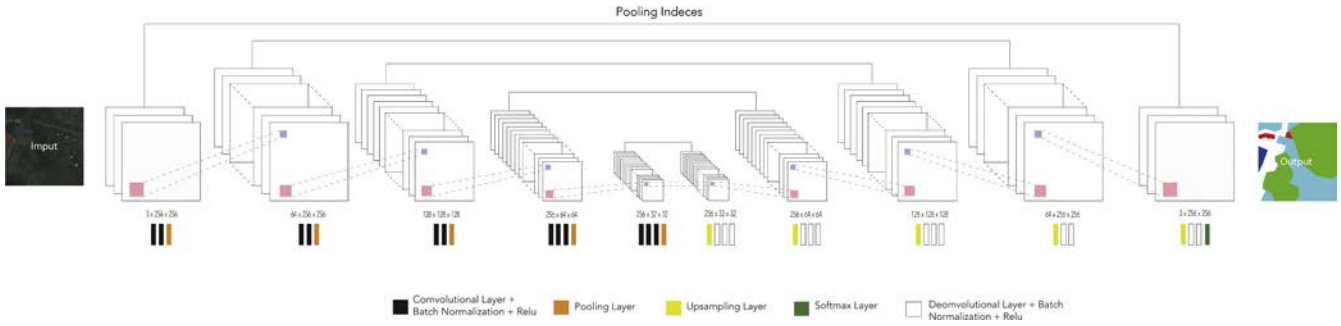


Fig. 6. The architecture of the SegNet model where the disposition of the convolutional, pooling, up-sampling, softmax and normalization layers are explained.

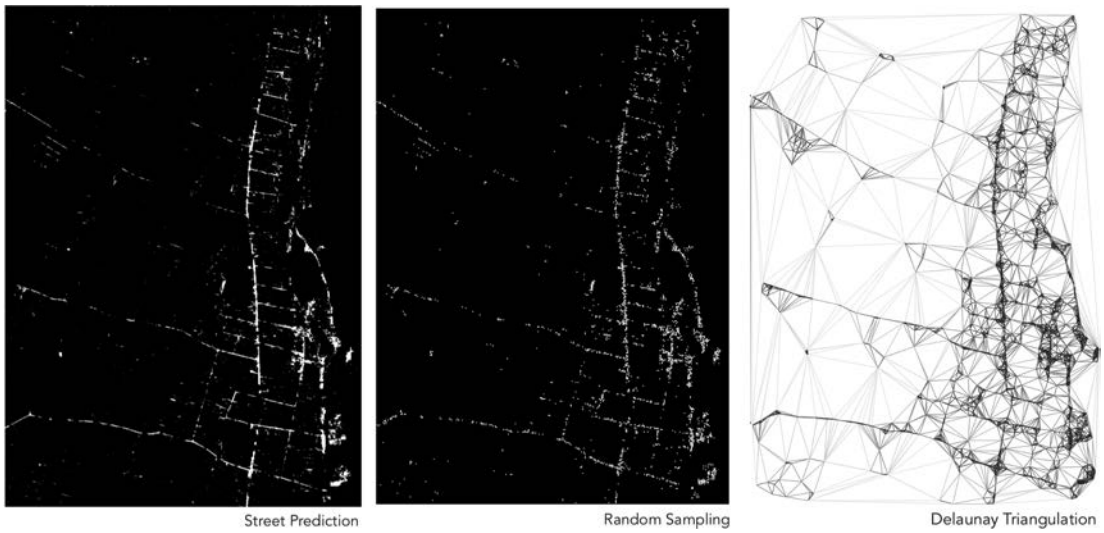


Fig. 7. The first image, the extract layer form the SegNet model corresponding to the street network, second images, the process from pixels to scatter points followed by the Delaunay triangulation, and the third images the spatial pattern of the streets.

model with the ISPRS commission II/4 dataset. The subdivision process of input imagery and treatment was described in the subchapter *Data second CNN: Semantic Segmentation model*. We highlight that unlike the tree localization and classification training data, the patches for the street recognition have no overlap. After all, patches are processed, outputs are assembled. From this output, we extracted only the street layer and highlighted it on a black and white image.

The segmentation model is based on a modified SegNet model (Badrinarayanan et al., 2015) with the input size of $256 \times 256 \times 3$; the input image is processed by a set of hierarchical convolutional modules to reduce the size and gain much more channels. Each module consists of three to five convolutional layers with each one followed by one batch normalization layer. At the end of each module, there are one pooling layer and one activation layer. Then, the compressed images are feed into a set of hierarchical up-sampling modules. Each module starts with an up-sampling layer and followed by several convolutional layers, and the last one is followed by an activation layer as well. The pooling layer and the up-sampling layer in the modules of the same hierarchy share their pooling indices. The overall architecture of the model is illustrated in Fig. 6.

3.2.1. Path optimization

The detection results are irregular and for some streets disconnected; the leading causes of are: first, the original image may be affected by the distortion caused by the merging of many UAV-images into one, secondly, the street may be covered by tree crowns and other objects, which makes it difficult to keep consistency; besides to the instability of the detection model.

Instead of trying to extract precise lines that represent the street network from the image, an alternative method is proposed. With the hypothesis that the probability of two disconnected street segments is one street depending on their distance in-between, we determine that the shorter the distance, the higher the probability of belonging to the same street. Following this assumption, a random subsampling process is made on the resulting image, where each pixel labeled as a street has a certain probability of being a node and taken into account in the next stage of the process. The subsampling phase converts the street system from image to scattered points, where their density represents the hierarchy (importance) of the street. The scattered points are stored in a list, and their positions inside this list are considered as their index.

A Delaunay triangulation is made on the scattered points, representing the whole street network. Its edges are weighted by the inversed square of their lengths, meaning shorter edges have higher priority. Then, the shortest path between two points of the network can be calculated by the Dijkstra algorithm[5] (Dijkstra, 1959), obtaining results that match the spatial pattern of the streets (Fig. 7).

By overlapping the density map and the street network, the nodes of

[5] Dijkstra algorithm, or Dijkstra shortest path algorithm, proposed by the computer scientist Edsger W. Dijkstra, is an algorithm that finds the shortest path between nodes in a weighted graph. The algorithm exists many variants: the most common one fixes a node as the source, iterates over all the other nodes and produces a shortest path tree. The original one, however, stops early when the target node is reached and therefore only returns the shortest path between the source and the target nodes.

