FCRN-BASED MULTI-TASK LEARNING FOR AUTOMATIC CITRUS TREE DETECTION FROM UAV IMAGES

L. E. C. La Rosa[1,3,*], M. Zortea[1], B. H. Gemignani[2], D. A. B. Oliveira[1], R. Q. Feitosa[3]

[1] IBM Research Av. Paster, 146, Rio de Janeiro, Brazil, 22290-240 - lauracue.rosa@ibm.com, (mazortea,dariobo)@br.ibm.com
[2] 3DGEO - bruno@3dgeo.com.br
[3] Dept. of Electrical Engineering, Pontifical Catholic University of Rio de Janeiro, Brazil - (lauracue,raul)@ele.puc-rio.br

KEY WORDS: Citrus Trees, Fully Convolutional Regression Network, Multi-Task Learning, Unmanned Aerial Vehicle.

ABSTRACT:

Citrus producers need to monitor orchards frequently, and would benefit greatly from having automated tools to analyze aerial images acquired by drones over the plantations. However, analysing large aerial data sets to enable producers to take management decisions that would optimize productivity and sustainability over time and space remains challenging. Motivated by the success of deep learning in computer vision, this work proposes a novel approach based on Fully Convolutional Regression Networks and Multi-Task Learning to detect individual full-grown trees, tree seedlings, and tree gaps in citrus orchards for inventory tracking. We show that the proposal can identify eight-year-old orange trees with accuracy between 95–99% in high-density commercial plantations where adjacent crowns overlap. This quality of detection was achieved on RGB orthomosaics with a pixel size of about 9.5 cm and requires the nominal spacing between adjacent trees as a priori information. Our results also highlight that detecting tree seedlings and tree gaps remains a challenge. For these two categories, classification sensitivity (recall) was between 59–100% and 63–94%, respectively.

1. INTRODUCTION

Citrus growers need to monitor orchards to keep up-to-date records of the number of bearing trees, to inspect how seedlings develop, and to detect potential anomalies in the plantation that may influence productivity. Traditional orchard monitoring relying on plot sampling and manual inspection of trees in situ is a laborious task and becomes challenging for large commercial plantations. Orchard monitoring using remote sensing is a promising alternative to complement traditional field inspections. In this context, advances in unmanned aerial vehicles (UAV or "drones") technology have opened the possibility of on-demand image acquisition, allowing farmers to monitor crops frequently. Drone operators can plan flights that carry sensing equipment, such as standard digital RGB cameras, multispectral, and hyperspectral sensors delivering images with a centimeter-level resolution on the ground, sufficient to see subtle details in the field that would be difficult to resolve using, for instance, satellite remote sensing. In agriculture, drones are present in various applications such as field mapping, weed management, plant stress detection, inventory counting, biomass estimation, and chemical spraying (Hassler, Baysal-Gurel, 2019).

The automatic tree-based inventory is an active research topic in the last decades and has been investigated, for example, to extract tree stand characteristics (e.g., mean tree diameter and height) needed in commercial forest management planning (Kaartinen et al., 2012). Many methods have been used to detect individual trees, using imagery from a variety of optical and Light Detection and Ranging (LiDAR) sensors. Ke and Quackenbush (Ke, Quackenbush, 2011) review various techniques developed for automatic detection of individual trees in images, including local maxima filtering, image binarization, scale analysis, and template matching, among others. Traditional image processing and pattern recognition methods inspired most of these algorithms. In some applications, individual tree detection using such approaches can be further processed using machine learning classifiers to produce the final identification and assign confidence to each tree detection (Zortea et al., 2017). Using texture descriptors also helps classification (Kobayashi et al., 2019). However, a challenge remains when adjacent tree crowns overlap.

Deep learning (LeCun et al., 2015) is improving computers' ability to analyze and learn patterns from large data sets. It excels at computer vision tasks such as object detection and semantic segmentation. Deep learning are models composed of a sequence of filters organized in the form of multiple layers of processing. When applied to images, deep learning approaches such as convolutional neural networks (CNNs) extract attributes and discriminate the classes of interest defined by the user. In a supervised setting, the weights of such filters are tuned iteratively using a set of training samples and the backpropagation algorithm. Deep learning methods are being used in agriculture (Kamilaris, Prenafeta-Boldú, 2018) and studies have show that, in principle, detecting and counting individual (oil-palm) trees in commercial plantations is more accurate with CNNs than neural networks, template matching, and local maximum filter (Li et al., 2017). For instance, Li and colleagues (Li et al., 2017) used a CNN running in sliding windows to detect and count oil-palm trees in high spatial resolution multispectral satellite images. It improved results compared to more traditional algorithms. Another sliding window approach was proposed in (Zortea et al., 2018b) that combined the detection probabilities, estimated by a pair of binary CNNs classifiers, trained in image patches of different spatial resolutions, to detect oil-palm trees in RGB orthomosaics from aerial photographs acquired by drones. The idea was to improve robustness for a possible poor estimation of the values of the CNN filters.

