An Approach for Semantic Segmentation of Tree-like Vegetation

S. Tejaswi Digumarti[1], Lukas Maximilian Schmid[2], Giuseppe Maria Rizzi[2],
Juan Nieto[3], Roland Siegwart[3], Paul Beardsley[4], Cesar Cadena[3]

***Abstract*— This paper presents a pipeline for semantic segmentation of trees into their components. Given a single RGB-D image of a tree, we employ a deep network to predict labels to classify each pixel of the tree into trunk, branches, twigs and leaves. Multiple convolutional neural network architectures to combine the complementary modalities of depth and colour data are investigated. An asynchronous training approach where two networks trained separately on RGB and depth encoded as a 3-channel HHA image are combined using a late fusion architecture with different learning rates performs the best. Training and evaluation are performed on a synthetic dataset of 6 species of broadleaf trees. We further demonstrate the network's generalization capabilities, across various tree species on the synthetic dataset, achieving an accuracy of upto 92.5%. Furthermore, we present a qualitative evaluation of our approach on real-world data.**

I. Introduction

Automation in agriculture and plantation management paves way for a cost-effective and time-saving approach to increasing crop yield. Robotic systems are being deployed to perform laborious tasks such as dormant pruning [1], fruit picking [2], weed detection [3] and orchard management [4]. The effectiveness of these tasks directly depends on the quality of models of vegetation that can be generated from scans of the environment and in understanding the semantics within. In this paper we propose an algorithm that segments parts of a tree into its components, namely trunk, branches, twigs and leaves.

Recent advances in sensing and motion planning have enabled the capture of multi-modal, high resolution data from the environment [5]. This data, vital for robot vision tasks like mapping and localization, has promoted a growing amount of research in a wide variety of fields. However, these methods do not directly translate to environments with vegetation. This is because, unlike human-made structures, natural structures like trees are characterized by complex geometry, non-parametric surfaces, repeating elements, self occlusion, non-rigidity and limited variation of colour. For example, reconstruction techniques such as [6], [7], [8] perform sub-optimally as they are forced into local minima due to small repetitive structures in the leaf-regions of a tree. Similarly camera pose estimation techniques that rely on correspondence matching [9], [10], [11] again fail due to the lack of distinctive salient points in these regions. Segmenting or masking out these regions is one way to address this issue and forms a strong motivation for the work presented in this paper.

Such a task involves understanding the semantics of the scene. While semantic segmentation has been employed for various robotics applications such as robot manipulation [12], autonomous navigation [13] and camera localization [14], its use in segmenting natural structures, like trees, into their components is still an open area for research. In this paper we present an approach that bridges this research gap. Moreover, segmenting trees into wood and leaves enables computation of metrics such as leaf area index [15] and volume of wood, which find use in forestry and environment monitoring applications.

In this paper we propose a pipeline based on deep Convolutional Neural Networks (CNNs) the segments a tree into its components. Given a single RGB-D image, the network predicts labels classifying each pixel as trunk, branch, twig or a leaf. As the input data is multi-modal, i.e. consisting of both colour and depth information, we explore and evaluate multiple approaches to fuse the two modalities preserving their complementary benefits. Training a deep neural network requires a large amount of data and annotating real data is both cumbersome and impractical. Hence we generate the required dataset in simulation. Extensive quantitative evaluation is performed on the synthetic data under various test scenarios to evaluate the performance of the network and is presented in this paper. Finally, we also present qualitative results on real data. In the scope of this paper, we restrict our analysis to broadleaf trees and do not consider coniferous trees with needle like leaves.

The key contributions of this paper are as follows.

- A learning-based approach for semantic segmentation of trees into their components.
- An extensive analysis on various approaches for incorporating depth information with colour image data, with emphasis on a late fusion approach, as well as of the training scheduling.
- An evaluation of the approach when the network is

This research has been partially funded by the Swiss National Science Foundation through the National Center of Competence in Research Robotics (NCCR).

