# Automatic Citrus Tree Detection from UAV Images based on Convolutional Neural Networks

Maciel Zortea*, Maysa M. G. Macedo*, Andrea Britto Mattos*, Bernardo C. Ruga* and Bruno H. Gemignani‡
* IBM Research
Av. Paster, 146, Rio de Janeiro,Brasil, 22290-240
Email: {mazortea,mmacedo,abritto,bcruga}@br.ibm.com
‡3DGEO
Email: bruno@3dgeo.com.br

***Abstract*—In the context of agribusiness, the task of tree detection and counting is fundamental for updating the forest inventory and the management plan of a farm. Because it constitutes an exhausting manual process, automated approaches dealing with remotely sensed data have proven to work effectively for this task, while reducing human effort and delivering results in a much smaller time frame. In parallel, the increasing popularity of drones has allowed the use of unmanned aerial vehicles (UAVs) as a low-cost alternative for acquiring aerial images, later processed by computer vision-based methods. In this work, we deal with the problem of citrus trees detection at images of high-density orchards, captured by UAVs. Because of the format in which the citrus trees are arranged at our input images, previous detection approaches working with sparsely distributed trees are not suitable for our data; therefore, we propose a novel approach based on Convolutional Neural Networks (CNNs). Our method is divided in three steps: (1) using a CNN on a sliding window scheme for inferring the line centers of the tree rows; (2) segmenting the probability areas of the line centers followed by the partition of the tree rows in candidate regions; and (3) using a CNN for classifying the candidates into tree regions. Experiments on seven different test sites achieved overall accuracy values of 94%.**

## I. Introduction

Counting trees on a commercial orchard is essential for calculating productivity, taxes, and managing pests when they result in the total loss of the tree. Currently, a traditional sampling method for estimating stocks of fruit trees employed at some Brazilian farms is done manually, where an operator physically registers each tree by a GPS device. This results in a very expensive and slow method, that does not allow a dynamic data update.

Aiming to overcome this problem, unmanned aerial vehicles (UAVs) are increasingly being used for crop management, often paired with specialized sensors and/or computer vision-based methods for automatic image analysis. UAVs are able to reduce drastically the tree counting process, in comparison to the manual procedure. Considering the same orchard, depending on the crop size, the aforementioned traditional method could take a month, while imaging by UAVs combined with manual counting would take a week, and the UAV imagery using a computational method of tree detection could be done in one day. This example makes it clear that the potential of UAVs is indisputable, which holds not only for large enterprises but also for the medium and small producers. However, as reported in [1], analyses with drone data are much more frequent for certain crops, such as corn and wheat, as opposed for citrus, for example.

Nevertheless, according to data from the United States Department of Agriculture (USDA), the global orange production for 2017/18 is forecast to 49.3 million metric tons, with Brazil being the largest producer of orange juice and accounting for three-quarters of global exports [2]. Therefore, the goal of this work is to explore citrus tree detection from aerial pictures captured by UAVs, using images acquired on a commercial orchard in Brazil. Our goal is to evaluate image-only information so our approach may be scaled for different UAVs and camera models and be suitable for small producers as well.

Fig. 1. Example of UAV images from Brazilian citrus orchards used in this research.

The difficulty of the automatic counting of our dataset comes mainly from the high density of the trees, resulting in a barely visible limit between the crowns, as shown in Figure 1.

This paper is structured as follows. In Section II, we report previous works on detecting and counting trees automatically. In Section III, we describe the steps of our tree detection method, that is based on Convolutional Neural Networks (CNNs). Section IV describes experiments on different test sites followed by a discussion in Section V, and the paper is concluded on Section VI.

## II. Related work

Previous works have addressed the tasks of extracting and counting trees from remotely sensed data. These ap-

proaches differ widely due to several aspects, such as: (i) the imaging acquisition device - UAVs [3], satellite images [4], Laser/LIDAR surveys [5], Aerial image [6], or a combination of these; (ii) the methods employed - Hough Transform [7], Mathematical Morphology [8], CNNs [9], [10], among others; and (iii) the plantation characteristics itself.

Concerning the latter, some approaches are only effective on spatially well-arranged trees and are not able to produce satisfactory results for dense plantations with overlapping of tree crowns [9]. On this topic, Larsen *et al.* [11] evaluated different algorithms for tree detection (namely maxima detection, valley following, region-growing, template matching, scale-space theory, and stochastic frameworks techniques) and performed experiments on six diverse forest images captured at different geographical locations. The authors concluded that no single algorithm can successfully analyze a scenario of various forest conditions (dense, complex, mixed, etc).