*Corresponding author

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-3-W2-2020-65-2020 | © Authors 2020. CC BY 4.0 License.
Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165654

As recent examples of orchard analysis, Csillik and colleagues (Csillik et al., 2018) detected citrus and other crop trees from UAV images using a CNN, followed by a classification refinement using superpixels derived from a Simple Linear Iterative Clustering (Achanta et al., 2010) algorithm and filtering of local probability maxima. The authors used multispectral orthomosaics containing the red, green and two infra-red bands with a resolution 12 cm/pixel and achieved a precision of 94.6% and recall of 97.9%. The image examples shown in their work suggest that the crown of adjacent trees did not overlap. (Zortea et al., 2018a) detected individual orange trees in UAV images using two CNNs. The first was trained to detect the plantation rows. The rows were refined using mathematical morphology. Then, a second CNN, focusing in closer vicinity to the plantation rows made the final classification of individual full-grown trees, tree seedling, and tree gaps. Experiments on eight-year-old orange orchards revealed an average precision of 98.7% and recall of 89.2% in the detection of full-grown trees using RGB orthomosaics with 10 cm/pixel.

In this work, we present a novel end-to-end architecture that tackles individual citrus inventory considering a single multi-task learning architecture based on fully convolutional regression networks to estimate a density map. Our proposal enables handling location and classification altogether, improving precision and recall in comparison to other tree inventory approaches. We demonstrate effective detection of the location of mature orange trees (living, bearing or non-bearing trees), tree seedlings, and plantation gaps in orchards using RGB orthomosacis obtained by drones as input.

2. FUNDAMENTALS

In this section, we present the theoretical fundamentals that support our method: fully convolutional regression networks, counting by density maps and multi-task learning.

Fully Convolutional Regression Networks: Recently, Long et al. (2015) introduced fully convolutional networks (FCN) for semantic labeling problems (Long et al., 2015). These networks are trained to predict jointly all labels in an input image little loss in spatial resolution and in the past few years have been used with great success in the remote sensing community (Volpi, Tuia, 2016). In (Long et al., 2015), the authors reinterpret the fully connected layers of a traditional CNN as a convolutional layer. After many downsampling stages (typically convolution followed by pooling layers) the method employs an upsampling strategy (bilinear, deconvolution) in order to recover the original input image size, and predict the labels at pixel-level. The network is trained end-to-end and is able to learn spatial, intra- and inter-class relationship across the input image. In this context, FCNs are well-know to perform structured prediction by combining context with spatial information. Hence, a Fully Convolutional Regression Network (FCRN) (Xie et al., 2018) consists of a FCN applied in a regression problem instead of a semantic labeling task. The network learns a mapping between an input image I and a density map d. Regression-based methods have been wildly used in counting and detection applications such as counting bacteria or cells in microscopic images (Xie et al., 2018), dense crowd (Zhang et al., 2016), among others.

Counting by density maps: Introduced by (Lempitsky, Zisserman, 2010), object counting using density maps avoids the difficulties of explicit detection and segmentation of all objects instances, and at the same time takes into account spatial relationships, representing the state-of-art when objects are heavily overlapped (Sindagi, Patel, 2018, Jiang et al., 2019). In (Lempitsky, Zisserman, 2010), density estimation is tackled trough a supervised learning algorithm, where the parameters of the classifier are learned by minimizing the error between the true and the pixel-wise density prediction. The final density map is obtained by running the predictor as a sliding window over the whole image, then, a post-processing step is commonly applied to count the objects. The integral of the density map, non-maximum suppression, k-means clustering, and gaussian mixture models have been successfully used to predict the number of objects from density maps (Ma et al., 2015). Recent works use FCN to perform patch-wise density prediction to speed up the training and inference time (Kang et al., 2018).

