SEMANTIC SEGMENTATION OF ENDANGERED TREE SPECIES IN BRAZILIAN SAVANNA USING DEEPLABV3+ VARIANTS

D. L. Torres[1],*, R. Q. Feitosa[1], L. E. C. La Rosa[1], P. N. Happ[1], J. Marcato Jr[2], W.N. Gonçalves[2], J. Martins[2], V. Liesenberg[3]

[1] Dept. of Electrical Engineering, Pontifical Catholic University of Rio de Janeiro, Brazil
[2] Faculty of Engineering, Architecture and Urbanism and Geography, Federal University of Mato Grosso do Sul, Brazil.
[3] Department of Forest Engineering, Santa Catarina State University, Lages, Brazil.

KEY WORDS: Fully Convolution Neural Networks, Unmanned Aerial Vehicles (UAVs), Deep Learning, Semantic Segmentation.

ABSTRACT:

Knowing the spatial distribution of endangered tree species in a forest ecosystem or forest remnants is a valuable information to support environmental conservation practices. The use of Unmanned Aerial Vehicles (UAVs) offers a suitable alternative for this task, providing very high-resolution images at low costs. In parallel, recent advances in the computer vision field have led to the development of effective deep learning techniques for end-to-end semantic image segmentation. In this scenario, the DeepLabv3+ is well established as the state-of-the-art deep learning method for semantic segmentation tasks. The present paper proposes and assesses the use of DeepLabv3+ for mapping the threatened *Dipteryx alata* Vogel tree, popularly also known as cumbaru. We also compare two backbone networks for feature extraction in the DeepLabv3+ architecture: the Xception and MobileNetv2. Experiments carried out on a dataset consisting of 225 UAV/RGB images of an urban area in Midwest Brazil demonstrated that DeepLabv3+ was able to achieve in mean overall accuracy and F1-score above 90%, and IoU above 80%. The experimental analysis also pointed out that the MobileNetv2 backbone overcame its counterpart by a wide margin due to its comparatively simpler architecture in view of the available training data.

1. INTRODUCTION

Over the years, many remote sensing techniques like LiDAR (light detection and ranging), hyperspectral, optical and SAR (synthetic aperture radar) imaging, have been widely used for performing large-scale analysis of forest systems (Jeronimo et al., 2018, Laurin et al., 2013, Alonzo et al., 2014). Within the larger field of mapping forest trends, monitoring of endangered species populations has received increasing attention (Wang et al., 2016, Santos et al., 2019). In this context, for single tree detection, it is fundamental to understand crown morphology. This involves not only delineating and measuring the size, but also obtaining afterwards the volume, and diameter of the single tree crowns (Mohan et al., 2017).

In the last years, advances in unmanned aerial vehicles (UAVs) technology offered a suitable alternative to standard remote sensing solutions, providing high-resolution images at lower costs (Mohan et al., 2017). In this way, UAV images have aroused great interest within the remote sensing community and have been used for a wide variety of subjects (Honkavaara et al., 2013, Lizarazo et al., 2017, Rauhala et al., 2017, Chenari et al., 2017). In particular, studies focused on individual tree level mapping through UAV images, suggest its potential for the detection and delineation of tree crowns, and subsequently estimate parameters of its morphology (Lim et al., 2015, Grznárová et al., 2019, Tang, Shao, 2015).

In contrast with traditional machine learning algorithms, deep learning (DL) methods present the ability to learn representations directly from data. This capacity constitute a huge advance in the computer vision field, since they are able to automatic extract high level features for a particular classification task. For instance, Convolutional Neural Networks (CNNs) boosted the state of the art methods and have already proved to be very efficient in several tasks such as classification (Fassnacht et al., 2016, Krizhevsky et al., 2017), object detection (Baena et al., 2017, Santos et al., 2019) and semantic segmentation (Dechesne et al., 2017, Maschler et al., 2018). Therefore, in the forest monitoring context, many state-of-the-art DL methods have been used recently, especially to classify and detect tree species (Mizoguchi et al., 2017, Guan et al., 2015, Hartling et al., 2019). However, most of those solutions are based on LiDAR data.