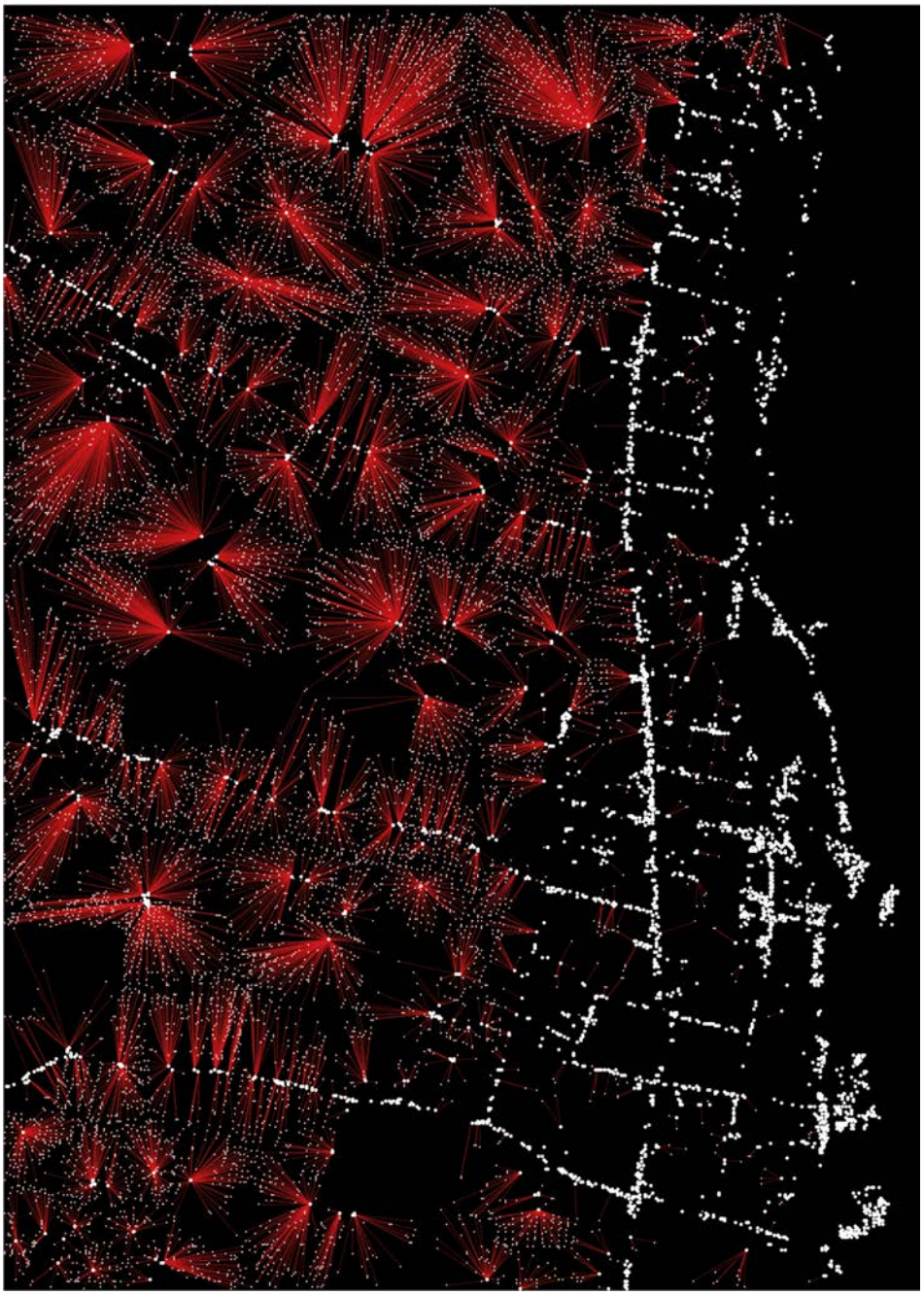


Fig. 8. Weighted nodes by the number of reachable trees within a predefined threshold of the maximal distance.

the network can be weighted by the number of trees they reach; as seen in Fig. 8. The overlapping between localization data and segmentation data allows us making queries about the network. For instance, we could ask the optimal path to harvest as many crops as possible within 10 min of traveling or vice versa. The starting point, ending point, number of trees and time of the query are user-specified, and the search process is based on optimization algorithms. Therefore, this process adapts to the necessities of a specific scenario.

We use the Genetic Algorithm (Fraser, 1957) for path optimization. It searches for optimal solutions on randomly selected parent solutions by regularly applying mutation and crossover operations and selecting the offspring that gain higher scores on the objective function. The algorithm modifies the solution in its genotype rather than the phenotype. More specifically, we represent each path by a series of key points instead of all the points of it. The key points are in arbitrary sequences and locations, and they are assumed to be visited one after the other until the last point is reached. The in-between path between every two key points is calculated using Dijkstra algorithm on top of the triangulated graph, and the final path is the union of these results. Therefore, the genotype of the path would be the list of key points and the phenotype would be the result of the Dijkstra algorithm. By modifying the sequences and the locations of the key points new paths can be

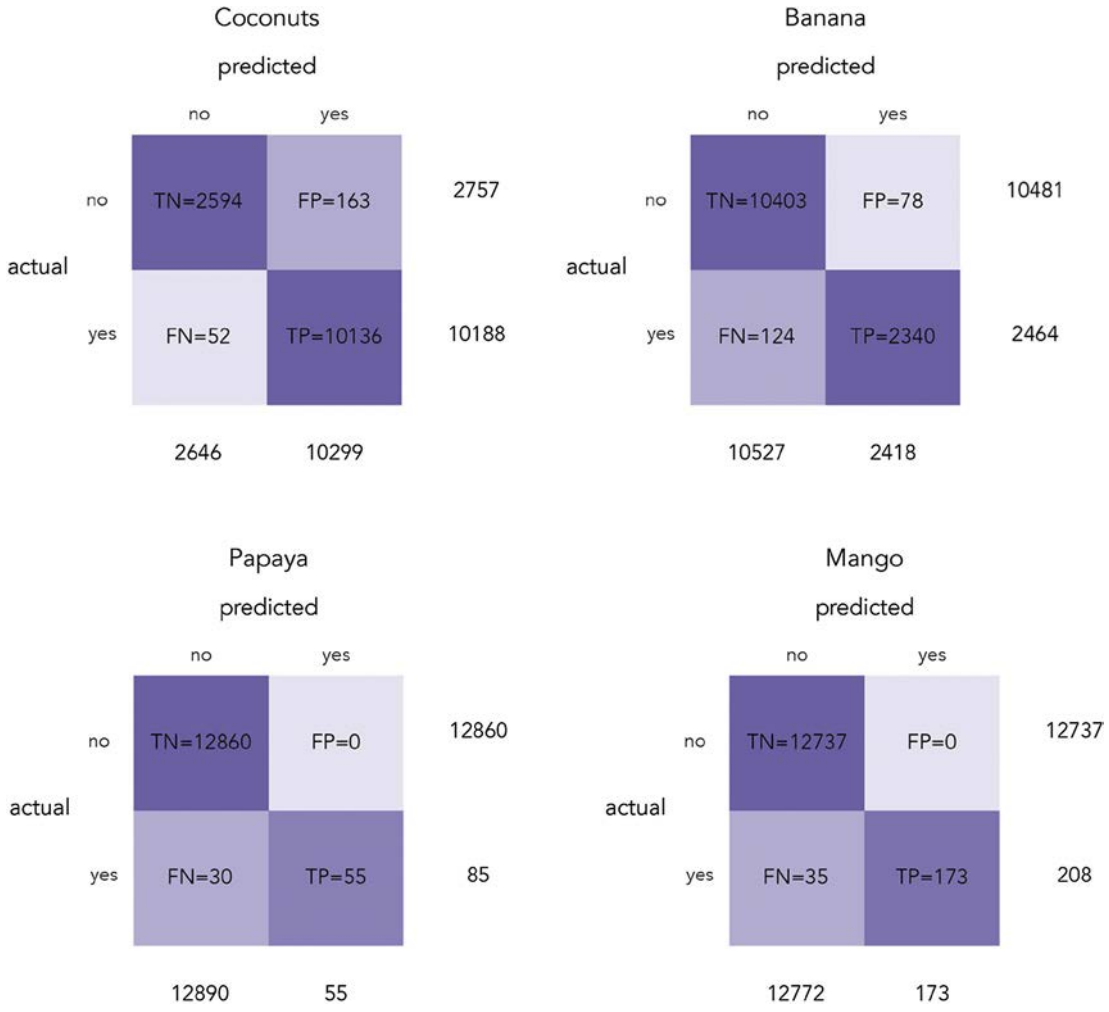


Fig. 9. Confusion matrix for the 4 type of trees.

generated from the original one. During the optimization, the mutation of the path is done by randomly modifying the key points sequence, and the crossover is done by taking two input paths and swapping part of their key points. The number of key points of the path is flexible, and therefore crossover can be applied to paths that have a different number of key points and produces in-between offspring.