[1]S. T. Digumarti is with the Autonomous Systems Lab, ETH Zürich, 8092 Zürich, Switzerland, and also with Disney Research, 8006 Zürich, Switzerland. (*Corresponding author*) (email: dtejaswi@mavt.ethz.ch)

[2]L. M. Schmid and G. M. Rizzi are with the department of Mechanical and Process Engineering, ETH Zürich, 8092 Zürich, Switzerland. (email: grizzi@student.ethz.ch, schmluk@student.ethz.ch)

[3]C. Cadena, J. Nieto and R. Siegwart are with the Autonomous Systems Lab, ETH Zürich, 8092 Zürich, Switzerland.(email: cesarcadena.lerma@gmail.com; jnieto@ethz.ch; rsiegwart@ethz.ch)

[4]P. Beardsley is with Disney Research, 8006 Zürich, Switzerland. (email: pab@disneyresearch.com)

S. T. Digumarti, L. M. Schmid and G. M. Rizzi contributed equally to the work presented here.

trained solely on simulated tree data and tested on real data.

The rest of the paper is organised as follows. In section II an overview of related literature is presented, with an emphasis on CNN based segmentation networks and segmentation of vegetation. Details regarding the generation of the synthetic dataset of broadleaf trees are presented in section III. Section IV describes the network architecture along with different approaches investigated for incorporating depth information. Results for all the proposed architectures under various test settings are presented in section V. A qualitative evaluation of the network on real data is also presented in the same section. Section VI concludes the paper.

II. Related Work

A. Semantic Segmentation

Deep learning based approaches have surpassed traditional geometry based approaches for semantic segmentation. These methods, mostly based on CNNs, typically consist of a pre-trained encoder as in [16] and a decoder. The Fully Convolutional Network (FCN) [17], extends CNNs by replacing the fully connected layers with convolutional ones thus allowing arbitrary input sizes. FCNs form the basis of most state-of-the-art segmentation networks. Another highly successful network, the SegNet [18] follows a similar architecture, with the novelty being the use of pooling indices computed in the max-pooling step of the corresponding encoder to perform non-linear up-sampling. An alternate solution to upsampling is presented in DeepLab [19]. Multimodal learning introduced in [20] leverages complementary benefits if the different modalities to improve performance. For example Kwang et al. [21] use multi-spectral images combining thermal and colour images to detect pedestrians in a scene. With the availability of commercial RGB-D sensors, approaches combining depth and colour information have been developed. Gupta et al. [22] propose the HHA encoding for depth images and combine features extracted on this image with colours images using an SVM classifier. Eitel et al. [23] instead apply a colour-map to normalized depth images and use that as another 3-channel image stream similar to colour images. More recently, Valada et al. [13] use depth information directly but replace the convolutional layers with residual [24] layers to increase the depth of the networks. The method proposed in this paper is similar to the method of [22] but instead we replace the classifier with a set of trainable convolutional layers. We also use an asynchronous training approach with different learning rates for different parts of the network. For a comprehensive review of deep semantic segmentation, we direct the reader to [25].

B. Semantic Segmentation of Vegetation

Previous literature on vegetation segmentation mainly addresses the task of segmenting crops from the background in an agricultural field. Zheng et al. [26] extract features from images in different colour spaces and apply the mean-shift algorithm followed by a simple neural network for segmenting crops from the field. Moorthy et al. [27] follow a similar approach but use a Bayesian classifier for prediction. More recently, [28] and [3] propose learning based methods to separate crops from weeds using RGB and Multi-spectral images respectively.

However, there is little research on segmenting a tree into its components. Li et al. [29] use geometric properties like normal orientation as discriminating features for segmentation, but require manual tuning or heuristics to improve the predictions. Surface curvatures extracted at multiple neighbourhood sizes are used as features for segmenting different parts of a tree in [30], but their method works on the 3D point cloud of an entire tree. In contrast our method works on single RGB-D images and does not require the complete tree to be reconstructed.

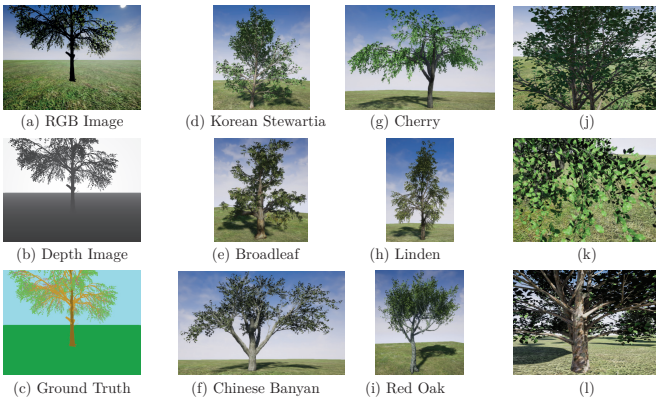


Fig. 1: Overview of our dataset. A data point consists of colour, depth and ground truth images (a)-(c). In (d)-(i), the 6 different tree species are shown. Sample images (j)-(l) illustrate the differences in scale, lighting and species.