Combining the power of DL methods with the ease of use of UAV platforms, (Santos et al., 2019) proposed a method for detecting endangered tree species, more specifically the *Dipteryx alata* Vogel (Fabaceae) tree, popularly known as cumbaru (henceforth cumbaru), using available RGB images. In this case, a CNN is trained to delineate a bounding box around the crown of the single target trees, providing information about their position and location. Although this information is valuable, some key important details regarding the morphology of the tree crown, like its individual shape or contour, are not provided. In this sense, semantic segmentation based algorithms arise as an alternative to achieve fine-grained information towards complete scene understanding, presenting the potential to capture object forms more accurately than single object detection.

Recent DL architectures for semantic image segmentation typically fall into one of two major approaches. The first, considers semantic segmentation as a classification problem using a typical sliding window procedure. In this case, the image is split into several snips or patches of the same size and then fed into a traditional CNN to classify the central pixel of each patch in a certain category (Farabet et al., 2012). One important drawback of this approach is its high computational complexity (Vigueras-Guillén et al., 2019). The second

*Corresponding author

This contribution has been peer-reviewed.

Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165625

approach comprises the efficient Fully Convolutional Networks (FCNs), in which the image classification is performed at a pixel level using an end-to-end network. Moreover, the learning and inference processes are performed for the whole input image at once (Long et al., 2014).

Regarding FCNs, one of the earliest models was proposed in (Badrinarayanan et al., 2017), introducing the encoder-decoder architecture. The encoder uses the basic convolutions and pooling operations to learn low- and high-level features, while the decoder path recovers the resolution of the input image. Even though the richest information is encoded in the last feature map, detailed location/spatial information related to object boundaries are missing due to the pooling or striding operations (Chen et al., 2014). In this context, modern networks such as the DeepLab architecture extended the previous approaches by introducing the atrous convolution (Chen et al., 2014). As a consequence, the high resolution of the feature maps are maintained, storing detailed information about the object boundaries, without increasing the number of parameters. This version was extended later by the addition of the Atrous Spatial Pyramid Pooling (ASPP), which is able to capture context information at multiple scales (Chen et al., 2016). At the same time, a combination with probabilistic models, like the Conditional Random Field (CRF), was proposed to improve localization performance (Chen et al., 2016). Furthermore, (Chen et al., 2017) presented a version to extend the ASPP module by using the image-level features of (Zhao et al., 2016) to add global context information and extra performance.

More recently, attempting to combine the advantages of these methods with the faster computation of encoder-decoder models, (Chen et al., 2018) presented the state-of-the-art DeepLabv3+ models, built on top of powerful CNNs. Their effectiveness were demonstrated by their significant results in the challenging PASCAL VOC 2012 (Everingham et al., 2015) and Cityscapes (Cordts et al., 2016) datasets.

Motivated by the state-of-the-art of DL semantic segmentation models and the high cost-benefit of using UAV RGB images for forest monitoring, this work aims to evaluate the usage of the DeepLabv3+ for the individual segmentation of the canopy of the endangered cumbaru trees. This paper has two major contributions: (i) evaluating the applicability of UAVs in generating RGB high-resolution observations for canopy tree segmentation, (ii) Assessing two backbone model variants of the state-of-the-art DeepLabv3+, more specifically the Xception and MobileNetv2 models, on the performance of cumbaru segmentation.

The rest of this paper is organized into four sections. First, in section 2, we introduce the fundamentals of DeepLab approach and its variants. On the sequence, in section 3, we describe the study area and the methodology adopted to evaluate the different models. In section 4, we present and discuss the results obtained in the experimental analysis. Finally, section 5 reviews the main conclusions and points to future directions.