We proposed two types of search, as examples of the query process. One is to maximize the number of crops to harvest with a limited length of travel distance, and the other is to minimize the travel distance with a limited number of crops to harvest. The objective functions for each are:

$$n/\max\left(1, \left(\frac{1}{l}\right)^{d}\right)$$

And:

$$\min\left(1, \left(\frac{n}{n'}\right)^{d}\right)/(l+1)$$

where n is the number of crops, l is the length of the path, d is a constant and l', n' are the two bounds respectively.

4. Results and discussion

The performance of the Tree Localization and Classification model was measured by evaluating how precise the classifier was to localize trees correctly. The average Euclidian distance between the center point of the original trees and the predicted trees is 8.86406 pixels (less than one meter). The classifier was able to count 16,457 trees, out of which 12,945 were correctly located. Considering the original 13,393 trees, the overall Localization accuracy of the model is 80%.

We draw a confusion matrix from each type of trees (Fig. 9), in order to evaluate the accuracy of classification of our Model, arranged as follows: Coconut trees: type 1, Banana trees: type 2, Papaya trees: type 3 and Mango trees: type 4. We achieve a Classification accuracy of 98%.

Having for each class:

(Mean TP + Mean TN)/total = 0.987691

Our Misclassification Rate is 1%, based on how often the classifier was wrong.

(Mean FP + Mean FN)/total = 0.00990861

And its Precision is 97% according to how many times it predicts correctly a TP.

Fig. 10. Area of interest on Tonga, with the classification results, from the four types of trees; green: coconut, red: banana, blue: mango, and yellow: papaya. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

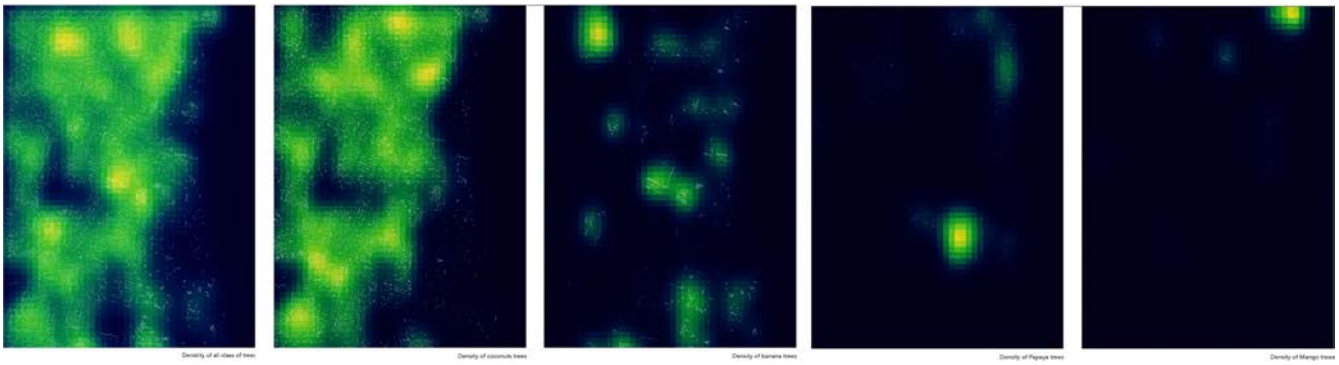


Fig. 11. Density Maps of all species, followed by specific density maps of Coconut trees Banana trees, Mango trees, and Papaya trees.

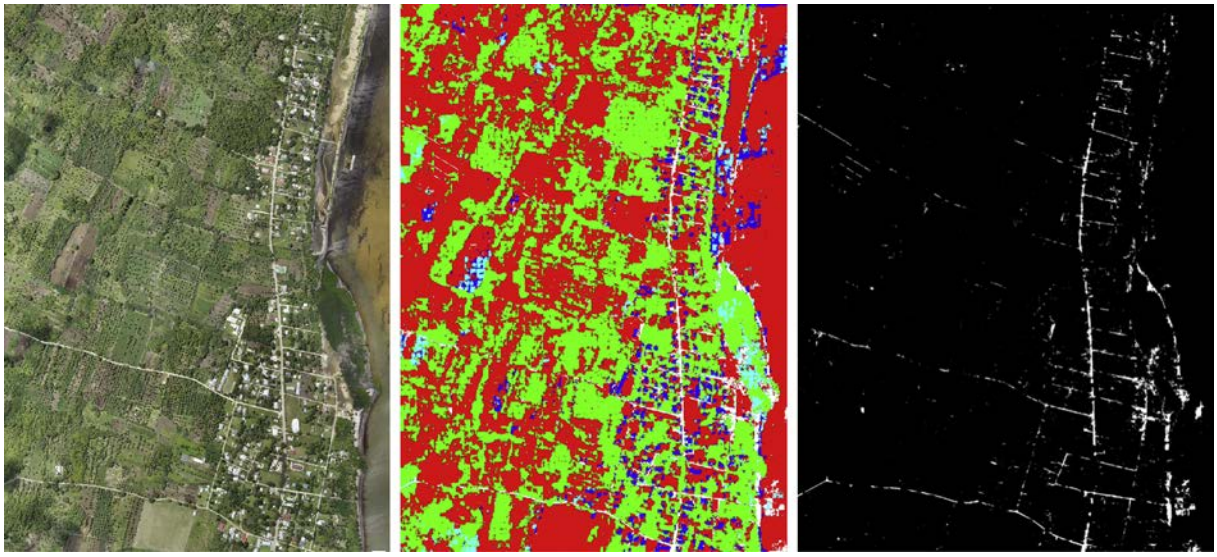


Fig. 12. First images, original aerial imagery, second image the output of SegNet model and third image the layout showing the masking of streets.

Mean TP/predicted yes = 0.97

The F1Score of the model is 0.89, according to the average of the Localization Accuracy (80%) and the Classification Accuracy (98%). Fig. 10 illustrates the localization results. All trees are localized with a bounding box representing the class they belong to; green: coconut, red: banana, blue: mango, and yellow: papaya.

The corresponding density map of the results shows that high-density areas are punctual, whereas mid-density areas appear as large surfaces connecting high-density areas. Null-density areas are displayed in a scattered manner between high and mid-density areas (Fig. 11). In the coconut heat map, the same global heat map reappears. However, in the banana, mango and papaya trees heat map, only high-density separate areas appear.

Dos Santos et al. (2017) demonstrated that the estimation of trees density is an essential first step toward large-scale monitoring. Besides this, it provides a broad view of resource distribution that enables the identification of areas with higher, mid and low densities. These results can yield actions to planning, harvesting, and management of these tropical fruits by the interests of landowners, producer associations, and humanitarian organizations – The importance of improving these actions is essential not only to Tonga's industry and economy but also to the thousands of families who depend on their extraction for subsistence.

The second model was able to discriminate different urban classes (Fig. 12 middle): building footprints in blue, vegetation in green, open spaces in red, and road network in white. By applying a filter, we could mask the streets and see their structure. Since the training data was from an urban scenario, and the site is rural, the accuracy of the model to detect streets was low. For example, some parts of the buildings were mistakenly labeled as streets. Therefore, these results required further post-processing as described in the subchapter *Path Optimization*.

Resampling and graph-making processes provide a systematic approach to overcome this issue. The precise extraction of the street can be bypassed, making it possible to use these results in applications like pathfinding and path recommendation. The two proposed tasks for pathfinding show that this bypassing is possible and useful. Some path query results are illustrated in Fig. 13 and more are shown in Appendix A.

