

## UTILISING SIMULATED TREE DATA TO TRAIN SUPERVISED CLASSIFIERS

P. Rönholm<sup>1,\*</sup>, S. Wittke<sup>1,2</sup>, M. Ingman<sup>1</sup>, P. Putkiranta<sup>1</sup>, H. Kauhanen<sup>1</sup>, H. Kaartinen<sup>2,3</sup>, M.T. Vaaja<sup>1</sup>,

<sup>1</sup> Department of Built Environment, Aalto University, P.O. BOX 14100, 00076 AALTO, Finland

<sup>2</sup> Department of Remote Sensing and Photogrammetry, Finnish Geospatial Research Institute, 02430 Masala, Finland

<sup>3</sup> Department of Geography and Geology, University of Turku, 20500 Turku, Finland

– (petri.ronholm, samantha.wittke, matias.ingman, pauli.putkiranta, heikki.kauhanen, matti.t.vaaja)@aalto.fi, harri.kaartinen@nls.fi

### Commission II, WG 6

**KEY WORDS:** convolutional neural network, YOLO, simulation, feature image, classification

### ABSTRACT:

The aim of our research was to examine whether simulated forest data can be utilized for training supervised classifiers. We included two classifiers namely the random forest classifier and the novel convolutional neural network classifier that utilizes feature images. We simulated tree parameters and created a feature vector for each tree. The original feature vector was utilised with random forest classifier. However, these feature vectors were also converted into feature images suitable for input into a YOLO (You Only Look Once) convolutional neural network classifier. The selected features were red colour, green colour, near-infrared colour, tree height divided by canopy diameter, and NDVI. The random forest classifier and convolutional neural network classifier performed similarly both with simulated data and field-measured reference data. As a result, both methods were able to identify correctly 97.5 % of the field-measured reference trees. Simulated data allows much larger training data than what could be feasible from field measurements.

### 1. INTRODUCTION

There is increasing demand for single tree inventory. With the higher resolution of available remote sensing datasets this becomes possible. Tree species classification is a main task for automatic tree inventory from remote sensing data. The species information can be utilized, for example, by the forest owner to confirm boundaries and quality as well as market values of a stand. Knowing about each single tree, also enables close monitoring of tree growth.

Tree species can be classified using different machine learning methods such as random forests and support vector machines (Raczko and Zagajewski, 2017) and fairly recently convolutional neural networks (Ferreira et al., 2020; Weinstein et al., 2019; Natesan et al., 2019; Nezami et al., 2020). In point cloud-based approaches, voxelization and waveform representations (Guan et al., 2015) have been utilized. In addition, support vector machines, random forests and neural networks (Deng et al., 2016) have been suggested for tree species classification from 3D point clouds. Despite the variety of approaches, the tree species classification problem remains an open research topic for single trees over a wide area

Since McCulloch and Pitts (1943) published ideas of neural networks, they have developed significantly. Early deep learning approaches (e.g. Ivakhnenko, 1971) with several layers were difficult to train, and deep learning did not become popular until the 2010s. Convolutional neural networks are specialized to handle grid-like data, such as images. Modern convolutional neural networks were introduced by Lecun et al. (1998). One significant step that made convolutional neural networks popular was AlexNet (Krizhevsky et al., 2012) that consists total of eight layers from which five are convolutional layers and the remaining three layers are fully connected.

Convolutional neural networks have become widely applied for object detection and image classification, seeing use particularly in databases with large numbers of parameters to be trained (Dhillon and Verma, 2020). In recent years, many applications of convolutional neural networks have been studied, such as detection of trees depending on age (Mubin et al., 2019), recognition of human actions in videos (Serrano et al., 2018), object classification for autonomous driving (Dreossi et al., 2017), and semantic segmentation of point clouds (Zhang et al., 2019). The architectures utilized for different applications are diverse, with varying input image sizes, and parameter numbers ranging from a few thousand to over a hundred million (Shin et al., 2016). With a large number of pre-trained networks available, transfer learning can be a viable approach (Gopalakrishnan et al., 2017), as a fine-tuned model can provide adequate results with a relatively small set of training data (Afridi et al., 2018).