2. METHODS

2.1 DeepLab Fundamentals

In the deep learning area, semantic image segmentation can be efficiently achieved by modifying the fully-connected layer of a traditional CNN into convolutional layers (Long et al., 2014). Typically, these networks consist of an encoder module, which reduces the feature maps resolution by convolution and pooling operations through consecutive layers, and a decoder module that retrieves the spatial resolution. The convolutional layers extract meaningful features by convolving the input image with kernels or filters. During convolution, each filter operates over a local region of the input volume, which is equivalent to the filter size. The spatial extent of the input image considered in calculating a position of an activation map is called receptive field or field of view. To enlarge the size of the receptive field, we can use larger filters or add more layers. Both imply more parameters, more operations, and higher computational complexity. To compensate for this effect and reduce the computational cost, pooling layers reduce the resolution of the feature maps. In consequence, part of spatial information gets lost, mainly at fine details.

To increase the field of view without increasing the number of parameters, (Chen et al., 2014) proposed the atrous convolutions. The basic idea consists in expanding a filter by including zeros between the kernel elements. For example, if a $k \times k$ filter is expanded by an expansion rate r, r-1 zeros are inserted between each adjacent element of the original filter along each dimension. Thus, the receptive field is expanded to $[k + (k-1)(r-1)] \times [k + (k-1)(r-1)]$ (Li et al., 2018) (see Figure 1). In this way, we increase the receptive field of the output layers without increasing the number of learnable kernel elements and the computational effort.

Later, (Chen et al., 2016) proposed the Atrous Spatial Pyramid Pooling (ASPP), based on parallel multiple layers of atrous convolutions. The objective is to capture contextual information at multiple scales, as shown in Figure 1. Notice that, the receptive field gets larger with increasing rates while maintaining the number of parameters (Li et al., 2018).

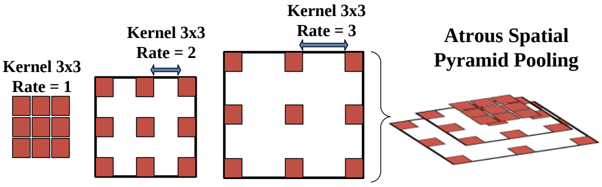

Figure 1. Atrous Spatial Pyramid Pooling

2.2 DeepLabv3+

The DeepLabv3+ version was built on its predecessor DeepLabv3 and brought back the encoder-decoder concept by adding a decoder module to improve the spatial accuracy (Chen et al., 2018). It applies separable convolution to both, the encoder and the decoder stages. Conceptually, the spatial separable convolution brakes down the convolution into two separate operations: a depthwise and a pointwise convolution, as illustrated in Figure 2.

In the traditional convolution, the kernel is as deep as the input and operates on all input channels. A depthwise separable convolution involves two steps. First, a spatial convolution is carried out independently over each channel of the input. After completing the depthwise convolution, a so-called pointwise convolution performs a 1×1 convolution across the channels.

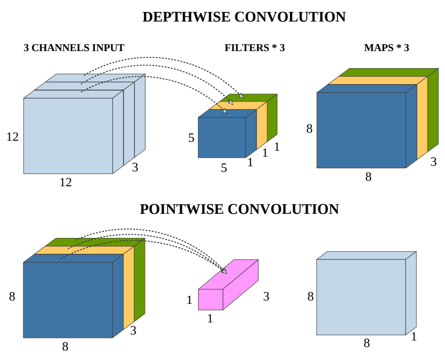

Figure 2. Depthwise Separable Convolution

The main advantage of depthwise separable convolutions is that they involve fewer parameters compared to regular convolutions, implying fewer operations and faster computation (Chollet, 2016).

In the encoder stage, the DeepLabv3 uses ASPP as feature extractor augmented with an image level feature (Zhao et al., 2016). This scheme concatenates the convolutions from kernels with different dilation rates to exploit multi-scale information, along with an image pooling to capture global context. The features are then bilinearly upsampled by a factor of 4 and then concatenated with the corresponding low-level features from the encoder stage. The low-level features are first convolved with a 1×1 filter before the concatenation, to reduce the numbers of channels (Chen et al., 2018). Then, a 3×3 convolution is applied to refine the features, followed by another bilinear upsampling by a factor of 4, see Figure 3.