In fact, previous works on detecting citrus trees employed methods such as circular Hough transform [7], radial symmetry transform [4], [12], [13], and template matching [14], which produced good results for sparsely located data, but are not suitable for the images addressed in our research, showing orchards of dense distributed trees. When dealing with overlapped canopy forests, authors employ a variety of techniques; to name a few, thermal and narrowband multispectral imaging sensors [15], 3D photogrammetry and hyperspectral imaging [16], 3D reconstruction [17], and structure from motion (SfM) [18].

Nevertheless, although using specialized sensors may capture additional data, it brings limitations as well, regarding the costs of acquisition and processing: traditional photogrammetry analyses require high-cost cameras and platforms as well as rigorous processing chains for accurate results [17]. Besides cost, size and weight of existing multi-modal imaging systems may also inhibit large-scale deployment on-board UAVs [19] and make the solution unfeasible for small producers.

Alternatively, CNNs have been employed successfully for processing images on a wide range of remote sensing applications [20], and to a lesser extent, for detecting and delimiting trees. Focusing on image data from UAVs, Fan *et al.* [10] addressed tobacco detection at plantations disposed similarly to our citrus images. The authors first detect candidate tobacco plant regions using morphological operations and watershed segmentation, then, they use a CNN-based classifier for differentiating the candidates into tobacco or nontobacco plant regions, followed by a post-processing stage. In our work, the strength of the CNNs is explored not only for classification but also for candidate detection. Additionally, our post-processing relies on spatial characteristics of the considered plantation.

## III. Method

The proposed detection algorithm, depicted in Figure 2, comprises three main steps combining CNN classification, morphological image processing, and ancillary data with the nominal spacing of the trees during plantation:

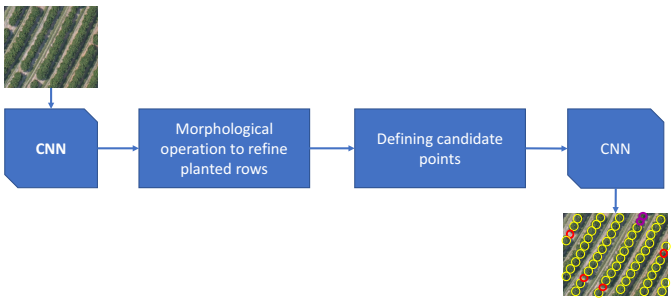



Fig. 2. Flowchart of the proposed method.

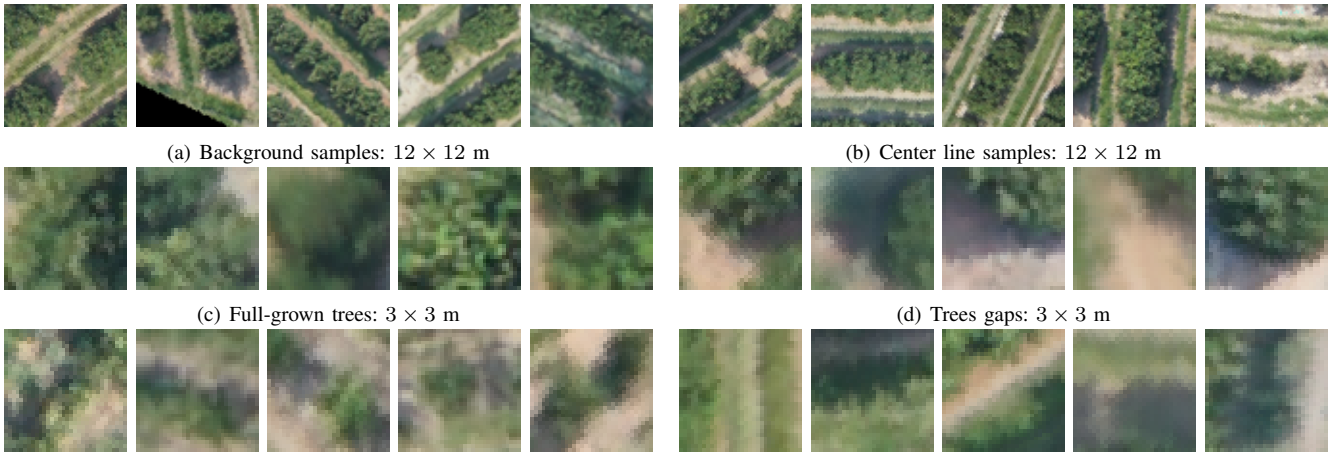
**Detection of planted rows using a CNN:** First, a CNN infers if the center of the planted tree row is centered in a certain (small) analyzed image patch. For this, the CNN is trained with examples of image patches centered in the rows versus the background areas (see examples in Figure 3). By applying this strategy to the whole input image, using sliding windows, we produce a probability image with high values in the regions near the center of the plantation rows (see an example in Figure 4). A mild Gaussian low-pass filter is used to reduce spatial noise of the resulting image.