Simulation is an alternative to field measurements for obtaining data to train supervised classifiers. Especially, neural networks require relatively large training datasets for the network to correctly recognize features, and real data may be difficult to obtain (Ødegaard et al., 2016). Simulated data can be used to get a large training dataset with varying features without need for field measurements (Ji et al., 2019). Simulated data has also been widely utilized to train convolutional neural networks in various applications, such as electron detection (van Schayck et al., 2020), ultrasound image enhancement (Perdios et al., 2018), and identification of fish species (Allken et al., 2019). A network trained with simulated data can potentially be directly applicable to real data (Nair et al., 2018), though real data may differ from the simulated data, and the network may need to be adjusted to address this difference (van Oort et al., 2019).

Recent tree species classification approaches typically utilise unmanned aerial vehicles (UAVs) and either RGB (e.g. Franklin, 2018; Natesan et al., 2019; Ferreira et al., 2020) or hyperspectral

\* Corresponding author

(Sothe et al., 2019) cameras or mobile laser scanners (Ramalho de Oliveira et al., 2021), or combinations of these (Colgan et al., 2012; Wu and Zhang, 2020). Lower-density airborne laser scanning data and imaging (Weinstein et al., 2019), as well as satellite imaging (Krahwinkler and Rossmann, 2013) and terrestrial laser scanning (Holmgren et al., 2008; Othmani et al., 2013) have also been applied.

The aim of this paper is to examine whether simulated forest data can be utilized for training supervised classifiers. In addition, we examine a novel classification method that utilises feature images and convolutional neural networks. As a comparison, we tested the random forest classifier with the same data. We conducted also a preliminary examination how well training with simulated data performed with reference data from field inventory.

## 2. MATERIALS AND METHODS

### 2.1 Simulating tree features

In order to experiment and compare classification methods, a simulated set of tree features was created. We selected three tree species, namely spruce, pine and birch, to be simulated. Simulation was implemented in Matlab.

The simulation allows to change the number of trees. The starting point is a trunk diameter at breast height (DBH). We set mean DBH, standard deviation as well as minimum and maximum DBH. From DBH, we derive tree height by utilizing the Näslund's height curve (Näslund, 1936):

$$h = \frac{DBH^m}{(b_0 + b_1 DBH)^m} + 1.3 \quad (1)$$

Parameter  $m$  depends on the tree species and is 2 for pine and birch, and 3 for spruce. Coefficients  $b_0$  and  $b_1$  also vary depending on tree species. We utilized mean parameters estimated in Siipilehto and Kangas (2015), presented in Table 1. We added some variation with given standard deviation to resulting heights. However, it was ensured that a tree could not exceed the characteristic maximum height of the corresponding tree species.

Table 1. The values of applied coefficients  $b_0$  and  $b_1$  (Siipilehto and Kangas, 2015).

	$b_0$	$b_1$
Spruce	1.635	0.330
Pine	1.195	0.242
Birch	0.898	0.242

The mean tree crown diameter of each tree species (Table 2) was taken from Korpela et al. (2014). Again, some variation with a given standard deviation was added.

Table 2. The values of applied mean crown diameters. (Korpela et al., 2014).

	crown diameter (m)
Spruce	2.9
Pine	2.8
Birch	2.6

We decided to simulate false colour, namely near infrared (NIR), red and green, values for each tree species. To add some challenge and imitate reality, we created two colour cases for each tree species corresponding to the cases where the treetop is

well illuminated and when it is in shadows. This is especially important for the convolutional neural network method because it is detecting patterns in images, and illuminated and shadowed cases look very different.

For simulation, we applied mean reflectance values (Table 3) from Korpela et al. (2014) that were converted into colour values by multiplying with 255 and then linearly enhanced according to real colour samples of a corresponding tree from an aerial image. We selected manually representative colours for both illuminated and shadowed cases for each tree species. The coefficients for linear transformation were found by regression (Figure 1 and Figure 2). In addition, we added some variation with given standard deviation to colours.