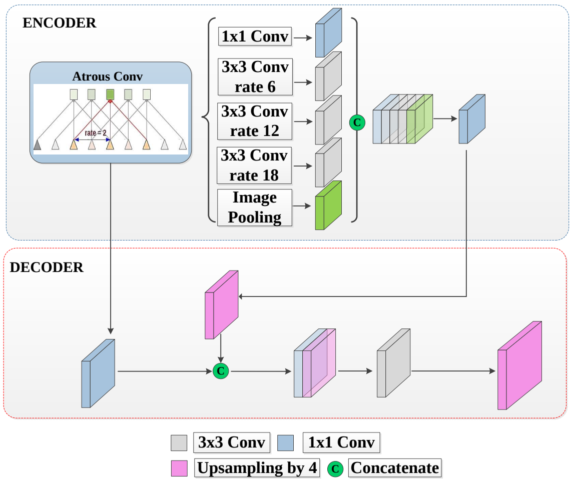

Figure 3. DeepLabv3+ Arquitecture adapted from (Chen et al., 2018)

DeepLabv3 + architecture can operate with two backbones: the powerful high-end Xception, or the computationally efficient MobileNetv2 for mobile devices.

2.2.1 Xception backbone A new network module, called Inception, was introduced in (Szegedy et al., 2014). Its central idea consists in factorizing a regular convolution explicitly into blocks of independent sequential operations: a cross-channel correlation followed by a spatial correlation. The Inception concept leads to less trainable parameters and faster computation with little or no harm to the ability to learn elaborated features. Chollet (Chollet, 2016) brought the Inception style to the extreme and proposed the Xception (*Extreme Inception*) architecture built entirely on depthwise separable convolutions. To address the segmentation task in DeepLabv3+ version, (Chen et al., 2018) applied the Modified Aligned Xception model as its feature extractor. This solution uses a deeper Xception module where all max-pooling operations are replaced by depthwise separable convolutions. A further batch normalization and a ReLU activation follow each depthwise convolution.

2.2.2 MobileNetv2 backbone The DeepLabv3+ version, called MobileNetv2, was conceived for mobile devices. It uses extensively depthwise separable convolutions to reduce the computational load (Howard et al., 2017). In particular, it introduced the so-called inverted residual block (Sandler et al., 2018). In standard residual blocks the input has a relative large number of channels (activation maps), which is first reduced in the subsequent layers and then expanded back to (approximately) the original depth. In the inverted residual block, occurs exactly the opposite. It starts with relative few channels, which are first expanded to be later compressed back. In both cases, a short-cut connections carries the input to be added to the residual computed in the block. Compared with the standard design the inverted counterpart is significantly more memory efficient. (Sandler et al., 2018).

3. EXPERIMENTS

3.1 Study Area and Data Acquisition

We evaluated the above-mentioned methods on a dataset that comprises 150,000 square kilometers in Campo Grande municipality, Mato Grosso do Sul, Brazil (Santos et al., 2019). The study site covers about 110 single cumbaru trees interspersed among roads, buildings, cars, and others, a typical urban scenario in Midwest Brazil. A total of 225 scenes were recorded with a UAV Phantom 4 equipped with a high-resolution RGB camera. The size of each image in the dataset is 5472×3648 with a spatial resolution of 0.82 cm. These images were acquired on August $13-th$ and $22-nd$ September 2018 at different times of the day, as seen in Figure 4 a-c. Observations were also taken at multiple distances from the trees (20m to 40m) to capture variations in scale. Each single cumbaru tree crown was delineated manually by a specialist. This procedure was also cross-checked with in-situ observations. Examples are depicted in Figure 4 d-f.

3.2 Experimental Setup

Both DeepLabv3+ variants, Xception and MobileNetv2, were trained with learning rate setting at 0.0001, using Adam optimizer with default beta values to update the gradient. As we have two possible outcomes, Cumbaru or background, the binary cross-entropy was used as loss function. The 225 images of the dataset were randomly split into three subsets: 70% for training, 10% for validation and 20% for test. The input images were cropped in patches of 512×512. The training batch size was empirically set to 2 and 6 for the Xception and MobileNetv2, respectively. Each model was trained up to 100 epochs to compensate for the smaller batch size and

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLII-3-W12-2020-355-2020 | © Authors 2020. CC BY 4.0 License.
Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165625