**Extraction of center lines:** The central line of objects (segments) with high probability scores are retrieved using morphological thinning [21]. An area constraint is applied for lines shorter than 10 pixels to reduce spatial noise. Point locations spaced at the nominal distance between adjacent trees in the detected lines are retained for further processing (see the inner blue dots in the example shown in Figure 4).

**Candidate point classification using a CNN:** Image patches centered in each candidate point location are classified using the same CNN of the first step, except for the softmax layer; in this case, four categories have to be distinguished: full-grown tree, tree gaps, tree seedlings, and background areas.

The CNN architecture used for steps 1 and 3 is presented in the Table I. [1] The model consists of a sequence of layers such as convolution, max pooling, rectified linear units (ReLU), and fully connected neurons, before reaching the final softmax classification. These are frequent choices among CNN practitioners [22]. The difference between steps 1 and 3 is the softmax layer: step 1 trains the CNN with two categories (planted row, background); step 3 trains the CNN with four categories (full-grown tree, tree gaps, tree seedlings, and background areas). The parameters of each layer need to be optimized using labeled samples. In our implementation, max pooling was applied on $2 \times 2$ pixels with a stride 2 and zero padding. Convolutions used a stride of 1.

[1] In the experiments reported in Section IV, we use squared image patches of side $w = 64$ pixels for inferring the center of the plantation rows, and a smaller $w = 32$ pixels for classifying the candidates citrus tree locations.



(a) Background samples: $12 \times 12$ m (b) Center line samples: $12 \times 12$ m

(c) Full-grown trees: $3 \times 3$ m (d) Trees gaps: $3 \times 3$ m

(e) Trees seedlings: $3 \times 3$ m (f) Background: $3 \times 3$ m

Fig. 3. Example of image patches used to train the two CNN models for: (a)–(b) center row detection, (c)–(f) final classification.

TABLE I
ARCHITECTURE OF THE CNN USED ON STEPS 1 AND 3.

| Layer | Processing |
|---|---|
| 1 | Input image: $w \times w \times 3$ |
| 2 | 16 $5 \times 5$ Convolutions |
| 3 | ReLU |
| 4 | $2 \times 2$ Max Pooling |
| 5 | 32 $3 \times 3$ Convolutions |
| 6 | ReLU |
| 7 | $2 \times 2$ Max Pooling |
| 8 | 32 $3 \times 3$ Convolutions |
| 9 | ReLU |
| 10 | $2 \times 2$ Max Pooling |
| 11 | 64 Fully Connected |
| 12 | ReLU |
| 13 | 32 Fully Connected |
| 14 | ReLU |
| 15 | Fully Connected |
| 16 | Softmax |
| 17 | Classification output |

## IV. EXPERIMENTS

### A. Dataset

The considered RGB images for detecting full-grown citrus trees were acquired over a commercial orchard located in the southwest of Brazil. The area was mapped using aerial photographs captured by a GYRO-500X4 quadcopter (GyroFly™, São Jose dos Campos, Brazil) equipped with a Sony RX100 III camera (Sony™, Tokyo, Japan). The collected data was processed to generate orthomosaics with a pixel spacing of about 9.5 cm using Pix4D software (Pix4D S.A.™, Lausanne, Switzerland). The imaged area included 14 orchards with producing orange trees aged eight years of three varieties ("Natal", "Pera", and "Valencia Americana"), planted at a nominal spacing of $2.5 \times 6.8$ m. The orchards were randomly split in two disjoint sets, seven used for training and validation, and the remaining seven used for testing the proposed method. Considering the test data, orchards have an average of $5.783 \pm 1.475$ trees.