III. Dataset

Segmentation of natural structures like trees is a very specific application. Large image segmentation datasets such as the SUN dataset [31] have images of nature but the ground truth segmentation labels are coarse and do not capture finer details such as leaves and twigs that are relevant to our work. At the same time annotating pictures of real trees is impractical. Therefore, we generated a synthetic dataset of trees for this paper.

3D models of trees were generated using SpeedTree® [32]. These models were imported into a simulation framework built in Unreal Engine 4.19 [33]. In order to simulate a real-world robot capture scenario, a drone with an RGB-D camera was simulated using the AirSim [34] plugin. The RGB-D camera was modelled after a Kinect One (v2) [35] sensor.

Colour (RGB), depth and ground-truth segmentation images were collected with the drone flying a spiral trajectory around the tree. The camera was always pointed towards the tree with its axis parallel to the ground plane. Images were also taken at multiple distances from the tree (1 m to 8 m) to capture variation in scale.

Fig. 2: A Kinect One (v2) sensor mounted on a garden pole used for hand-held scanning of trees. The device is powered using a portable battery.

Typically, a tree has self-occlusions and as a result the same structure appears completely different even from slightly different viewpoints. At the same time this also results shadows that are responsible for a lack of colour variation across different components of the tree. This was simulated by illuminating the trees using a natural directional light.

In this paper we limit our dataset to just broadleaf trees. We also do not consider external disturbances such as wind and changes in global illumination during the course of scanning. Furthermore, there are no flowers or fruits on the trees. To maintain generality, 6 tree species, with at least 5 instances per species were used providing sufficient variation in terms of tree topology, leaf density and leaf characteristics. For every tree, a total of 720 images was taken, resulting in an overall dataset size of 28800 samples. A set of sample images representing the dataset is shown in Figure 1.

Since the background is not of interest for our task, a simple background was chosen in the simulation. In general this is not a limitation since many state-of-the-art deep networks, such as SegNet [18], can segment the general class "vegetation" from other objects in an outdoor scene. We can use this as a pre-processing step to our algorithm. However, environments where the background also contains vegetation can be challenging and will be addressed in future work. In the scope of this paper, we exclude the background classes from our evaluation.

Real data for evaluation was collected using a Kinect One (v2) sensor mounted on a gardening pole as shown in Figure 2 Images were collected by walking around trees in a circular path with the Kinect sensor pointed towards the tree. A similar setup on a drone can also be used to collect images of larger trees.

IV. Network Architectures

In this section we describe all the network architectures considered for incorporating depth information along with colour images.

A. RGB network

One of the most successful networks for segmentation of both indoor and outdoor scenes is SegNet [18]. In this paper we use it as the baseline and refer to this as the RGB network. The VGG [16] weights of the encoder part of SegNet are initialized to the values optimized by training on the Imagenet dataset [36]. These are tuned further by training on our dataset.

(a) Depth Channel (b) Height Channel (c) Angle Channel

Fig. 3: Figure shows the 3 channels of an HHA encoded depth image. Blue represents small values while red represents large values. The depth image has no noise added for the sake of illustration.

B. RGB-D network

As mentioned earlier colour information alone may be ambiguous due to lack of sufficient variation. Hence using a complementary modality, such as depth information, is valuable to improve segmentation results [20]. There are two common approaches to incorporate depth information. The first approach is to perform early fusion where the depth image is remapped to the range of colour images i.e. $[0,255]$ and concatenated as a fourth channel with the colour image. With the exclusion of the first layer, no additional changes are made to the network architecture. However, as the input substantially differs from that of the RGB network, the pre-trained weights of the encoder are not relevant anymore. Hence, we re-train the network from scratch with weights initialized as described in [37]. We refer to this architecture as the RGB-D network.

C. RGB-HHA and RGB-HA network

The second approach to incorporate depth information is to perform late fusion, i.e. in the feature space. The depth image is encoded as a 3-channel HHA [22] image comprising of depth, height from the ground and angle of the surface normal with gravity as the three channels.