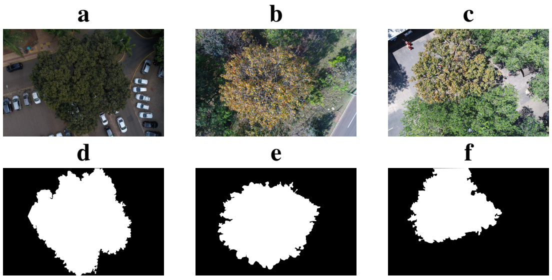

Figure 4. UAV observations of Cumbaru trees and their respective references.

early stopping was used as a regularization technique. The training stopped when the generalization error at the validation set degraded for 10 consecutive epochs. The best performing model in the validation stage was used for test.

Similar to (Chen et al., 2018), we adopted an output stride equal to 16, with atrous rates $r \in \{6, 12, 18\}$.

The performance of both methods is reported in terms of Overall Accuracy (OA), F1-score and Intersection over Union (IoU). The OA measures how well the binary classification correctly identifies the class *Cumbaru* (positives) and *background* (negatives). The F1-score conveys the balance between false positives and false negatives and is a good metric under uneven class distribution. IoU is computed by dividing the intersection area between the prediction and the ground-truth (Reference), by the total area covered by the prediction and ground-truth. These three metrics are defined by Equations 1, 2 and 3, respectively.

$$OA = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$F1 - score = \frac{2TP}{2TP + FP + FN} \tag{2}$$

$$IoU = \frac{|Reference \cap Prediction|}{|Reference \cup Prediction|} \tag{3}$$

where TP, FP, TN, and FN stand for true positive, false positive, true negative and false negative, respectively.

4. RESULTS AND DISCUSSIONS

Figure 5 summarizes the performance of each DeepLabv3+ variant. The values are averages over a 5-fold cross validation. Overall, the both techniques were quite successful. The DeepLabv3 with the Xception backbone achieved $88.9\%(\pm 2.7)$, $87.0\%(\pm 2.93)$ and $77.1\%(\pm 4.58)$, in terms of OA, F1-score and IoU, respectively. The MobileNetv2 variant performed better with $94.4\%(\pm 0.89)$ for OA, $93.5\%(\pm 1.24)$ for F1-score and $87.8\%(\pm 2.18)$ for IoU. The low variation across the folds further indicate the robustness of both architectures regarding the choice of training and test sets. It is, however, worth mentioning, that the MobileNetv2 performed a better than Xception also in this respect.

The superiority of MobileNetv2 may at first be regarded as unexpected because Xception has a higher capacity due to its more complex architecture. Indeed, Xception involves about twenty times as many parameters as MobileNetv2. When the amount of labeled data available for training does not match the number of parameters to be estimated, the network tends to generalize poorly. The results indicate that this might have occurred in our study. Under this assumption, it should be considered that significant improvements can still be obtained if a larger data set is available for network training.

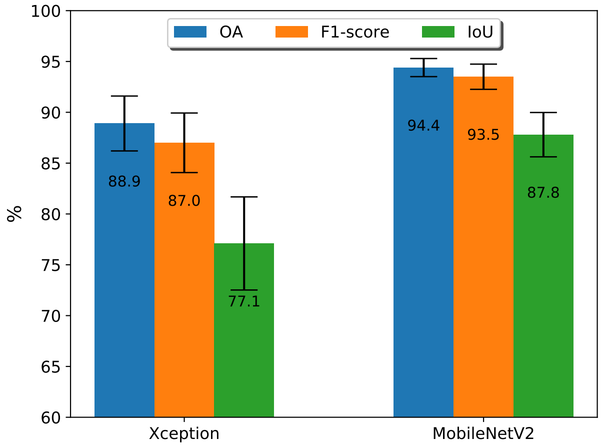

Figure 5. Results for 5-fold cross validation for DeepLabv3+ backbones

Figure 6 presents some visual results of both backbones for images at different scales and illumination patterns. Figure 6b) and Figure 6d) show small errors (false positives) over the street for both models. Similarly, portions of trees' canopies were predicted as background (false negatives), also by both models, as it also seen in Figure 6a),b) and d). The Figure confirms the general trend revealed in the plot of Figure 5, namely, that Xception were more prone to make errors than MobileNetv2.