Adjacent trees overlap in the planted rows, which makes individual tree counting difficult. Variable illumination and shadow orientations, tree gaps, and the presence of tree seedlings increase the challenging factor of the task. Considering that the productivity analysis only take into account full-grown trees, we focus on classification experiments only of this class.

### B. Experimental set-up

We randomly sampled 4,000 image patches of size $128 \times 128$ pixels ($12 \times 12$ m on the ground) centered in manually drawn lines passing by the center of the plantation rows in each of the seven orchards allocated for training. Another 4,000 tiles were randomly sampled in background areas, here defined as locations further than 1 m away from the centers of the rows in the orchard. The resulting set of 56,000 image patches were aggregated and stored to train the first CNN model, i.e., the center row detector. We noted that down-sampling these images patches by a factor of two to train the CNN allowed keeping a good compromise between spatial context representation on the ground and model accuracy. While the working resolution in this step would not be enough to see scene details such as fruits in the trees, we found that image patches of size $64 \times 64$ sampled with a resolution of about 20 cm/pixel are still reasonable to capture neighborhood context needed to infer the location of the center of the plantation rows, as suggested by the visual inspection of Fig. 3. Furthermore, the coarser resolution allows flying at higher altitudes, increasing the areas mapped by UAVs.

Similarly, we randomly sampled image patches of size $32 \times 32$ pixels ($3 \times 3$ m on the ground) centered in the manually drawn centers of full-grown trees, tree gaps, trees seedlings, and background. 1,000 samples for each class and orchard in the training set were used to train the second CNN, designed to classify the candidate point detection obtained processing the first CNN. The number of tree gaps and trees seedlings was substantially smaller than those of full-grown

TABLE II
FULL-GROWN TREE DETECTION RESULTS FOR SEVEN TEST ORCHARDS USING THE PROPOSED METHOD.

| Evaluation score | Site 22 | Site 32 | Site 45 | Site 54 | Site 61 | Site 76 | Site 82 | All |
|---|---|---|---|---|---|---|---|---|
| Number of correctly detected full-grown trees (a) | 7055 | 3787 | 3473 | 6513 | 3927 | 5017 | 6355 | 36127 |
| Number of all detected full-grown trees (b) | 7087 | 3914 | 3594 | 6569 | 3956 | 5053 | 6431 | 36604 |
| Number of full-grown trees in the ground truth (c) | 7692 | 4345 | 4386 | 7140 | 4334 | 5607 | 6978 | 40482 |
| Precision: a/b (%) | 99.5 | 96.8 | 96.6 | 99.1 | 99.3 | 99.3 | 98.8 | 98.7 |
| Recall: a/c (%) | 91.7 | 87.2 | 79.2 | 91.2 | 90.6 | 89.5 | 91.1 | 89.2 |
| Overall accuracy (%) | 95.6 | 92.0 | 87.9 | 95.2 | 94.9 | 94.4 | 94.9 | 94.0 |

trees in the orchards. We sampled with replacement in case of fewer samples available, and randomly rotated the patches. Therefore, we tested the idea of using a larger spatial context to help identifying the center of the planted rows (i.e., the first CNN "sees" both the row center and part of the adjacent rows), and then we use a second CNN, trained to analyze the small local context around each inference point, to make the final full-grown tree detection.

The parameters of the CNN model were estimated using the stochastic gradient descent method with momentum [22], minimizing the cross-entropy loss function, on batches of 128 image patches at each iteration, with respect to the input parameters.

Shape files delineating the contour of the individual orchards were used to crop the orthomosaic and process each orchard.

### C. Results

For the purpose of the experiments, only full-grown trees will be considered in the accuracy assessment. Table II summarizes the scores obtained in the seven orchards reserved for testing purposes. These orchards are spatially disjoint to the training ones. To compute the accuracy scores, automated detection classified as full-grown tree was considered correct if its center where within 1.5 m of the center of a full-grown tree in the reference ground truth. The average precision above 98% suggest that all point locations automatically classified as tree are very likely to be correct. We also found that the proposed method systematically detects fewer full-grown trees than the reference available to us (average recall of 89.2%). This is likely to be due to higher uncertainty in the detection of tree gaps and eventual tree seedlings, and also the undesired discontinuities in the detection of the plantation rows.