Table 3. The mean reflectance values for illuminated and shadowed canopies (Korpela et al., 2014).

	Red	Green	Near infrared (NIR)
Spruce, illuminated	0.032	0.047	0.224
Pine, illuminated	0.037	0.051	0.220
Birch, illuminated	0.043	0.060	0.322
Spruce, shadow	0.024	0.035	0.155
Pine, shadow	0.030	0.041	0.169
Birch, shadow	0.031	0.044	0.227

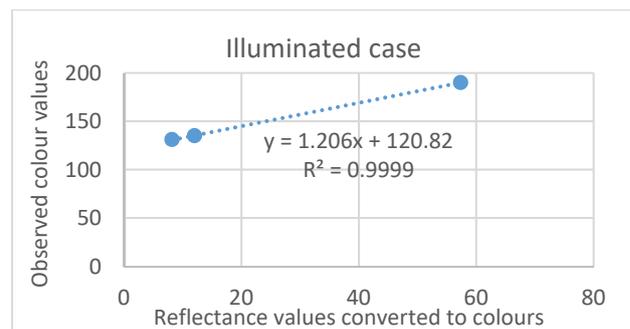


Figure 1. The regression for finding coefficients of a linear transformation in the illuminated case. The blue dots from left to right correspond to colours red, green and NIR, respectively.

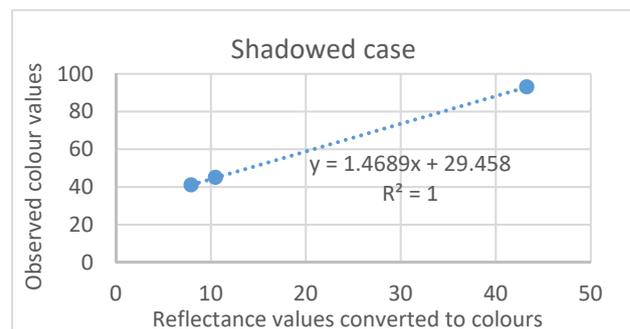


Figure 2. The regression for finding coefficients of a linear transformation in the shadowed case. The blue dots from left to right correspond to colours red, green and NIR, respectively.

The  $R^2$  values reveal that the relative ratios of applied reflectance values fit well with true R, G and NIR observations. Figure 3 illustrates simulated colour variations between tree species and compares them with a real sample of a pine from an aerial image. The simulation randomly selected if illuminated or shadowed case was created. In addition, a small variation with a given standard deviation was added to colour values. To get more descriptors for a feature vector, some derived measures were

utilized, such as tree height divided by crown diameter and the normalized difference vegetation index (NDVI):

$$NDVI = \frac{NIR-red}{NIR+red} \quad (2)$$

The final feature vector ( $f$ ) that was utilized for tree species classification was:

$$f = \begin{bmatrix} red \\ green \\ NIR \\ tree\ height/crown\ diameter \\ NDVI \end{bmatrix} \quad (3)$$

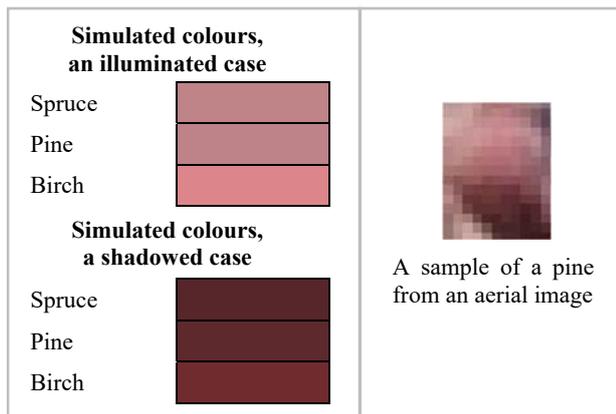


Figure 3. Simulated colour values and a sample of a pine taken from an aerial image.