In order to validate our approach, we applied the tree localization model to a different dataset, one from the three areas of interest that covered 10 Km^2 with a resolution of 8 cm (explained in the chapter *2.1. Data first CNN: Object Localization model*) (Fig. 14). After successfully obtaining the location of the trees from the imagery we conclude that the tree recognition model can work in different scenarios, and is robust enough to find differences in occlusion, variation, illumination and scale among the retrieved trees.

Besides, we applied the whole pipeline of the experiment to another area of interest in Tonga, successfully retrieving trees and streets. The site and the results are shown in Appendix B. The processing time of this approach is proportional to the size of the site of interest.

5. Conclusion

This paper has investigated the use of Convolutional Neural Networks to efficiently localize and transport four types of tropical trees using aerial imagery. This new approach reduces costs and time of inventory, mapping, harvesting, and management of agricultural resources, and assess the impact of disasters on food security.

We introduce a specific case where this method fulfills the necessities of rapid assessment after natural disasters. Together with two Convolutional Neural Networks models, we have also proposed a method to determine the density of trees and a method to optimize the harvesting process based on specific scenarios.

This experiment provides a framework where we can draw some conclusions about the advantage and disadvantages of Convolutional Neural Networks. The advantages of the models are: reduces the need of featuring engeneering,[6] outperforms other approaches implemented for comparison purposes (feature extraction,[7] area-based techniques statistics, texture, color, and shape-based algorithms) and, it has the faculty to learn the underlying patterns of the data. The disadvantages of the models are: takes a longer time to train than other traditional approaches, needs large datasets, lacks publicly available datasets for researchers to work with, and in many cases, researchers need to develop their own sets of images.

State of the art approaches on this field of study (agricultural objects detection from aerial imagery) involved: different datasets, pre-

[6] Hand-engineered components require considerable time, an effort that takes place automatically.

[7] Feature extraction starts from an initial set of measured data and builds derived values (features) intended to be informative and non-redundant, facilitating the subsequent learning and generalization steps.

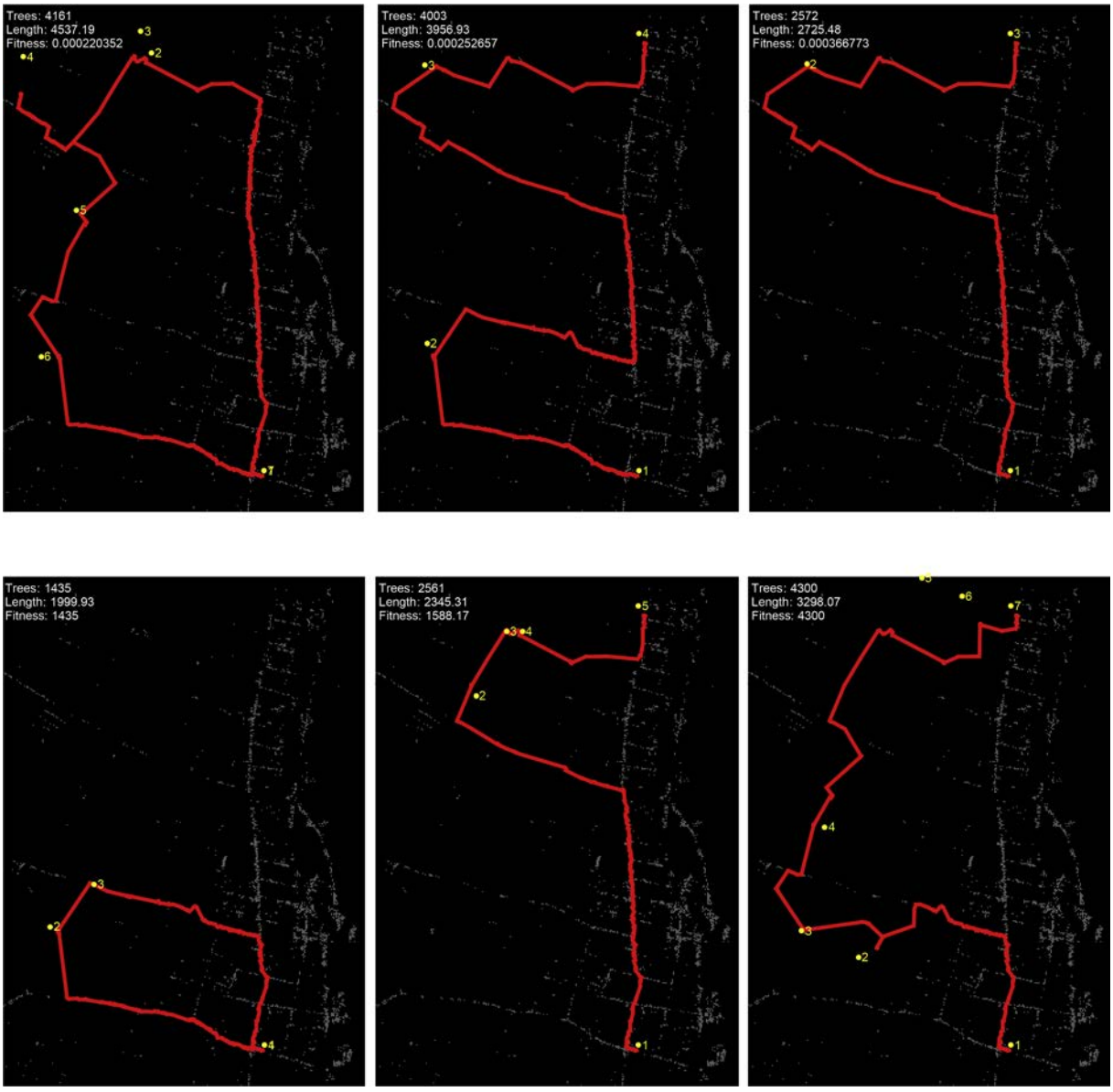


Fig. 13. The first row shows the longest paths accessing a maximal amount of trees; the second-row shows the shortest paths with some accessible trees above a given threshold.

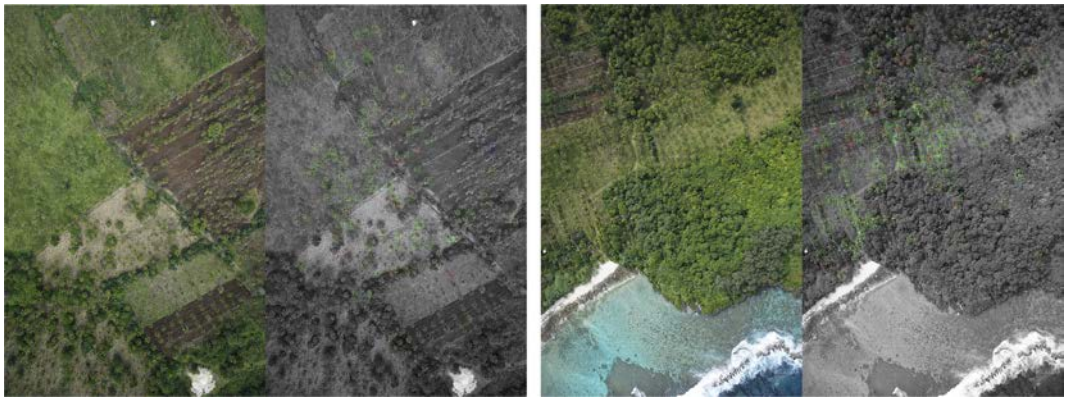


Fig. 14. Aerial imagery used as a validation data.

Table 1
Comparison among existing approaches.