Examples of detection are shown in Figure 4. The orange trees seen in the extract of orchard 82 are well developed and rather homogeneous, with few gaps. Conversely, the subset of orchard 45 is rather heterogeneous. Note that by analyzing a smaller spatial context, the second CNN helped to reject a false candidate line detected by the first CNN, that focused on a larger spatial context encompassing adjacent rows (see the red circles along the vertical line in the bottom right image in Figure 4, that were classified as background by the second CNN).

Given the related literature on tree detection, we believe that these results are satisfactory when considering image-only data with similar characteristics. Although Koc-san *et al.* [7] are able to achieve a positional accuracy ranging from 93%–100% on the detection of citrus trees, the three test sites considered by the authors comprised sparsely distributed trees, which probably help in the detection process. Despite dealing with a different plant type, Fan *et al.* [10] took high-density image inputs which were visually more similar to the data used in our research and achieved 93% accuracy on the detection of tobacco plants using CNN.

## V. DISCUSSION

Most "empty spaces in Figs. 4–5 were assigned to the alternative classes, i.e., the tree gaps, tree seedlings, or background and were successfully not classified as full-grown trees. Although we focus on the full-grown trees in this research, visual analysis suggests that increasing the number of training samples, especially for the tree gaps and tree seedling classes, could improve detection.

Except for the final layers, the structure of the CNN used in steps 1 and 3 (Table I) is kept identical, i.e., we focus on a rather standard CNN architecture and show that it performs reasonably well for the problem considered. We do not claim that the proposed CNN is the optimal choice for this task. Therefore, users wiling to devote additional time to optimize the CNN architecture, eventually using different networks for both classification steps, could further improve accuracy. The presented solution can be a baseline for further research on automatic citrus tree detection from UAVs images. Because of the identical CNNs architecture, once one CNN is trained, fine-tuning [22] could be used to help training the other CNN. This may also be useful when the number of labeled samples increases and retraining the classifiers is considered.

While we focused the analysis on full-grown trees, it remains clear from visual inspection of detections shown in Figs. 4–5 that additional improvements would be needed in case there is interest to detect also younger trees. In the specific orchard data available to us, younger trees appeared with lower frequency and we had difficulty to get reliable training samples. In this context, it is plausible that increasing the training samples availability could lead to improved detection.

Instead of grouping the final classification results into two classes, such as full-grown trees versus not full-grown trees,