Multi-Task learning: The core idea behind multitask learning (MTL) is to train a model capable of solving a number of related tasks simultaneously in the same processing structure. In the context of fully convolutional networks, one of the most typical approach for multi-task learning consist on a shared subnet or encoder, followed by task-specific subnets or decoders. Such models tends to better generalization, since each different task targets specific outcomes, through specific losses. Alongside, MTL reduces the number of labelled samples required per task to attain a good performance (Shui et al., 2019) and helps to focus its attention on the set of features maps that actually are relevant since the task all-together provide information about the importance of those features (Ruder, 2017). Typically, in MTL there are more than one loss function to be optimized during training.

3. METHODOLOGY

Our method is composed of two main steps: (1) training a FCRN in a multi-task learning setting to infer density maps centered at point locations of full-grown orange trees, tree seedlings, and tree gaps; and (2) post-processing and final classification. In the post processing step we first estimate the center of the plantation rows by applying morphological operations to the density maps; then we generate candidate point locations evenly spaced along the predicted plantation rows, using the nominal spacing between adjacent trees known a priori; and finally we classify the final candidate points. The flow chart of our proposal is shown schematically in Fig. 1 and detailed below.

3.1 Fully Convolutional Regression Network Multi-Task learning (FCRN-MTL)

The design of our method is based on hypotheses raised from data observations:

1. Adjacent crowns of full-grown citrus trees may overlap substantially in commercial plantations. Density maps, with pixel values proportional to the distance to closest plantation point, present higher values along the plantation rows.
2. The symmetric shape of tree crowns and the fact that orchard trees are planted in (straight) rows on a regular

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-3-W2-2020-65-2020 | © Authors 2020. CC BY 4.0 License.
Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165654

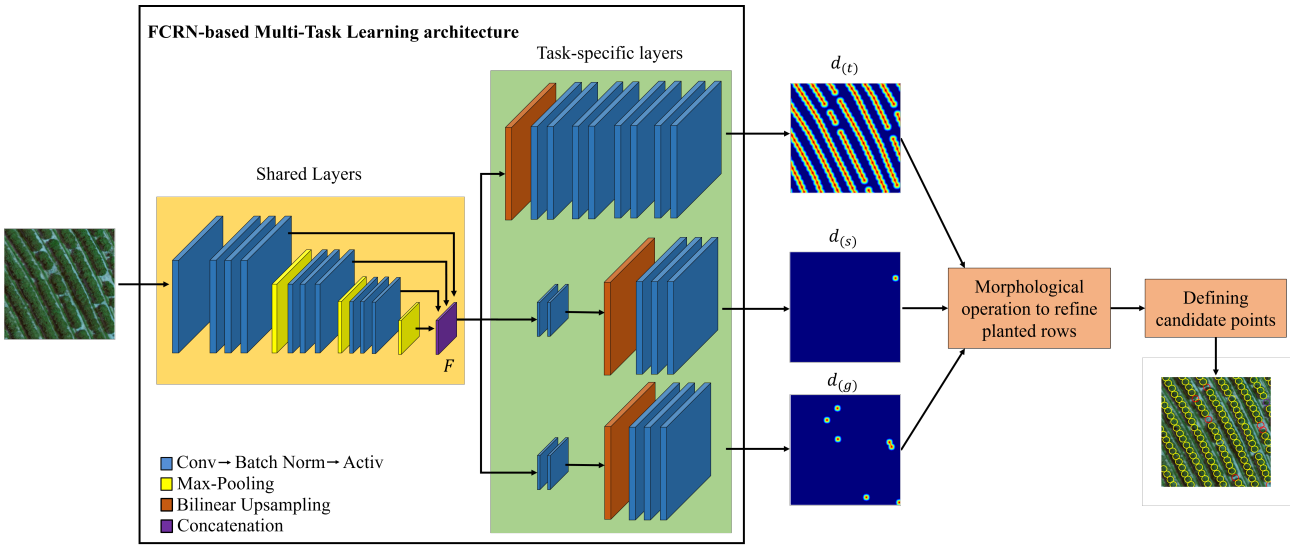

Figure 1. Flowchart of the proposed method. d_t, d_s, d_g are the density maps for mature orange trees, tree seedlings, and plantation gaps, respectively.

planting pattern can help to separate trees from other background areas in the image due to its spatial context.