A prominent characteristic of natural structures is that the woody regions, i.e. trunk and branches, appear cylindrical while, leaves appear flat. The HHA encoding implicitly captures these characteristics, but still maintains enough raw data for the network to learn features independently. Hence this encoding was chosen in this paper.

We compute the angle channel using the algorithm proposed in [38]. All channels are linearly scaled to map observed values across the training dataset to the range $[1,255]$. The value 0 is reserved for pixels with invalid depth. The depth channel is thresholded and scaled inversely mapping closest depth to 255 and farthest depth to 1 to prevent discontinuities between far away points and out-of-range values. Figure 3 shows the HHA encoding for a sample depth image (without noise for the sake of illustration).

We use an architecture similar to the one proposed in [23] where the HHA image is trained in parallel to the RGB network, referred to as the HHA-net. In this approach, the

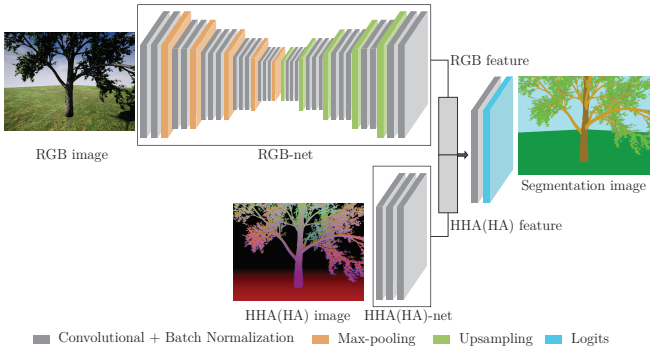


Fig. 4: Architecture of the RGB-HHA(-a) or RGB-HA(-a) networks. Top row shows the RGB network trained on colour images while the bottom row is the smaller HHA-net(or HA-net) trained on HHA(or HA) images. The two modalities are combined using a late fusion layer followed by the prediction layer.

HHA-net is only meant to improve the results of the RGB network but not to achieve stand-alone optimal performance. It consists of four convolutional layers of $8 \rightarrow 16 \rightarrow 32 \rightarrow 64$ filters of dimension $24 \rightarrow 12 \rightarrow 6 \rightarrow 3$ respectively. The larger size in the first layers helps in increasing the *receptive field*, thus compensating the absence of max pooling. The increased number of filters instead boosts the discriminating power of the features learnt from the previous layers. Finally, the outputs of the two branches are concatenated and passed through another convolutional layer before final classification. We refer to this architecture as the RGB-HHA network. It is illustrated in Figure 4.

In a general robot scanning scenario, the height above ground may not always be available, since only a relative pose with respect to an arbitrary initial reference frame is typically estimated. In order to account for this lack of height information, we also train a variation of the RGB-HHA network where the depth image is encoded as a two channel image with just depth and angle with the vertical direction as the channels. We refer to this architecture as the RGB-HA network.

D. Asynchronous training

In the case of a multi-modal architecture, one modality might out-weigh the other when the modalities are fused and trained together. In order to mitigate this effect, we introduce a technique which we refer to as *asynchronous training*. In this approach the RGB network and the HHA-net (or HA-net) are trained separately independent of each other. The two networks are then combined using late fusion as above and fine-tuned together, but with different learning rates for each network. A very small learning rate was used for the weights of the HHA(HA)-net thus reducing the influence of the RGB network on the HHA(HA)-net after fusion. In comparison, a slightly larger learning rate than that of the HHA(HA)-net was used for the weights of the RGB network encoder. This is because the RGB network weights are initialized using the VGG weights. The rest of the trainable variables were trained with higher training rates. The resulting architecture is *asynchronous* in the sense that different sections of the network adapt at different speeds. We refer to these architectures as the RGB-HHA-a and RGB-HA-a networks.

TABLE I: Table shows class frequencies over the entire dataset.