Reference	Task	Data	Labels	Model	Pre-processing	Performance	Score
Chen et al. (2014)	Classification	Hyperspectral imagery on Kennedy Space Center, USA and Pavia city, Italy	13 classes for KSC data and 9 classes for Pavia data	Hybrid of PCA, autoencoder, and logistic regression	Band removal for denoising	Classification Accuracy (CA) F1Score (F1)	KSC: 75.34% (CA) 0.7463 (F1). Pavia: 84.61% (CA) 0.8441 (F1)
Luus et al.(2015)	Classification	Aerial ortho-imagery with a 0.3048-m pixel resolution	21 classes	Author defined CNN	From RGB to HSV, resized to 96 × 96 pixels, creation of multiscale views	Classification Accuracy (CA)	93.48% (CA)
Lu et al. (2017)	Classification	UAV imagery	2 classes	Author defined CNN	Correct distortion, resized to 40 × 40 pixels	Classification Accuracy (CA)	89.5% (CA)
Kussul et al. (2017)	Classification	Satellites imagery	11 classes	Author defined CNN	Calibration, filtering and restoration of missing data	Classification Accuracy (CA)	94.6% (CA)
Mortensen et al. (2016)	Segmentation	Photograph by Sony a7 with 35 mm lens	7 classes	Adapted version of VGG16	Resized to 1600 × 1600 pixels and then divided into 400 × 400 pixels patches	Classification Accuracy (CA) Intersection over Union(IoU)	79% (CA), 0.66 (IoU)
Sørensen et al. (2017)	Classification	Photograph by Canon PowerShot G15	2 classes	DenseNet	Image cropping, random flip horizontally and vertically, random transposing	Classification Accuracy (CA)	97% (CA)
Milioto et al. (2017)	Classification	UAV imagery	2 classes	Author defined CNN	Background separation, making vegetation blobs, resized to 64 × 64 pixels	Classification Accuracy (CA)	Dataset A 97.5% (CA) Dataset B 94% (CA)
Ours	**Localization, Segmentation**	**Aerial ortho-imagery**	**4 classes**	**Adapted version of YOLO and SegNet**	**Divided into 256 × 256 patches**	**Localization Accuracy (LA) Classification Accuracy (CA) F1Score (F1)**	**80% (LA) 97.5%(CA) 0.89 (F1)**

processing techniques, metrics, models, and parameters; it is difficult to compare the current research among them (Kamilaris and Prenafeta-Boldú, 2018); thus, our comparisons have been strictly limited to the used of techniques and the score of each paper. In the following Table 1 a description of the task, data, labels, model, pre-processing, performance and score is displayed.

As explained above the score of the model varies depending on the experiment. Therefore, we compare our results with the ones that had the same validation performance, in the case of Chen et al. they achieved a 0.79 F1Score, and our model achieved 0.89 F1Score. Miliot et al., Sørensen et al., Kussul et al., Luus et al. scored above 90% results on Classification Accuracy, our model can be added to this list since it achieved 97% in Classification accuracy. It is worth mentioning that all the above experiments dealt only with the task of Classification.

In this experiment, we proposed a model that not only classify but locate different class of trees; hence, we have an additional Score performance: Localization Accuracy 80%, this value shows how accurate the model was to located any class of tree. The F1Score – the average value of classification accuracy and localization accuracy – demonstrates that our model is suitable to perform the task of classification and localization of 4 types of trees.

Outlooks

This approach can be used in localization, classification or transportation of resources; for instance, in the assessment of damage in buildings after a natural disaster, food supply chain, urban and regional planning, etc. Other potential uses could be informal settlements detection, and more specifically the monitoring of rooftop materials as a means determine localized socio-economical conditions.

Data and code

The data and code of this pipeline are open source and can be accessed via this link: https://github.com/guozifeng91/south-pacific-aerial-image.

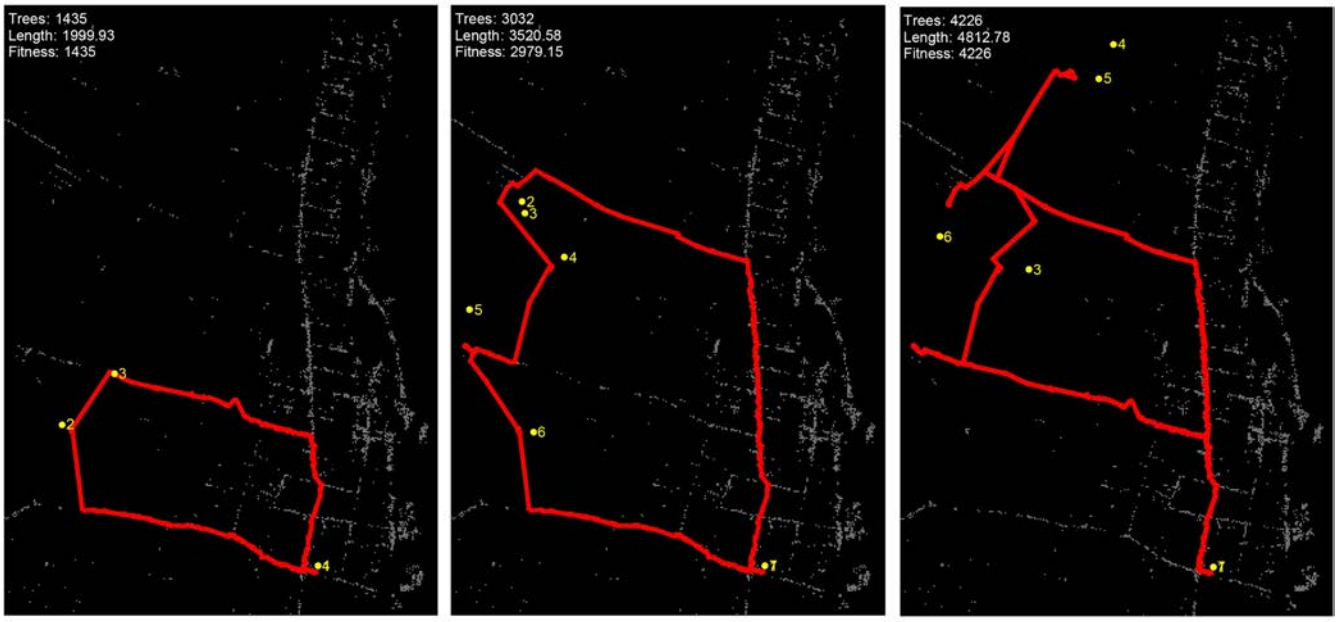
Declarations of interest

None.

Appendix A

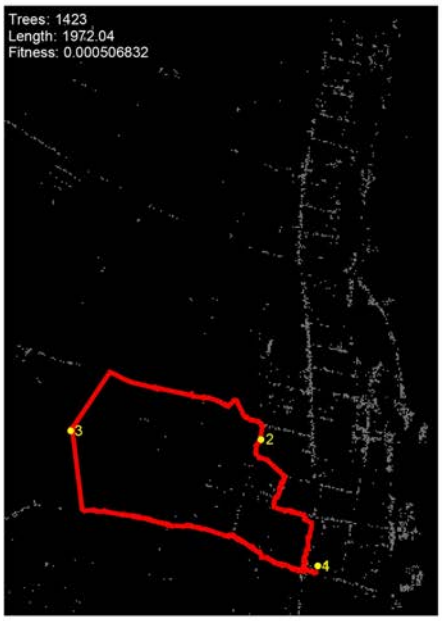
Path optimization

Get a path that contain as many trees as possible with its length belows an upper bound.