3. Due to the sparsity of tree seedling location and gaps, their resemblance to grass and exposed soil located outside the plantation rows, training an independent model for each class can lead to erroneous detection as reported by (Zortea et al., 2018a). Hence, the use of multi-task learning can brings extra spatial context from the presence of full-grown trees to help detect tree seedlings and tree gaps within the plantation rows.

As in (Xie et al., 2018), we use a FCRN to learn a mapping between an image and a density map. However, different from the authors, our network is implemented to perform multi-task learning. Given an input image $I \in R^{m \times n}$ the network learns a function to map from the input space to three different density maps $d_t, d_s, d_g \in R^{m \times n}$, one for full-grown orange trees, one for tree seedlings, and tree gaps, respectively (see Fig. 1). From a set of H training images $I_1, I_2, ..., I_h$, we assumed that each images I_i is annotated with a set of 2D points P_i, were each $p \in P_i$ correspond to one of the classes of interest (i.e., tree, seedlings and gaps). We obtain the ground truth density map for point p by convolving a circular filter with center at the point location and calculating the Euclidean Distance Transform (EDT) (see examples in Fig. 1). This distance map has the same dimensions of the input image and each pixel contains the euclidean distance to the closest background or zero value pixel. All distances were normalized between 0 and 255, and the resultant map was saved as an image. Note that point annotation requires less human effort than traditional bounding-box annotations, which make our method attractive for counting on large-scale remote sensing data.

The FCN comprises an encoder (the orange block of Fig. 1) and different decoders depending on the target (the green block of Fig. 1). We implement the encoder using three convolutional blocks (CBs), where each CB is composed of successive convolution $\rightarrow$ batch normalization $\rightarrow$ activation, followed by a max-pooling layer of 2×2. These operations reduce the resolution of the feature space by 2^3. After the last max-pooling layer, the network includes short-cut connection (three arrows connected to the purple box in Fig. 1). The short-cut connections concatenate the output feature maps (before each max-pool) from previous CBs to the feature maps produced by the last max-pooling layer. We applied extra max-pool operations to the feature maps from previous CBs to match the resolution of the final CB. We can interpret the concatenation of features from different levels of convolution as multi-scale feature fusion. From then on, we feed the coarse feature representation F into three different decoders to reconstruct the density maps. Our method perform MTL in such a way that the encoder layers are shared layers and the decoder ones are task-specific layers (orange and green blocks in Fig. 1). For the decoder correspondent to full-grown trees, we resize F to be equal to the input size using bilinear interpolation. Using bilinear interpolation introduces no extra checkerboard artifacts, which are present in other upsampling methods such as deconvolution (Odena et al., 2016). Then, we pass the resized F to four CBs of two convolutions each one to reconstruct the density map. For the decoders of tree seedlings and tree gaps (middle and bottom of green block in Fig. 1), we first process F by one CB and then we resize the activation map to the input size using bilinear interpolation. Finally, we use one more CB to generate per-pixel density values.

Since the density map of our model can also be interpreted as a depth map with higher values at the center of the point annotations, we use the scale-invariant error as a training loss which encourages neighboring pixels to have a similar depth/density values (Eigen et al., 2014). For a predicted density map d and the ground truth $\hat{d}$, each with q pixels, the scale-invariant mean squared error is defined as:

$$L(d, \hat{d}) = \frac{1}{q}\sum_{i=1}^{y}(y_i)^2 - \frac{\lambda}{q^2}(\sum_{i=1}^{y} y_i)^2 \quad (1)$$

where $y_i = \log d_i - \log \hat{d}_i$ and $\lambda \in [0, 1]$. As in (Eigen et al., 2014), we use $\lambda = 0.5$, that can be seen as an average of element-wise l_2 ($\lambda = 0$) and the exactly scale-invariant error ($\lambda = 1$). The final loss function is the sum of the loss for each of the density maps:

$$L = w_1 L_t(d_t, \hat{d}_t) + w_2 L_s(d_s, \hat{d}_s) + w_3 L_g(d_g, \hat{d}_g) \quad (2)$$

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-3-W2-2020-65-2020 | © Authors 2020. CC BY 4.0 License.
Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165654

where w_1, w_2, w_3 are constants used to weight the contribution of each density map.

After training, given a novel image, we predict the density values using a sliding window over the image with an overlap of 0.3, keeping the prediction of the central region of each patch. This way, we minimize eventual border effects near adjacent images patches.