## 2.2 Creation of feature images and labelling

In order to utilize convolutional neural networks, the feature vectors were converted into feature images (Figure 4). The size of the final feature image was decided on the basis of divisibility by 32, as that is the requirement for an unmodified YOLO v3 neural network. Therefore, the final size of our feature images was decided to be 96 x 96 x 3 pixels. Applying a colour image structure minimised the need to modify YOLO. First, we created an empty image of the size of 6 x 6 pixels. Then, the five values from a feature vector (eq. 3) were placed to area [2:4, 2:3] (last pixel was left empty). Two last elements were rescaled to the colour range [0, 255]. Finally, the images were scaled to the size of 96 x 96 pixels and the first layer was copied to two other layers. Figure 5 illustrates samples from resulting feature images for following six tree classes: illuminated spruce, shadowed spruce, illuminated pine, shadowed pine, illuminated birch, and shadowed birch.

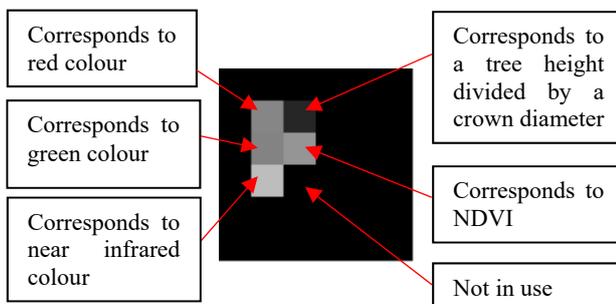


Figure 4. The areas of a feature image corresponds to selected feature elements.

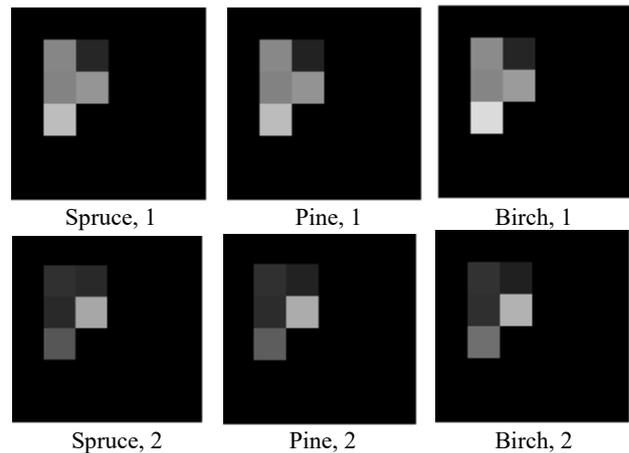


Figure 5. Examples of feature images. Number 1 corresponds to an illuminated case and number 2 corresponds to a shadowed case.

## 2.3 Simulated and field reference data

We created a training data set of 1000 trees per tree species. However, because this was divided randomly to the illuminated and shadowed cases, each sub-class did not have exactly 500 samples but close to it. Separate data sets were created for testing.

To test classification with real data, we applied data from forest inventory done in three forest sample plots in Evo, Finland. Inventory data was collected in 2014 as a joint effort by Finnish Geospatial Research Institute FGI and University of Helsinki. However, crown diameters were measured only from pines (*Pinus sylvestris*). Pine was the dominant tree species in the area. According to known locations and tree heights, trees were superimposed into an aerial image (UltraCam Eagle Mark 1 f100, year 2019). Red, green and near-infrared colours were attached to trees by selecting the median colour value from a small area within a tree area. Figure 6 illustrates the distribution and location of 40 reference pines in three inventory areas. Test plots included only few grown-up spruces and birches. Unfortunately, the crown diameter data of them were not available. However, the potential to detect spruces (*picea abies*) and birches (*betula sp.*) was examined by giving the expected crown diameter to them.