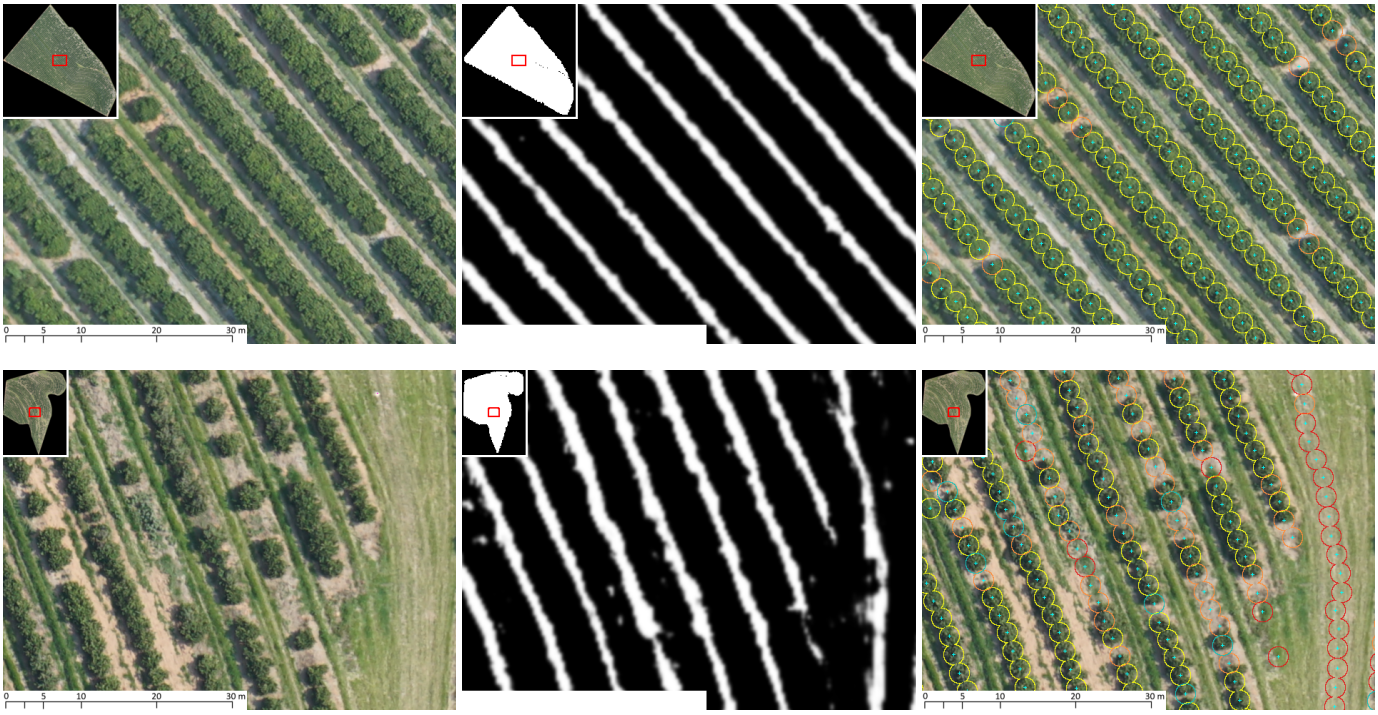


Fig. 4. Example of analysis at input orchards 82 (top row) and 45 (bottom row), each shown in the corresponding upper left inserts. The left column displays the input RGB images, while the middle column shows the segments classified with high probability of containing row centers. Finally, the right column depicts the tree centers in blue dots and yellow circles with diameter 2.5 m being centered at the point locations classified as full-grown trees. Also, it displays orange circles centered at the tree gaps, blue circles centered at tree seedlings, and red circles centered at candidate points classified as background by the CNN.

we disclose four output labels to provide additional label information that may help analysts to filter detection during visual inspection of the results. The four classes could also be used to decide on further processing.

Further research could analyze the network architecture in more detail and compare the proposed two-stage procedure with the planting row detection and tree detection with a single step CNN solution. This would allow understanding if the learning process is faster or easier with two stages than a single CNN model. Furthermore, it would help understanding whether using handcrafted intermediate processing steps, inspired by human interpretation to guide the detection flow is beneficial, and the impact in terms of demand of labeled training samples to build the detection models.

## VI. Conclusion

This paper presented a novel method for detecting citrus trees at high-density orchards, taking into consideration the spatial arrangement of the plantation and the nominal tree planting space. Although techniques exploring 3D information such as oblique imaging and digital surface models could be valuable for our crown overlapping scenario, we chose to disregard specialized sensors and use image-only information. The goal of imposing such limitation is to access whether we are able to achieve a satisfactory solution that is also low-cost and scalable, allowing deployment on several UAV models. Our approach is based on CNNs and provided good results on seven test sites with distinct alignments.

In future work, we intend to fine-tune the considered parameters, compare our CNN-based solution with other methods – that may or may not employ neural networks –, and perform experiments on different datasets. As mentioned in Section II, previous work demonstrated that a single algorithm is not able to work successfully for plantations with very distinct characteristics. Nevertheless, despite focusing on citrus trees, the method has potential to work for different plantations that are arranged similarly.

## Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions for improving the manuscript.

## References

[1] DroneDeploy, “2018 commercial drone industry trends report,” https://www.dronedeploy.com/resources/ebooks/2018-commercial-drone-industry-trends-report/, Accessed: 2018-06.

[2] United States Department of Agriculture (USDA), “Citrus: World markets and trade,” https://www.fas.usda.gov/data/citrus-world-markets-and-trade, Accessed: 2018-06.

[3] Y. Seul Lim, H. La, J. Soo Park, M. Hee Lee, M. Wook Pyeon, and J.-I. Kim, “Calculation of tree height and canopy crown from drone images using segmentation,” vol. 33, pp. 605–613, 12 2015.

[4] A. Ozdarici-Ok, “Automatic detection and delineation of citrus trees from vhr satellite imagery,” *International Journal of Remote Sensing*, vol. 36, no. 17, pp. 4275–4296, 2015.