3.2 Post-processing and final classification

Unlike previous works, our solution does not require the use of the integral of the density map neither non-maximum suppression for detection and counting. Our customized post-processing algorithm comprises two processing blocks. First, we binarize each density map given certain thresholds, obtaining b_t, b_s and b_g. Next, we create a new binary map $\mathcal{M} = b_t + b_s + b_g$. Then, each pixel $p \in \mathcal{M}$ is set to one if $p \geq 1$ and zero otherwise. Hence, we have a global binary map of tree lines, including seedling and tree gaps. Finally, we apply a Gaussian filter to smooth the map $\mathcal{M}$ and get the central line of objects (segments) using skeletonization (Zhang, Suen, 1984).

In the second block, we iterate over each skeleton line at points spaced by the known nominal distance between adjacent trees in the orchard. For each candidate point i, j, we first check if $b_s(i, j) = 1$, if not, we check if $b_g(i, j) = 1$, and finally, if the point is not in the tree seedling neither to tree gap masks, we check whether $b_t(i, j) = 1$. We follow this order of priority since a point can have a value equal to one in more than one binary map. Hence, we prioritize labeling tree seedling and tree gaps before full-grown trees.

4. EXPERIMENTS

4.1 Dataset

The data used for the experiments is from a commercial orange tree plantation in Santa Cruz do Rio Pardo, São Paulo, southwestern Brazil. There, a Batmap® (Nuvem UAV®, Presidente Prudente-SP, Brazil) UAV, equipped with a 24MP Sony A6000 RGB camera (Sony®, Tokyo, Japan), took aerial photographs of orchards that were later processed to generate an orthomosaic with a ground sample distance of about 9.5cm/pixel, using the Pix4D® software (Pix4D S.A., Prilly, Switzerland). To facilitate data handling, we croped individual orchards in the georeferenced images using shapefiles delineating the borders of the farms. Images from seven orchards were randomly selected and reserved for testing (Table 2). An independent set of orchards, imaged nearby, was split for training and validation purposes. The nominal planting space between adjacent trees and rows was 2.5×6.8 m in the orchards reserved for testing. These orchards are spatially disjoint to the training ones. Crowns of adjacent mature trees, aged about eight years old, overlapped, adding to the challenge of individual tree detection.

4.2 Experimental set-up

We built training and validation sets containing 22,904 and 383 image patches, respectively. We cropped training patches with a 75% side overlap to increase the number of samples. The validation patches did not overlap. We considered patches of size 128, 256 and 512 pixels and, after preliminary experiments, selected 512×512 because patches of this size display more contextual information of plantation patterns. As mentioned earlier, for each image patch, we used ground truth with the coordinates of the center of the crowns of full-grown trees, three seedlings, and tree gaps (missing trees in the rows) to generate density maps to train the proposed model. Both, images patches and reference density maps were normalized between zero and one.

	Encoder			**Mature Tree Decoder**	
	Layer	Processing		Layer	Processing
	Input RGB	512×512×3		Input F	64×64×60
	conv1	24 3×3		Resize	8×8
CB1	conv2	20 3×3	CB4	conv11	28 3×3
	conv3	25 3×3		conv12	24 3×3
	conv4	20 3×3			
	pool1	2×2			
CB2	conv5	20 3×3	CB5	conv13	20 3×3
	conv6	25 3×3		conv14	20 3×3
	conv7	20 3×3			
	pool2	2×2			
CB3	conv8	20 3×3	CB6	conv14	20 3×3
	conv9	25 3×3		conv15	16 3×3
	conv10	20 3×3			
	pool3	2×2			
	short-cut connection		CB 7	conv16	16 3×3
	F = concat(pool1,pool2,pool3)			conv17	1 3×3
	Seedling Decoder			**Tree Gaps Decoder**	
	Layer	Processing		Layer	Processing
	Input F	64×64×60		Input F	64×64×60
CB8	conv18	28 3×3	CB10	conv23	28 3×3
	conv19	16 3×3		conv24	16 3×3
	Resize	8×8		Resize	8×8
CB9	conv20	16 3×3	CB11	conv25	16 3×3
	conv21	8 3×3		conv26	8 3×3
	conv22	1 3×3		conv27	1 3×3

Table 1. Architecture of the network.