Class frequency [%]					
ground	sky	trunk	branch	twig	leaf
35.2	28.0	4.4	7.9	1.9	22.6

E. Class clustering and weighted loss

There are 6 classes in the dataset as acquired from the simulation; 4 for the components of a tree (*trunk*, *branch*, *twig*, *leaf*) and 2 for background (*ground*, *sky*) . As mentioned earlier, the background classes are not of importance and are omitted from the analysis. This division of the tree into 4 components was designed to explore the capabilities and limitations of the different network architectures for the segmentation of finer structures (leaves and twigs), as well as highly similar classes (branches versus twigs), which is sometimes hard even for a human observer. However, for applications such as 3D reconstruction, a simpler division of the tree into wood and leaf structures might be sufficient. We compare both the cases where the refined division is referred to as the 6 class case and the latter the 4 class case.

Classes appear in the images with different frequencies, as shown in Table I. To compensate for this unbalance, we explored the use of median frequency weighting as proposed in [39]. The frequency weights for each class are computed as the class frequency divided by the median class frequency over the training data, for every architecture and evaluation. Note that the distribution of classes is an inherent property of natural structures and only partly depends on how the images were taken.

V. Experimental results

A. Comparison of Architectures

All architectures were trained and tested on the synthetic dataset, featuring all 6 species of trees. Training was performed on 4 instances of each tree resulting in $17,280$ images, while testing was performed on the 5th tree with a total of $4,320$ images. The resulting per-class and overall accuracies are presented in Table II.

The RGB network is considered as the baseline for the sake of this analysis. The RGB-D network performs poorly indicating that the network is unable to learn meaningful features directly from the depth image in an early fusion scheme. However, the late fusion architectures show an improvement over the RGB network. The performance is further increased by employing asynchronous training, gaining on average 1% in accuracy from RGB-HHA to RGB-HHA-a and 5.5% in accuracy from RGB-HA to RGB-HA-a for the 6 channel case.

TABLE II: Table shows class accuracies for all different network architectures, class cases and weighting schemes. The total accuracy column excludes the background classes of ground and sky.

		Class accuracies									
		Weighted					Non Weighted				
		trunk	branch	twig	leaf	total	trunk	branch	twig	leaf	total
6 classes	RGB	0.802	0.724	0.170	0.923	0.737	0.894	0.756	0.361	0.858	0.829
	RGB-D	0.312	0.432	0.175	0.740	0.625	0.444	0.167	0.339	0.523	0.369
	RGB-HHA	0.865	0.738	0.329	0.928	0.839	0.825	0.857	0.693	0.844	0.843
	RGB-HA	0.931	0.769	0.286	0.889	0.815	0.970	0.678	0.307	0.873	0.809
	RGB-HHA-a	0.910	0.886	0.277	0.921	0.838	0.827	0.780	0.522	0.907	0.861
	RGB-HA-a	0.877	0.789	0.407	0.920	0.86	0.922	0.800	0.568	0.908	0.878
4 classes	RGB	0.888			0.870	0.877	0.907			0.881	0.890
	RGB-D	0.724			0.749	0.744	0.541			0.373	0.406
	RGB-HHA	0.935			0.905	0.916	0.945			0.888	0.908
	RGB-HA	0.921			0.913	0.916	0.936			0.902	0.914
	RGB-HHA-a	0.942			0.913	0.924	0.924			0.925	0.925
	RGB-HA-a	0.944			0.834	0.867	0.932			0.920	0.925

In general the class accuracy for twigs is low while for the other more frequent classes the accuracies are high. Employing a weighted loss mostly affects the twig class since it is the rarest. However, its accuracy decreases since the classification is biased towards twigs and they are still rare. For the other classes it is a trade-off with approximately equal accuracy gains and losses. Nevertheless, even when applying weighting, the recall for the twigs class only increased by 26% on average, reaching a maximum of 69% for the RGB-HHA-a network at the expense of recall for other classes. When combining all the *woody* classes together, we see that the accuracies are higher than with the refined division. We further observe that the HHA and HA predictions are generally comparable, suggesting that height above ground may be omitted if not available.

B. Inter-species generalization

To estimate the generalization capabilities of the networks across various tree species, the two best performing networks, RGB-HHA-a and RGB-HA-a for 4 and 6 classes respectively, were selected for further analysis. The number of training and testing images is kept constant among different training settings and equal number of images are sampled from each species and tree instance. A total of 5760 training and 1440 testing images were used for each setting. In every setting 3 combinations of species were randomly chosen, trained on and tested. We refer to predictions on a tree as being *in-species* if the training set contains another instance from the same species as that of the tree, else it is an *out-of-species* prediction. The resulting in- and out-of-species test accuracies are depicted in Figure 5. The in-species accuracy is approximately constant, indicating that the networks learn proper descriptors for all the species that they are trained on. The out-of-species accuracy decreases only by 6% and 10% for the 2 species case and increases quickly with more species trained on. This suggests that training on a finite set of species is sufficient to expect decent performance on a broader species dataset.