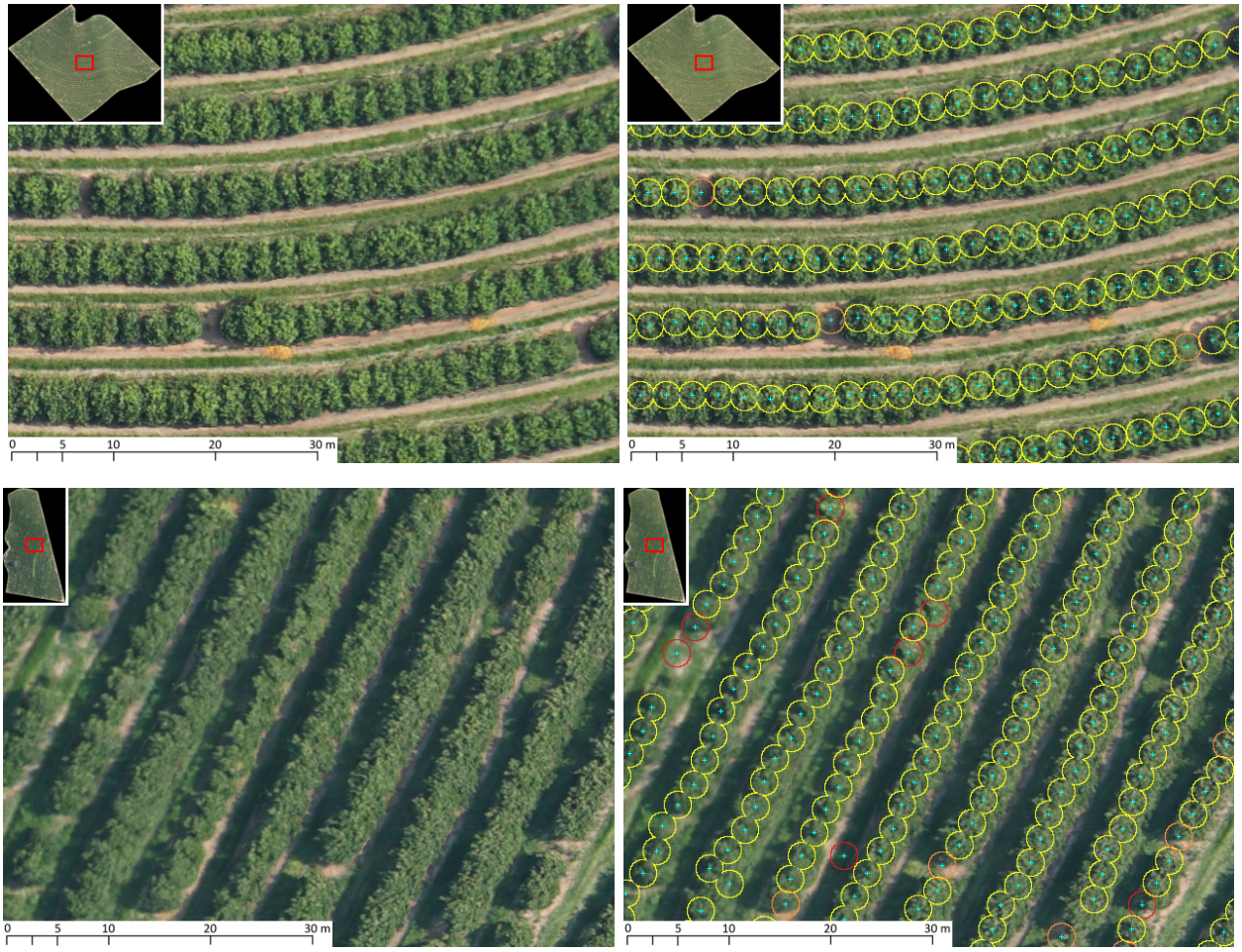



Fig. 5. Example of analysis of the orchards 22 (top row) and 32 (bottom row), each shown in the corresponding upper left inserts. The left column displays the input RGB images and the right column depicts the tree centers in blue dots and yellow circles with diameter 2.5 m being centered at the point locations classified as full-grown trees. Also, it displays orange circles centered at the tree gaps, blue circles centered at tree seedlings, and red circles centered at candidate points classified as background by the CNN.


[5] D. Li, H. Guo, C. Wang, W. Li, H. Chen, and Z. Zuo, "Individual tree delineation in windbreaks using airborne-laser-scanning data and unmanned aerial vehicle stereo images," *IEEE Geoscience and Remote Sensing Letters*, vol. 13, no. 9, pp. 1330–1334, Sept 2016.