The FCRN-MTL architecture is shown in Table 1. The input image patch is a 512×512 RGB image. As described on Section 3, the encoder consists in three CB (CB1–CB3) where each convolutional layer is followed by a batch normalization and by a exponential linear unit (ELU). All the convolutional filters are 3×3. The short-cut connection was implemented by concatenating the activation maps from pool1 downsampled by 4, from pool2 downsampled by 2, and pool3. From them on, the decoder use bilinear interpolation to recover the input spatial dimension and CBs (CB4–CB11) to process the coarse representation and output the density maps, i.e., the responses from conv17, conv22 and conv27. The convolution kernels were initialized with a Glorot uniform initializer (Glorot, Bengio, 2010). We used the ADAM (Kingma, Ba, 2014) optimizer with learning $1e^{-3}$, parameter values $\beta 1$ = 0.9 and $\beta 2$ = 0.999.

The batch size is set to 5 and, since the dataset is highly unbalanced, with seedling being the class with less samples, followed by tree gaps and full-grown trees, we set a higher weight to the loss function L_s in Equation 2. We obtained the best results with $w_1 = 0.2$, $w_2 = 0.5$ and $w_3 = 0.3$. As in (Zortea et al., 2018a), candidate points, classified as one of the three class of interest, were considered correct if its center where within 1.5 m of the center of the corresponding class at the reference ground truth.

4.3 Results and discussion

Table 2 summarizes the detection scores for the seven orchards reserved for testing. The location and the number of full-grown trees were detected with precision and recall above 95% in all

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-3-W2-2020-65-2020 | © Authors 2020. CC BY 4.0 License.
Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165654

Table 2. 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
correctly detected full-grown trees (a)	7641	4171	4169	7071	4273	5523	6972	39820
all detected full-grown trees (b)	7668	4232	4230	7099	4301	5558	7060	40148
full-grown trees in the ground truth (c)	7692	4345	4386	7140	4334	5607	6978	40482
Precision: a/b (%)	99.6	98.6	98.6	99.6	99.3	99.4	98.8	**99.2**
Recall: a/c (%)	99.3	96.0	95.1	99.0	98.6	98.5	99.9	**98.4**
Overall accuracy (%)	99.5	97.3	96.8	99.3	99.3	98.9	99.3	**98.8**
correctly detected tree seedlings (a)	0	0	131	28	37	66	46	308
all detected tree seedlings (b)	9	0	325	57	69	116	68	644
tree seedlings in the ground truth (c)	0	0	189	47	50	109	76	471
Precision: a/b (%)	0.0	100	40.3	49.1	53.6	56.9	67.6	**47.8**
Recall: a/c (%)	0.0	100	69.3	59.6	74.0	60.6	60.5	**65.4**
Overall accuracy (%)	0.0	100	54.8	54.3	63.8	58.7	64.2	**56.6**
correctly detected tree gaps (a)	301	216	855	354	262	261	676	2925
all detected tree gaps (b)	426	308	1038	413	364	444	769	3762
tree gaps in the ground truth (c)	318	343	1115	402	338	317	779	3612
Precision: a/b (%)	70.7	70.1	82.4	85.7	72.0	58.8	87.9	**77.8**
Recall: a/c (%)	94.7	63.0	76.7	88.1	77.5	82.3	86.8	**81.0**
Overall accuracy (%)	82.7	66.6	79.5	86.9	74.7	70.6	87.3	**79.4**