Indeed, MobileNetv2 delivered better segmentation outcomes, mainly along object boundaries. Nonetheless, both methods underperformed in instances affected by poor lighting. This effect is especially apparent in Figure 6c).

In addition to the quality assessment, we also compared the algorithms in terms of processing time measured on the following hardware infrastructure: Intel(R) Core(TM) i7 processor, 64 GB of RAM and NVIDIA GeForce GTX228 1080Ti GPU. Table 1 summarizes the results.

Methods	Training Time (h:min)	Inference Time (sec)
Xception	20:33	4.44
MobileNetv2	10:46	2.26

Table 1. Computational Complexity for the DeepLabv3+ Xception and MobileNetv2 backbones

As expected, the mobile version (MobilNetv2) required less training and inference time, almost half as much as the Xception model. The Xception backbone was more computationally demanding due to its depth and the higher number of parameters.

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLII-3-W12-2020-355-2020 | © Authors 2020. CC BY 4.0 License.
Primary publication at IEEE Xplore: https://doi.org/10.1109/LAGIRS48042.2020.9165625

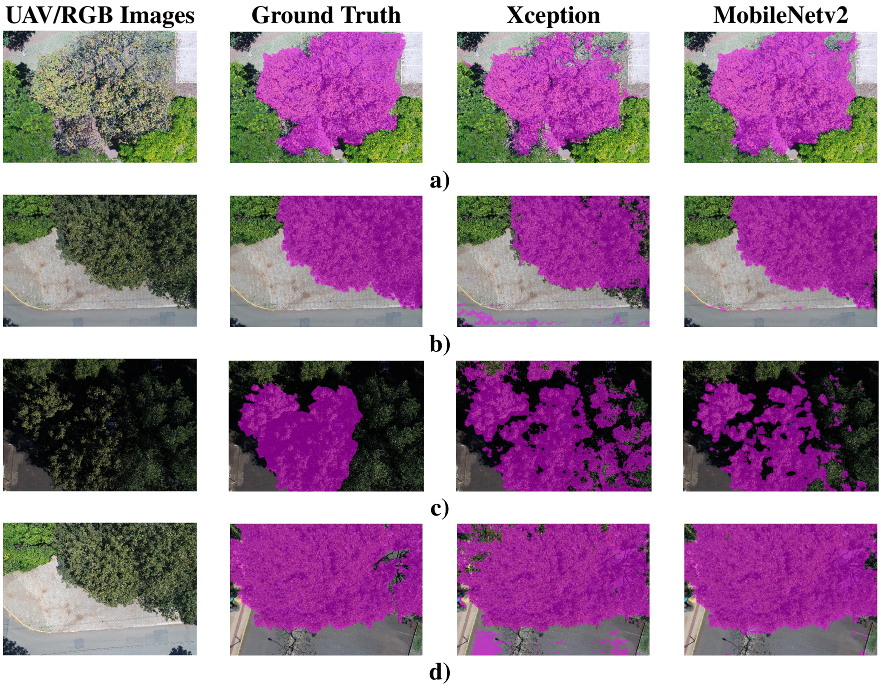

Figure 6. Qualitative results for DeepLabv3+ method using Xception and MobileNetv2 backbones

5. CONCLUSIONS AND FUTURE WORKS

In this work, we assessed the state-of-the-art DeepLabv3+ Fully Convolutional Network for the segmentation of instances of a tree species from high-resolution images, captured by a UAV platform. Two variants were tested: with the Xception and with the MobileNetv2 backbones.

Both DeepLabv3+ variants delivered encouraging results in an experimental analysis conducted on a data set comprising 225 RGB images.

The MobileNetv2 variant, consistently outperformed the Xception counterpart, by 5.5%, 6.5% and 10.7% in terms of overall accuracy, F1-score, and IoU, respectively. These results are significant, considering that they add up to absolute values above 75.0%. A visual analysis corroborated the quantitative results, demonstrating that the crowns were generally well delineated by both methods. Concerning the computational cost, the DeepLabv3+ model using MobileNetv2 backbone was about two times faster than Xception both in training as in inference time.