[6] J. Secord and A. Zakhor, "Tree detection in urban regions using aerial lidar and image data," *IEEE Geoscience and Remote Sensing Letters*, vol. 4, no. 2, pp. 196–200, April 2007.

[7] D. Koc-San, S. Selim, N. Aslan, and B. T. San, "Automatic citrus tree extraction from UAV images and digital surface models using circular hough transform," *Computers and Electronics in Agriculture*, vol. 150, pp. 289 – 301, 2018.

[8] Y. Bazi, H. Al-Sharari, and F. Melgani, "An automatic method for counting olive trees in very high spatial remote sensing images," in *2009 IEEE International Geoscience and Remote Sensing Symposium*, vol. 2, July 2009, pp. II–125–II–128.

[9] E. K. Cheang, T. K. Cheang, and Y. H. Tay, "Using convolutional neural networks to count palm trees in satellite images," *CoRR*, vol. abs/1701.06462, 2017. [Online]. Available: http://arxiv.org/abs/1701.06462

[10] Z. Fan, J. Lu, M. Gong, H. Xie, and E. D. Goodman, "Automatic tobacco plant detection in UAV images via deep neural networks," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 11, no. 3, pp. 876–887, March 2018.

[11] M. Larsen, M. Eriksson, X. Descombes, G. Perrin, T. Brandtberg, and F. A. Gougeon, "Comparison of six individual tree crown detection algorithms evaluated under varying forest conditions," *International Journal of Remote Sensing*, vol. 32, no. 20, pp. 5827–5852, 2011.

[12] A. O. Ok and A. Ozdarici-Ok, "Detection of citrus trees from UAV DSMS," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. IV-1/W1, pp. 27–34, 2017. [Online]. Available: https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/IV-1-W1/27/2017/

[13] A. . Ok and A. . Ok, "Automated detection of citrus trees from a digital surface model," in *2017 25th Signal Processing and Communications Applications Conference (SIU)*, May 2017, pp. 1–4.

[14] P. Maillard and M. F. Gomes, "Detection and counting of orchard trees from vhr images using a geometrical-optical model and marked template matching," *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. III-7, pp. 75–82, 2016. [Online]. Available: https://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/III-7/75/2016/

[15] J. A. J. Berni, P. J. Zarco-Tejada, L. Suarez, and E. Fereres, "Thermal and narrowband multispectral remote sensing for vegetation monitoring from an unmanned aerial vehicle," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 3, pp. 722–738, March 2009.

[16] O. Nevalainen, E. Honkavaara, S. Tuominen, N. Viljanen, T. Hakala, X. Yu, J. Hyyppä, H. Saari, I. Pölönen, N. N. Imai, and A. M. G. Tommaselli, "Individual tree detection and classification with UAV-based photogrammetric point clouds and hyperspectral imaging," *Remote Sensing*, vol. 9, p. 185, 2017.

[17] R. A. Daz-Varela, R. de la Rosa, L. Len, and P. J. Zarco-Tejada, "High-resolution airborne UAV imagery to assess olive tree crown

parameters using 3d photo reconstruction: Application in breeding trials," *Remote Sensing*, vol. 7, no. 4, pp. 4213–4232, 2015. [Online]. Available: http://www.mdpi.com/2072-4292/7/4/4213

[18] A. J. Mathews and J. L. R. Jensen, "Visualizing and quantifying vineyard canopy lai using an unmanned aerial vehicle (uav) collected high density structure from motion point cloud," *Remote Sensing*, vol. 5, no. 5, pp. 2164–2183, 2013. [Online]. Available: http://www.mdpi.com/2072-4292/5/5/2164

[19] J. Das, G. Cross, C. Qu, A. Makineni, P. Tokekar, Y. Mulgaonkar, and V. Kumar, "Devices, systems, and methods for automated monitoring enabling precision agriculture," in *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, Aug 2015, pp. 462–469.

[20] R. Baeta, K. Nogueira, D. Menotti, and J. A. dos Santos, "Learning deep features on multiple scales for coffee crop recognition," in *2017 30th SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, Oct 2017, pp. 262–268.

[21] L. Lam, S.-W. Lee, and C. Y. Suen, "Thinning methodologies-a comprehensive survey," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 14, no. 9, pp. 869–885, 1992.

[22] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.