C. Noise analysis

In real-world data, the depth images are typically not as clean as the synthetic data. To investigate the influence of noise on the prediction output, the HHA networks were tested on depth images perturbed by noise as described in [40] which is models the noise of the Kinect One (v2) sensor. Please note that this noise model is Gaussian and only covers part of the noise that is seen in real depth images. Other sources of noise such as shadows at object boundaries are difficult to model and have not been considered in the simulation. We observe from Table III that there is drop in accuracy of about 5% as compared to the accuracies with clean data.

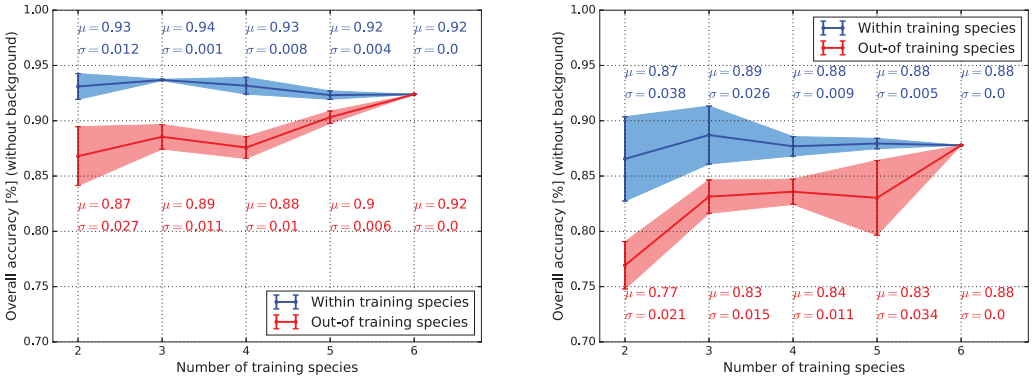
D. Qualitative evaluation on real-world data

In order to test the potential of the networks on real-world data, predictions of the networks on real RGB-D images, acquired using a Kienct One(v2) sensor, were qualitatively evaluated. These are shown in Figure 6. A major difference between real and synthetic data is that real depth images typically contain missing data especially around object contours (due to infrared shadows) and suffer from other limitations summarized in [35]. While complicated methods for improving the quality of these images are available, in the the scope of this analysis, we limit ourselves to using a median filter, with a window of 7x7 pixels, to get rid of small holes and noise.

The RGB-HHA networks trained on noisy data are used for this analysis. As expected, we see that depth information helps in predicting correct labels where there is colour ambiguity. This preliminary deployment of synthetic training on real world data motivates simulation based training. It further suggests that incorporating depth in a skip lightweight architecture augmentation can improve transfer capabilities to real world applications. We would like to mention that quantitative evaluation on real data was not possible because of the absence of labelled datasets of vegetation.

TABLE III: Table shows prediction accuracies for the HHA networks, trained and evaluated on data perturbed by Kinect One noise model. The total accuracy column excludes the background classes of ground and sky.

		Class accuracies for noise analysis									
		Weighted					Non Weighted				
		trunk	branch	twig	leaf	total	trunk	branch	twig	leaf	total
6 cls	RGB-HHA	0.644	0.447	0.330	0.926	0.757	0.763	0.748	0.144	0.865	0.791
	RGB-HHA-a	0.673	0.636	0.302	0.868	0.771	0.717	0.650	0.089	0.926	0.803
4 cls	RGB-HHA	0.788			0.916	0.881	0.717			0.938	0.862
	RGB-HHA-a	0.764			0.929	0.870	0.814			0.897	0.876



(a) RGB-HHA-a prediction for 4 classes.

(b) RGB-HA-a prediction for 6 classes.

Fig. 5: Figure shows in-species (blue) and out-of-species (red, dashed) prediction accuracies for the (a) RGB-HHA-a and (b) RGB-HA-a architectures, represented as mean and standard deviation of 3 randomly sampled combinations of training species for each step.