We intend to extend this study by including other semantic segmentation methods based on Fully Convolutional Networks. In the continuation of this research, we also plan to evaluate the benefits of post-processing based on Conditional Random Field, as in earlier DeepLab approaches. We further plan to test these methods in multi-temporal acquisitions to check crown changes over time and also in more challenging scenarios, specifically on datasets that comprise multiple tree species.

REFERENCES

Alonzo, M., Bookhagen, B., Roberts, D. A., 2014. Urban tree species mapping using hyperspectral and lidar data fusion. *Remote Sensing of Environment*, 148, 70–83.

Badrinarayanan, V., Kendall, A., Cipolla, R., 2017. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12), 2481–2495.

Baena, S., Moat, J., Whaley, O., Boyd, D. S., 2017. Identifying species from the air: UAVs and the very high resolution challenge for plant conservation. *PLOS ONE*, 12(11), 1-21.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A. L., 2014. Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs. *arXiv*, abs/1412.7062.

Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A. L., 2016. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *CoRR*, abs/1606.00915.

Chen, L., Papandreou, G., Schroff, F., Adam, H., 2017. Rethinking Atrous Convolution for Semantic Image Segmentation. *CoRR*, abs/1706.05587.

Chen, L., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *CoRR*, abs/1802.02611.

Chenari, A., Erfanifard, Y., Dehghani, M., Pourghasemi, H. R., 2017. Woodland Mapping at Single-Tree Levels Using Object-Oriented Classification of Unmmanned Aerial Vehicle (UAV) Images. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-4/W4, 43-49.

Chollet, F., 2016. Xception: Deep Learning with Depthwise Separable Convolutions. *CoRR*, abs/1610.02357.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B., 2016. The Cityscapes Dataset for Semantic Urban Scene Understanding. *CoRR*, abs/1604.01685.

Dechesne, C., Mallet, C., Le Bris, A., Gouet-Brunet, V., 2017. Semantic segmentation of forest stands of pure species combining airborne lidar data and very high resolution multispectral imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 126, 129–145.

Everingham, M., Eslami, S. M. A., Gool, L. V., Williams, C. K. I., Winn, J., Zisserman, A., 2015. The Pascal Visual Object Classes Challenge: A Retrospective. *International Journal of Computer Vision*, 111(1), 98–136.

Farabet, C., Couprie, C., Najman, L., LeCun, Y., 2012. Learning hierarchical features for scene labeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(8), 1915–1929.

Fassnacht, F., Latifi, H., Stereńczak, K., Modzelewska, A., Lefsky, M., Waser, L., Straub, C., Ghosh, A., 2016. Review of studies on tree species classification from remotely sensed data. *Remote Sensing of Environment*, 186, 64–87.

Grznárová, A., Mokros, M., Surovy, P., Slavík, M., Pondelík, M., Merganič, J., 2019. The Crown Diameter Estimation from Fixed Wing Type of UAV Imagery. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13, 337–341.

Guan, H., Yu, Y., Ji, Z., Li, J., Zhang, Q., 2015. Deep learning-based tree classification using mobile LiDAR data. *Remote Sensing Letters*, 6(11), 864–873.

Hartling, S., Sagan, V., Sidike, P., Maimaitijiang, M., Carron, J., 2019. Urban tree species classification using a WorldView-2/3 and LiDAR data fusion approach and deep learning. *Sensors*, 19(6), 1284.

Honkavaara, E., Saari, H., Kaivosoja, J., Pölönen, I., Hakala, T., Litkey, P., Mäkynen, J., Pesonen, L., 2013. Processing and assessment of spectrometric, stereoscopic imagery collected using a lightweight UAV spectral camera for precision agriculture. *Remote Sensing*, 5(10), 5006–5039.

Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications. *ArXiv*, abs/1704.04861.