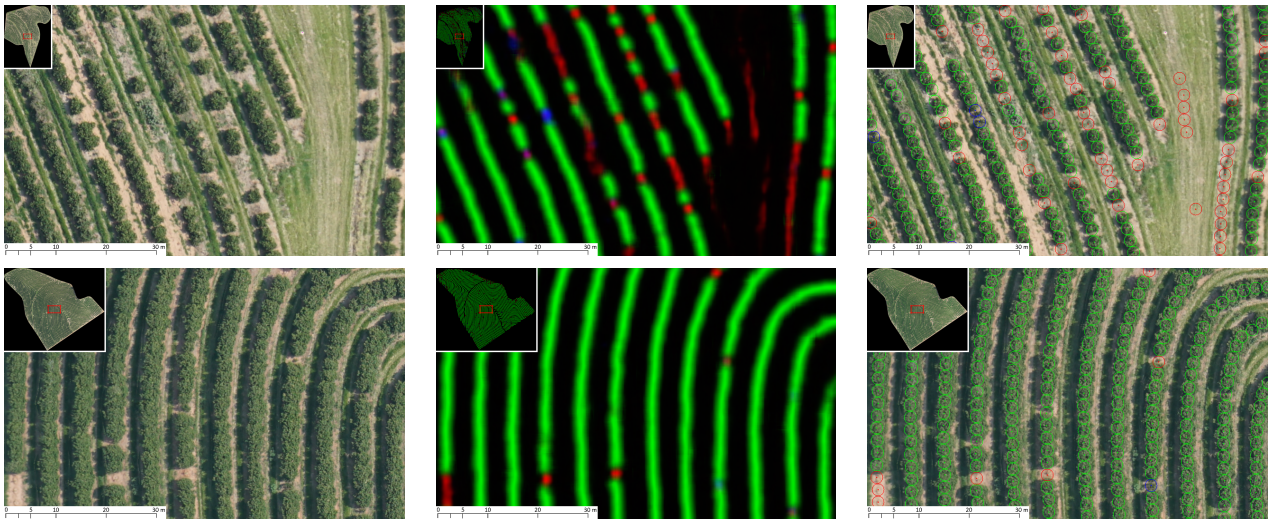

Figure 2. Example of results for orchards 45 (top row) and 54 (bottom). For visualization purposes, the density maps obtained by the proposed method are shown in color compositions using the RGB channels, placing the densities of full-grown trees (at green channel), tree gaps (red channel), and tree seedlings (blue channel). High density values are consistent with the location of plantation rows, and black color represent low density values associated with background areas. The circles overlaid in the right-most images have a diameter of 2.5 m (such as the nominal planting distance) and are centered at points retained after the post-processing. The color of the circle is according to the final classification obtained by the proposed method.

sites, with an average reaching 98.8%. This corresponds to a 4.8% increase in the average score reported in (Zortea et al., 2018a) using a sliding window-based CNN approach (94.0%) for the same orchards. Detection of tree seedlings and tree gaps proved challenging, with overall accuracy of 56.6% and 79.4%, respectively. Often, both classes got confused because pixels in the small crowns of tree seedling were mixed with the exposed soil from the background. For these two categories, the recall was between 59–100% and 63–94%, respectively. It is worth pointing out that the performance of the method strongly depends on the weight assigned to each task's loss. Tuning these hyperparameters is an expensive process which exceeds the scope of this study.

Fig. 2 shows a detail of orchard 45, characterized by many tree gaps and background vegetation in the plantation rows. In this challenging scenario, most full-grown trees have been detected. The algorithm confused an area that appears to be a tractors pathway (to the right of the image) with tree gaps. The method erroneously interpreted this pathway as a line tree, however other regions with grass or soil outside the tree lines were correctly identified as background. Conversely, orchard 54 in Fig. 2 had fewer gaps and detection produced by the algorithm was correctly placed along the plantation rows. In all cases, prior information on the adjacent tree spacing facilitated the analysis.

5. CONCLUSIONS

We presented a method to help in orchard inventory using color aerial images acquired by drones. The key aspect of our proposal is a novel detection approach that combines FCRNs with multi-task learning. This proved very accurate in detecting full-grown orange trees. Future research efforts should prioritize detection of tree seedlings and tree gaps,

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-3-W2-2020-65-2020 | © Authors 2020. CC BY 4.0 License.
Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165654

which remains a challenge in the commercial orchards considered in this study, where adjacent trees are spaced 2.5 m apart and adjacent tree crowns overlap. Studying how the proposed method compares to an alternative approach that trains a FCRN for each class, how the method generalize to other areas, plantation patterns, and fruit types is a future venue. Our learning strategy enables solving other similar problems in object identification.

REFERENCES

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., Süsstrunk, S., 2010. Slic superpixels. Technical report.

Csillik, O., Cherbini, J., Johnson, R., Lyons, A., Kelly, M., 2018. Identification of citrus trees from unmanned aerial vehicle imagery using convolutional neural networks. *Drones*, 2(4), 39.

Eigen, D., Puhrsch, C., Fergus, R., 2014. Depth map prediction from a single image using a multi-scale deep network. *Advances in Neural Information Processing Systems*, 2366–2374.

Glorot, X., Bengio, Y., 2010. Understanding the difficulty of training deep feedforward neural networks. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 249–256.

Hassler, S. C., Baysal-Gurel, F., 2019. Unmanned Aircraft System (UAS) Technology and Applications in Agriculture. *Agronomy*, 9(10), 618.