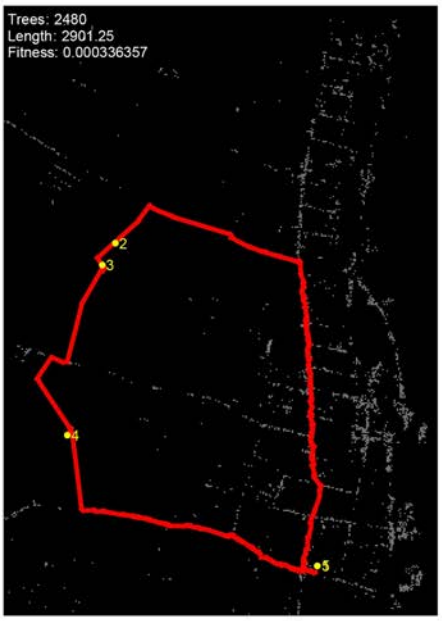


At Most 2000 Pixels At Most 3500 Pixels At Most 5000 Pixels

Get a path as short as possible with the tree number larger than a lower bound.



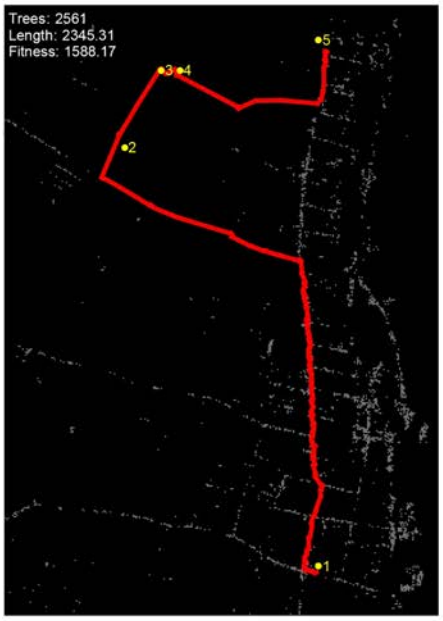
At Least 1000 Trees



At Least 2500 Trees

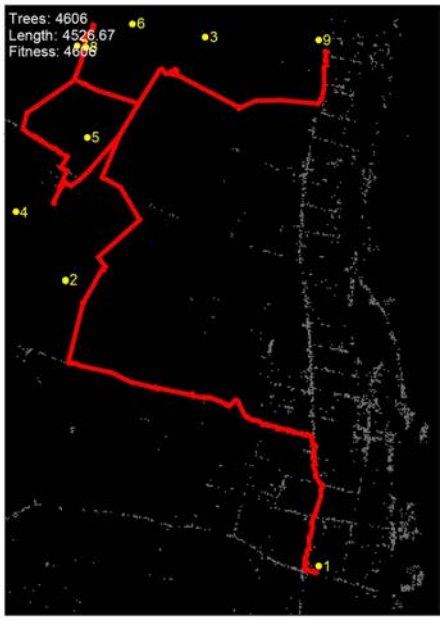
At Least 4000 Trees

Get a path contains as many trees as possible with its length belows an upper bound. The path starts and ends at different locations.



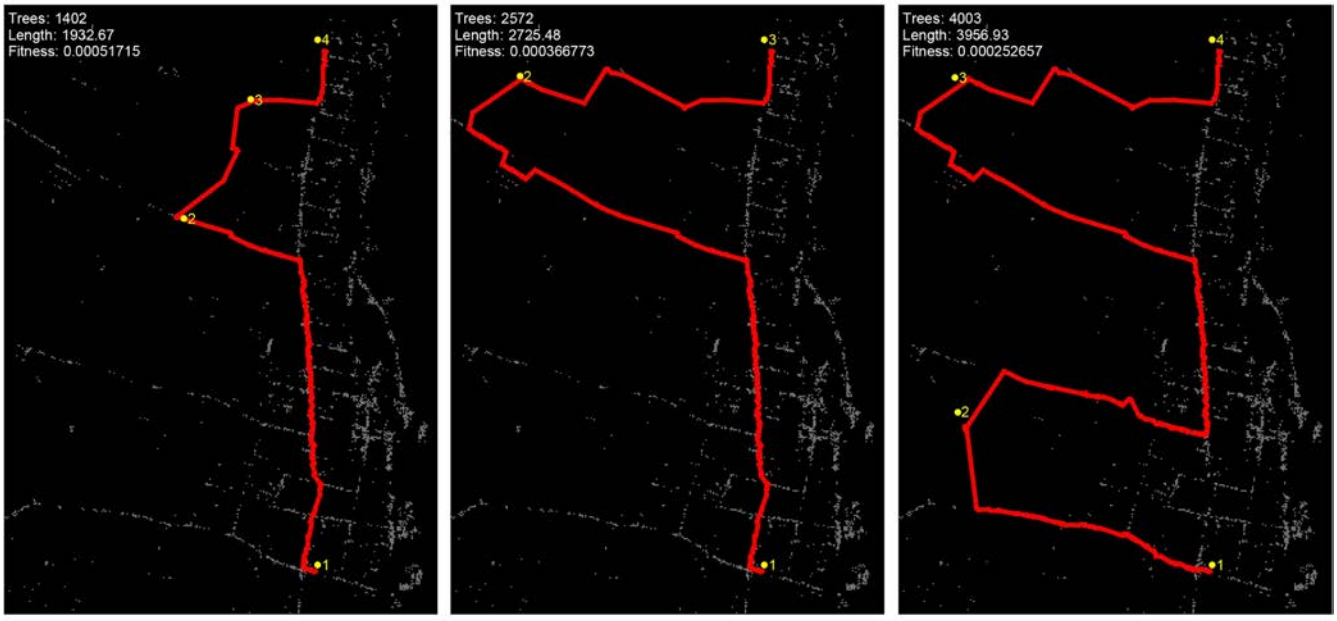
At Most 2000 Pixels

At Most 3500 Pixels



At Most 5000 Pixels

Get a path as short as possible with the tree number larger than a lower bound. The path starts and ends at different locations.



At Least 1000 Trees At Least 2500 Trees At Least 4000 Trees

Appendix B

Complete framwork apllied in a Aera of interest in Tonga.