Jeronimo, S. M., Kane, V. R., Churchill, D. J., McGaughey, R. J., Franklin, J. F., 2018. Applying LiDAR individual tree detection to management of structurally diverse forest landscapes. *Journal of Forestry*, 116(4), 336–346.

Krizhevsky, A., Sutskever, I., Hinton, G. E., 2017. ImageNet Classification with Deep Convolutional Neural Networks. *Commun. ACM*, 60(6), 84–90.

Laurin, G. V., Liesenberg, V., Chen, Q., Guerriero, L., Del Frate, F., Bartolini, A., Coomes, D., Wilebore, B., Lindsell, J., Valentini, R., 2013. Optical and SAR sensor synergies for forest and land cover mapping in a tropical site in West Africa. *International Journal of Applied Earth Observation and Geoinformation*, 21, 7–16.

Li, Y., Zhang, X., Chen, D., 2018. CSRNet: Dilated Convolutional Neural Networks for Understanding the Highly Congested Scenes. *CoRR*, abs/1802.10062.

Lim, Y., La, H., Park, J., Lee, M., Pyeon, M., Kim, J.-I., 2015. Calculation of Tree Height and Canopy Crown from Drone Images Using Segmentation. *Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography*, 33(6), 605-613.

Lizarazo, I., Angulo, V., Rodríguez, J., 2017. Automatic mapping of land surface elevation changes from UAV-based imagery. *International journal of remote sensing*, 38(8-10), 2603–2622.

Long, J., Shelhamer, E., Darrell, T., 2014. Fully Convolutional Networks for Semantic Segmentation. *CoRR*, abs/1411.4038.

Maschler, J., Atzberger, C., Immitzer, M., 2018. Individual tree crown segmentation and classification of 13 tree species using airborne hyperspectral data. *Remote Sensing*, 10(8), 1218.

Mizoguchi, T., Ishii, A., Nakamura, H., Inoue, T., Takamatsu, H., 2017. Lidar-based individual tree species classification using convolutional neural network. *Videometrics, Range Imaging, and Applications XIV*, 10332, 193–199.

Mohan, M., Silva, C. A., Klauberg, C., Jat, P., Catts, G., Cardil, A., Hudak, A. T., Dia, M., 2017. Individual tree detection from unmanned aerial vehicle (UAV) derived canopy height model in an open canopy mixed conifer forest. *Forests*, 8(9), 340.

Rauhala, A., Tuomela, A., Davids, C., Rossi, P. M., 2017. UAV remote sensing surveillance of a mine tailings impoundment in Sub-Arctic conditions. *Remote sensing*, 9(12), 1318.

Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., Chen, L., 2018. Inverted Residuals and Linear Bottlenecks: Mobile Networks for Classification, Detection and Segmentation. *CoRR*, abs/1801.04381.

Santos, A. A. d., Marcato Junior, J., Araújo, M. S., Di Martini, D. R., Tetila, E. C., Siqueira, H. L., Aoki, C., Eltner, A., Matsubara, E. T., Pistori, H. et al., 2019. Assessment of CNN-Based Methods for Individual Tree Detection on Images Captured by RGB Cameras Attached to UAVs. *Sensors*, 19(16), 3595.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S. E., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2014. Going Deeper with Convolutions. *CoRR*, abs/1409.4842.

Tang, L., Shao, G., 2015. Drone remote sensing for forestry research and practices. *Journal of Forestry Research*, 26(4), 791–797.

Vigueras-Guillén, J. P., Sari, B., Goes, S. F., Lemij, H. G., van Rooij, J., Vermeer, K. A., van Vliet, L. J., 2019. Fully convolutional architecture vs sliding-window CNN for corneal endothelium cell segmentation. *BMC Biomedical Engineering*, 1(1), 4.

Wang, B., Ma, Y., Chen, G., Li, C., Dao, Z., Sun, W., 2016. Rescuing Magnolia sinica (Magnoliaceae), a critically endangered species endemic to Yunnan, China. *Oryx*, 50(3), 446–449.

Zhao, H., Shi, J., Qi, X., Wang, X., Jia, J., 2016. Pyramid Scene Parsing Network. *CoRR*, abs/1612.01105.