Jiang, X., Xiao, Z., Zhang, B., Zhen, X., Cao, X., Doermann, D., Shao, L., 2019. Crowd counting and density estimation by trellis encoder-decoder networks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kaartinen, H., Hyyppä, J., Yu, X., Vastaranta, M., Hyyppä, H., Kukko, A., Holopainen, M., Heipke, C., Hirschmugl, M., Morsdorf, F. et al., 2012. An international comparison of individual tree detection and extraction using airborne laser scanning. *Remote Sensing*, 4(4), 950–974.

Kamilaris, A., Prenafeta-Boldú, F. X., 2018. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, 147, 70–90.

Kang, D., Ma, Z., Chan, A. B., 2018. Beyond Counting: Comparisons of Density Maps for Crowd Analysis Tasks—Counting, Detection, and Tracking. *IEEE Transactions on Circuits and Systems for Video Technology*, 29(5), 1408–1422.

Ke, Y., Quackenbush, L. J., 2011. A review of methods for automatic individual tree-crown detection and delineation from passive remote sensing. *International Journal of Remote Sensing*, 32(17), 4725–4747.

Kingma, D. P., Ba, J., 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Kobayashi, F. K., Mattos, A. B., Macedo, M. M., Gemignani, B. H., 2019. Citrus tree classification from uav images: Analysis and experimental results. *Anais do XV Workshop de Visão Computacional*, SBC, 31–36.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature*, 521(7553), 436.

Lempitsky, V., Zisserman, A., 2010. Learning to count objects in images. *Advances in Neural Information Processing Systems*, 1324–1332.

Li, W., Fu, H., Yu, L., Cracknell, A., 2017. Deep learning based oil palm tree detection and counting for high-resolution remote sensing images. *Remote Sensing*, 9(1), 22.

Long, J., Shelhamer, E., Darrell, T., 2015. Fully convolutional networks for semantic segmentation. *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 3431–3440.

Ma, Z., Yu, L., Chan, A. B., 2015. Small instance detection by integer programming on object density maps. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Odena, A., Dumoulin, V., Olah, C., 2016. Deconvolution and Checkerboard Artifacts. *Distill*. http://distill.pub/2016/deconv-checkerboard.

Ruder, S., 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*.

Shui, C., Abbasi, M., Robitaille, L.-É., Wang, B., Gagné, C., 2019. A Principled Approach for Learning Task Similarity in Multitask Learning. *arXiv preprint arXiv:1903.09109*.

Sindagi, V. A., Patel, V. M., 2018. A survey of recent advances in CNN-based single image crowd counting and density estimation. *Pattern Recognition Letters*, 107, 3–16.

Volpi, M., Tuia, D., 2016. Dense semantic labeling of subdecimeter resolution images with convolutional neural networks. *IEEE Transactions on Geoscience and Remote Sensing*, 55(2), 881–893.

Xie, W., Noble, J. A., Zisserman, A., 2018. Microscopy cell counting and detection with fully convolutional regression networks. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, 6(3), 283–292.

Zhang, T., Suen, C. Y., 1984. A fast parallel algorithm for thinning digital patterns. *Communications of the ACM*, 27(3), 236–239.

Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y., 2016. Single-image crowd counting via multi-column convolutional neural network. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 589–597.

Zortea, M., Macedo, M. M., Mattos, A. B., Ruga, B. C., Gemignani, B. H., 2018a. Automatic citrus tree detection from uav images based on convolutional neural networks. *Conference on Graphics, Patterns and Images, 31. (SIBGRAPI), 2018, Foz do Iguaçu, PR, Brazil*. Available from http://urlib.net/rep/8JMKD3MGPAW/3S4HQQS.

Zortea, M., Nery, M., Ruga, B., Carvalho, L. B., Bastos, A. C., 2018b. Oil-palm tree detection in aerial images combining deep learning classifiers. *IGARSS*, IEEE, 657–660.

Zortea, M., Santos, M. N., Zadrozny, B., Schoeninger, E. R., Stetz, C. C., 2017. Detecting individual eucalyptus crowns in aerial photographs using template matching and classification. *Anais...*, 1, Simpósio Brasileiro de Sensoriamento Remoto, 18. (SBSR)., Instituto Nacional de Pesquisas Espaciais (INPE), São José dos Campos, 6749–6756. Available from http://urlib.net/rep/8JMKD3MGP6W34M/3PSMDFK.

This contribution has been peer-reviewed. The double-blind peer-review was conducted on the basis of the full paper.
https://doi.org/10.5194/isprs-annals-IV-3-W2-2020-65-2020 | © Authors 2020. CC BY 4.0 License.
Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165654