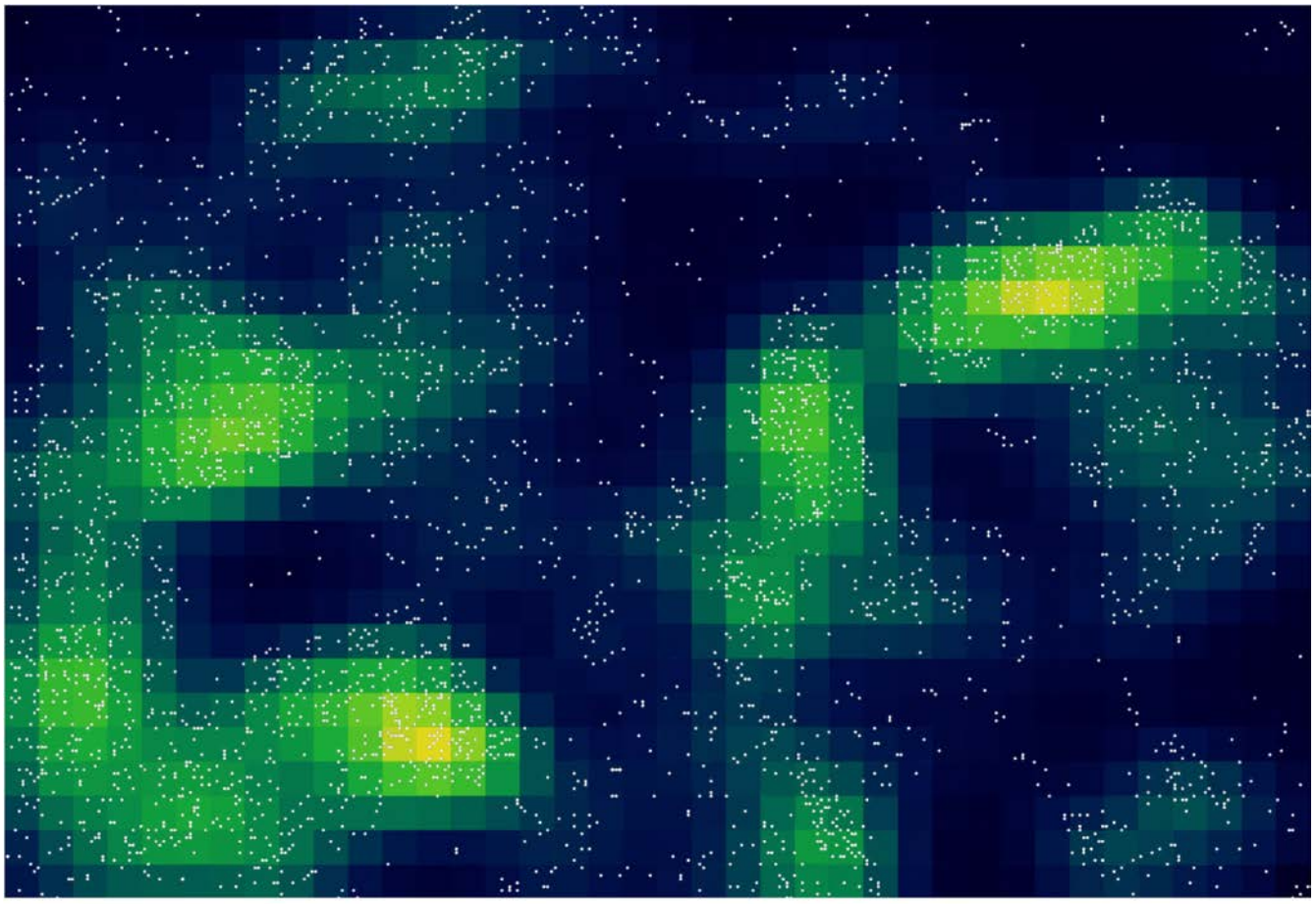
Sample selected from Tonga

We continue validating our Model, with an image from Tonga, where we select a sample of 15337 Width by 10722 Height pixels, where the Model locate and count 3493 trees, and the classes found where: banana tree 278, coconut tree 3214, mango tree 1, and papaya tree 0. In fig. 3. we can observed the result.

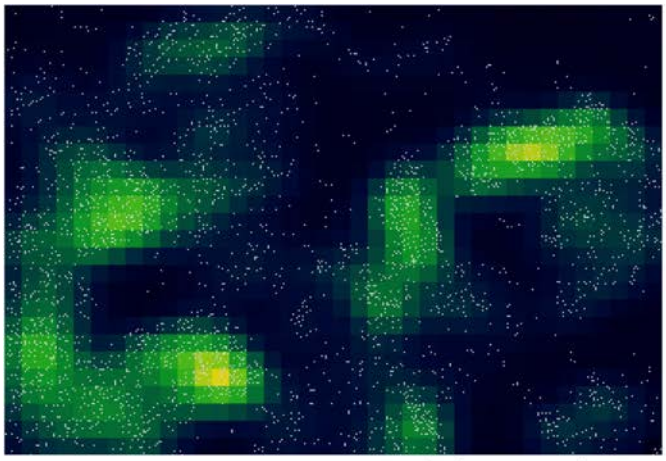
The result can be render as geolocation of the center point of each tree, making the exercise of security & transportation after a natural disaster more efficient, and time saving. An average performance of the Model (FScore) regarding precision and accuracy is 89%.