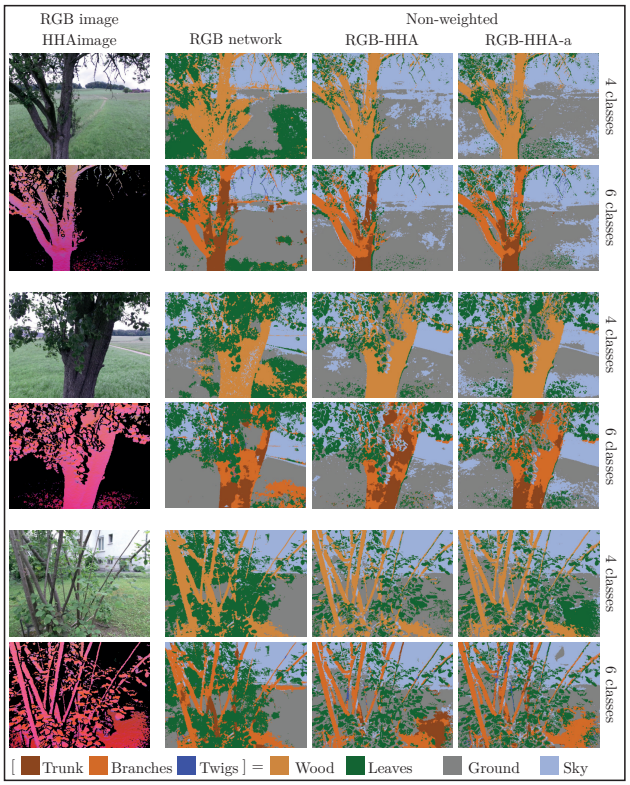


Fig. 6: Figure shows the predictions of select networks on images of real data.

VI. Conclusion

In this paper we proposed the use of deep convolutional networks for semantic segmentation of trees into their components. Multiple network architectures for incorporating depth information along with colour images were analyzed on a synthetic dataset, where a late fusion approach performed best. Further improvement in performance was achieved using an *asynchronous* training procedure with different learning rates.

For tasks where a division of the natural structure into just wood and non-wood classes is sufficient, higher accuracy in segmentation may be achieved. We also show that the networks generalize well among different species of trees.

Qualitative results on real data suggest that the features learnt by the network in simulation are meaningful and transferable to the real world.

References

[1] H. Medeiros, D. Kim, J. Sun, H. Seshadri, S. A. Akbar, N. M. Elfiky, and J. Park, "Modeling dormant fruit trees for agricultural automation," *Journal of Field Robotics*, vol. 34, no. 7, pp. 1203–1224, 2017.

[2] C. Schuetz, J. Baur, J. Pfaff, T. Buschmann, and H. Ulbrich, "Evaluation of a direct optimization method for trajectory planning of a 9-dof redundant fruit-picking manipulator," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 2660–2666.

[3] I. Sa, Z. Chen, M. Popović, R. Khanna, F. Liebisch, J. Nieto, and R. Siegwart, "weednet: Dense semantic weed classification using multispectral images and mav for smart farming," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 588–595, 2018.

[4] S. Bargoti, J. P. Underwood, J. I. Nieto, and S. Sukkarieh, "A pipeline for trunk detection in trellis structured apple orchards," *Journal of Field Robotics*, vol. 32, no. 8, pp. 1075–1094, 2015.

[5] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.

[6] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon, "Kinectfusion: Real-time dense surface mapping and tracking," in *Proceedings of the 10th IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, 2011, pp. 127–136.

[7] T. Whelan, R. F. Salas-Moreno, B. Glocker, A. J. Davison, and S. Leutenegger, "Elasticfusion: Real-time dense slam and light source estimation," *The International Journal of Robotics Research*, vol. 35, no. 14, pp. 1697–1716, 2016.

[8] R. A. Newcombe, D. Fox, and S. M. Seitz, "Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 343–352.

[9] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the seventh IEEE International Conference on Computer Vision (ICCV)*, vol. 2, 1999, pp. 1150–1157.

[10] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European Conference on Computer Vision (ECCV)*. Springer, 2006, pp. 404–417.

[11] S. Leutenegger, M. Chli, and R. Y. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2011, pp. 2548–2555.