Satellite Image Tonga

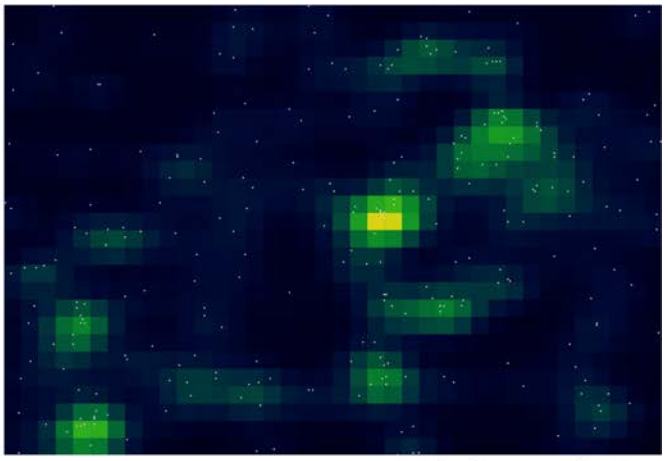
Predicted Location and Class of trees



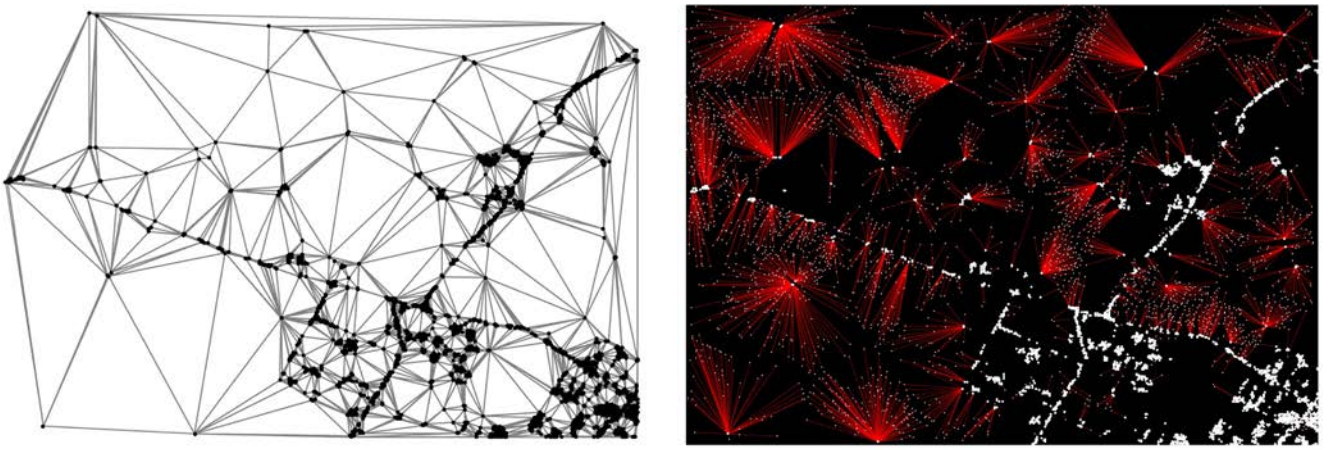
Density of all trees form Tonga



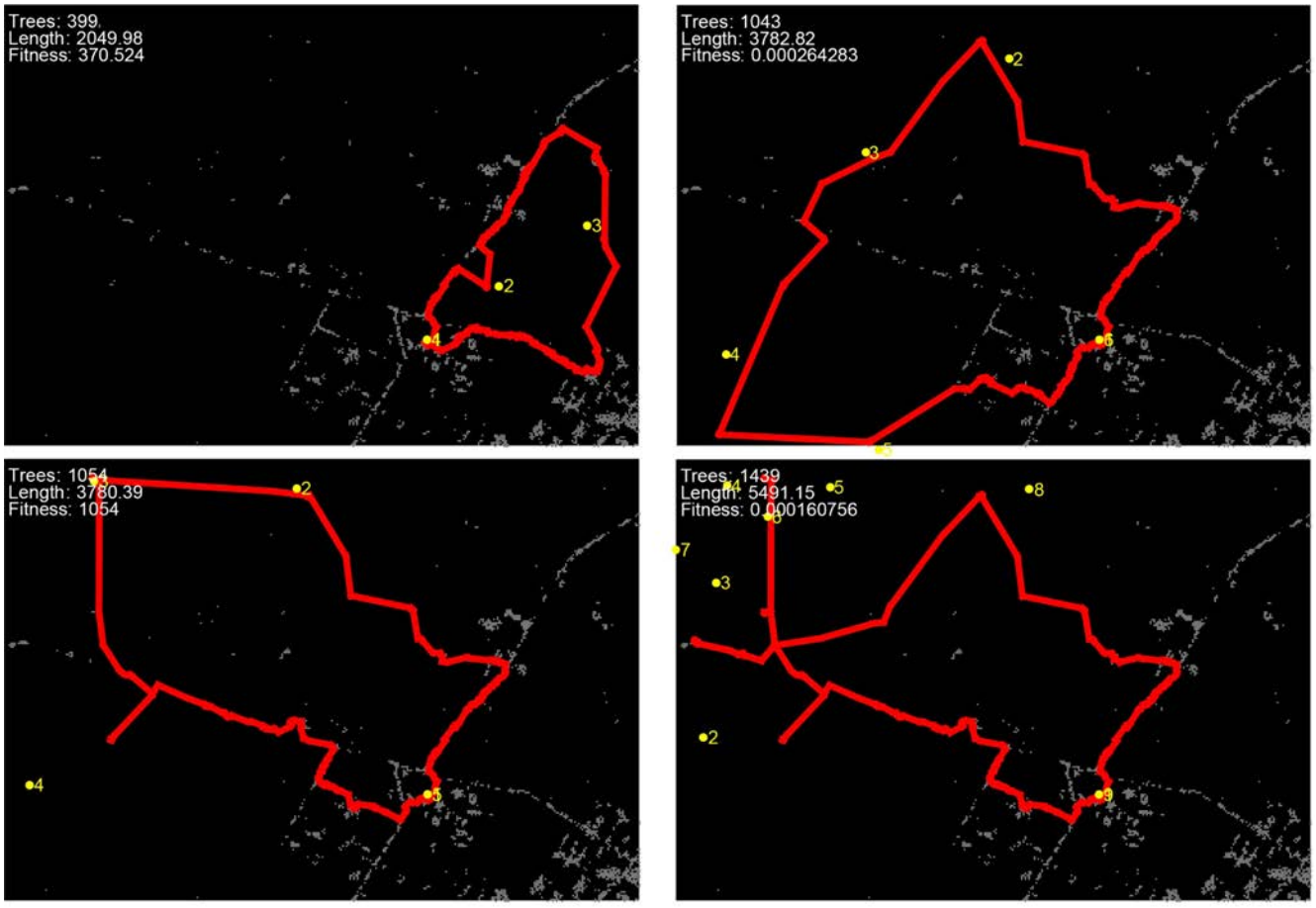
Demsity of cononuts trees from Tonga



Density of banana trees from Tonga



Street graph and tree assignment in validation set



Most Trees for Path Length of 2000 and 4000 Pixels

Shortest Paths for at Least 1000 and 1500 Trees

Appendix C. Supplementary material

Supplementary data to this article can be found online at https://doi.org/10.1016/j.compag.2019.03.028.

References

Badrinarayanan, V., Kendall, A., Cipolla, R., 2015. Segnet: a deep convolutional encoder-decoder architecture for image segmentation. arXiv preprint arXiv:1511.00561.

Chen, Y., Lin, Z., Zhao, X., Wang, G., Gu, Y., 2014. Deep learning-based classification of hyperspectral data. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 7 (6), 2094–2107.

Cheng, G., Han, J., 2016. A survey on object detection in optical remote sensing images. ISPRS J. Photogramm. Remote Sens. 117, 11–28.

Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L., 2009. Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, pp. 248–255.

Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. Numer. Math. 1 (1), 269–271.

dos Santos, A.M., Mitja, D., Delaître, E., Demagistri, L., de Souza Miranda, I., Libourel, T., Petit, M., 2017. Estimating babassu palm density using automatic palm tree detection with very high spatial resolution satellite images. J. Environ. Manage. 193, 40–51.
Everingham, M., Eslami, S.A., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A., 2015. The pascal visual object classes challenge: A retrospective. Int. J. Comput. Vision 111 (1), 98–136.
FAO, 2015. The impact of disasters on agriculture and food security, 76. https://doi.org/F0134/EN.
Fraser, A.S., 1957. Simulation of genetic systems by automatic digital computers I. Introduction. Aust. J. Biol. Sci. 10 (4), 484–491.
Gougeon, 1995. A Crown-following approach to the automatic delineation of individual tree crowns in high spatial resolution aerial images. Can. J. Remote Sens. 21, 274–284.
Halavatau, S.M., Halavatau, N.V., 2001. Food Security Strategies for the Kingdom of Tonga (PDF), Working Paper number 57, United Nations Centre for Alleviation of Poverty Through Secondary Crops' Development in Asia and the Pacific (CAPSA), archived (PDF) from the original on 10 September 2015.
Hassaan, O., Nasir, A.K., Roth, H., Khan, M.F., 2016. Precision forestry: trees counting in urban areas using visible imagery based on an unmanned aerial vehicle. IFAC-PapersOnLine 49 (16), 16–21. https://doi.org/10.1016/j.ifacol.2016.10.004.
Hung, C., Bryson, M., Sukkarieh, S., 2006. Vision based shadow aided tree crown detection and classification algorithm using imagery from an unmanned airborne vehicle. In: International Symposium for Remote Sensing of the Environment.
Kamilaris, A., Prenafeta-Boldú, F., 2018. Deep learning in agriculture: a survey. Comput. Electron. Agric. 147, 70–90. https://doi.org/10.1016/j.compag.2018.02.016. ISSN: 0168-1699.
Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems, pp. 1097–1105.
Kussul, N., Lavreniuk, M., Skakun, S., Shelestov, A., 2017. Deep learning classification of land cover and crop types using remote sensing data. IEEE Geosci. Remote Sens. Lett. 14 (5), 778–782.
LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86 (11), 2278–2324.
Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3431–3440.
Lu, H., Fu, X., Liu, C., Li, L.G., He, Y.X., Li, N.W., 2017. Cultivated land information extraction in UAV imagery based on deep convolutional neural network and transfer learning. J. Mountain Sci. 14 (4), 731–741.
Luus, F.P., Salmon, B.P., van den Bergh, F., Maharaj, B.T., 2015. Multiview deep learning for land-use classification. IEEE Geosci. Remote Sens. Lett. 12 (12), 2448–2452.
Milioto, A., Lottes, P., Stachniss, C., 2017. Real-time blob-wise sugar beets vs weeds classification for monitoring fields using convolutional neural networks. Proceedings of the International Conference on Unmanned Aerial Vehicles in Geomatics. Bonn, Germany.
Mortensen, A.K., Dyrmann, M., Karstoft, H., Jørgensen, R.N., Gislum, R., 2016. Semantic segmentation of mixed crops using deep convolutional neural network. International Conference on Agricultural Engineering. Aarhus, Denmark.
Nogueira, K., Penatti, O.A.B., dos Santos, J.A., 2017. Towards better exploiting convolutional neural networks for remote sensing scene classification. Pattern Recogn. 61, 539–556. https://doi.org/10.1016/j.patcog.2016.07.001.
Pinz, 1991. A computer vision system for recognition of trees in aerial photographs. In: International Association of Pattern Recognition Workshop, pp. 111–124.
Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 779–788.
Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Berg, A.C., 2015. Imagenet large scale visual recognition challenge. Int. J. Comput. Vision 115 (3), 211–252.
Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.
Sørensen, R.A., Rasmussen, J., Nielsen, J., Jørgensen, R., 2017. Thistle Detection using Convolutional Neural Networks. EFITA Congress, Montpellier, France.