[12] M. Schwarz, A. Milan, A. S. Periyasamy, and S. Behnke, "Rgb-d object detection and semantic segmentation for autonomous manipulation in clutter," *The International Journal of Robotics Research*, vol. 37, no. 4-5, pp. 437–451, 2018.

[13] A. Valada, J. Vertens, A. Dhall, and W. Burgard, "Adapnet: Adaptive semantic segmentation in adverse environmental conditions," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 4644–4651.

[14] N. Radwan, A. Valada, and W. Burgard, "Vlocnet++: Deep multitask learning for semantic visual localization and odometry," *arXiv preprint arXiv:1804.08366*, 2018.

[15] D. J. Watson, "Comparative physiological studies on the growth of field crops: I. variation in net assimilation rate and leaf area between species and varieties, and within and between years," *Annals of botany*, vol. 11, no. 41, pp. 41–76, 1947.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3431–3440.

[18] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 39, no. 12, pp. 2481–2495, 2017.

[19] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2018.

[20] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *Proceedings of the 28th International Conference on Machine Learning (ICML)*, 2011, pp. 689–696.

[21] S. Hwang, J. Park, N. Kim, Y. Choi, and I. So Kweon, "Multispectral pedestrian detection: Benchmark dataset and baseline," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1037–1045.

[22] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning rich features from rgb-d images for object detection and segmentation," in *European Conference on Computer Vision*. Springer, 2014, pp. 345–360.

[23] A. Eitel, J. T. Springenberg, L. Spinello, M. Riedmiller, and W. Burgard, "Multimodal deep learning for robust rgb-d object recognition," in *Proceedings of the IEEE Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 681–687.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[25] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, and J. Garcia-Rodriguez, "A review on deep learning techniques applied to semantic segmentation," *arXiv preprint arXiv:1704.06857*, 2017.

[26] L. Zheng, J. Zhang, and Q. Wang, "Mean-shift-based color segmentation of images containing green vegetation," *Computers and Electronics in Agriculture*, vol. 65, no. 1, pp. 93–98, 2009.

[27] S. Moorthy, B. Boigelot, and B. Mercatoris, "Effective segmentation of green vegetation for resource-constrained real-time applications," in *Precision agriculture*. Wageningen Academic Publishers, 2015, pp. 93–98.

[28] A. Milioto, P. Lottes, and C. Stachniss, "Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in cnns," *arXiv preprint arXiv:1709.06764*, 2017.

[29] S. Li, L. Dai, H. Wang, Y. Wang, Z. He, and S. Lin, "Estimating leaf area density of individual trees using the point cloud segmentation of terrestrial lidar data and a voxel-based model," *Remote Sensing*, vol. 9, no. 11, p. 1202, 2017.

[30] S. T. Digumarti, J. Nieto, C. Cadena, R. Siegwart, and P. Beardsley, "Automatic segmentation of tree structure from point cloud data," *To appear in IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3043–3050, Oct 2018.

[31] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba, "Sun database: Large-scale scene recognition from abbey to zoo," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010, pp. 3485–3492.

[32] Speedtree ue4. [Online]. Available: https://store.speedtree.com/store/ue4-starter-package-ue4/

[33] Unreal engine 4. [Online]. Available: https://www.unrealengine.com/en-US/what-is-unreal-engine-4

[34] S. Shah, D. Dey, C. Lovett, and A. Kapoor, "Airsim: High-fidelity visual and physical simulation for autonomous vehicles," in *Field and Service Robotics*, 2017. [Online]. Available: https://arxiv.org/abs/1705.05065

[35] O. Wasenmüller and D. Stricker, "Comparison of kinect v1 and v2 depth images in terms of accuracy and precision," in *Asian Conference on Computer Vision*. Springer, 2016, pp. 34–45.

[36] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 248–255.

[37] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proceedings of the IEEE International Conference on Computer Vision(ICCV)*, 2015, pp. 1026–1034.

[38] S. Gupta, P. Arbelaez, and J. Malik, "Perceptual organization and recognition of indoor scenes from rgb-d images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 564–571.

[39] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 2650–2658.

[40] P. Fankhauser, M. Bloesch, D. Rodriguez, R. Kaestner, M. Hutter, and R. Siegwart, "Kinect v2 for mobile robot navigation: Evaluation and modeling," in *Proceedings of the International Conference on Advanced Robotics (ICAR)*, 2015, pp. 388–394.