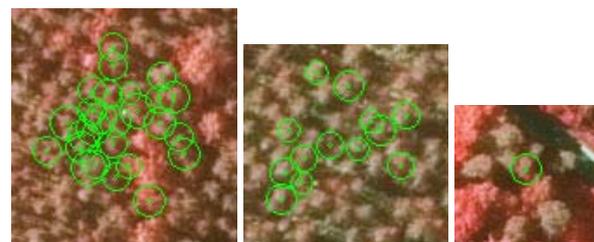


Figure 6. Reference trees superimposed into an aerial image.

## 2.4 Utilizing convolutional neural network

As a convolutional neural network, we selected YOLO (You Only Look Once) (Redmon et al., 2016) with the default full 106-layer yolov3 model (75 convolutional layers and 31 maxpool, route, up-sampling, and YOLO layers). Training was performed in the 64-bit c++ version of the open-source neural network Darknet (<https://pjreddie.com/darknet/>) in the Windows 10 operation system with GPU support. The pre-trained weights of yolov3.weights were applied as a starting point.

Training a convolutional neural network requires many iterations. However, at some point the model starts overfitting, i.e. the model performs much better on the training dataset than on the test dataset, hence does not generalize well. Therefore, the mean average precision (mAP) was followed to detect an optimal number of iterations. In our case, we found that the weights at 13000 iteration rounds performed the best. For testing the trained YOLO neural network, OpenCV libraries were utilized with Python.

### 2.5 Random forest classification

We decided to apply also the well-known random forest classifier (Breiman, 2001). Random forest classification is based on decision trees and has become a popular classification method. We utilized randomForest library in R (version 4.0.3). The random forest classifier was trained and tested with the same data set than YOLO. The “Number of Trees to Grow” parameter of the random forest classifier was set to 300.

## 3. RESULTS

In Table 4, the confusion matrices of classifications with both classifying methods are listed. In Table 5, illuminated and shadowed classes were combined into three main tree species classification. In Table 6, the overall accuracy and Cohen’s Kappa are presented for both classifiers. In Figure 7, we illustrate examples when YOLO has detected the feature image area and classified feature images to correct classes.

Table 4. Confusion matrices for YOLO v3 and Random Forest classifiers trained with 1000 reference tree for each tree species. The number 1 refers to an illuminated case and the number 2 to a shadowed case.

YOLO v3, N=3000							
truth \ predicted	spruce	spruce	pine	pine	birch	birch	
	1	2	1	2	1	2	
spruce1	475	0	30	0	0	0	
spruce2	0	494	0	1	0	0	
pine1	16	0	469	0	0	0	
pine2	0	0	0	515	0	0	
birch1	0	0	0	0	499	0	
birch2	0	0	0	0	0	501	
Random forest, N=3000							
	spruce	spruce	pine	pine	birch	birch	
	1	2	1	2	1	2	
spruce1	477	0	28	0	0	0	
spruce2	0	495	0	0	0	0	
pine1	23	0	462	0	0	0	
pine2	0	0	0	515	0	0	
birch1	0	0	0	0	499	0	
birch2	0	0	0	0	0	501	

Table 5. The results from Table 6, converted into three main tree classes.

YOLO v3			
truth \ predicted	spruce	pine	birch
spruce	969	31	0
pine	16	984	0
birch	0	0	1000
Random forest			
	spruce	pine	birch
spruce	972	28	0
pine	23	977	0
birch	0	0	1000

Table 6. The overall accuracy and Cohen’s Kappa (6 classes)

	Overall accuracy (%)	Cohen’s Kappa
YOLO	98.43	0.981
Random forest	98.30	0.980



Figure 7. Examples of correctly classified (YOLO v3) feature images.

In addition, both methods were applied to detect 40 reference pines that were measured in the field. Both methods succeeded to classify the same 39 reference trees correctly and classified one pine to the birch class leading to the 97.5% classification accuracy. However, to achieve this result the brightness of colours needed adjustment.

## 4. DISCUSSION

Applying 1000 simulated training samples per tree species seemed to work both with random forest and YOLO. Tests against simulated test data showed that the overall performance of tested methods was very similar even if there was some variation which trees were misclassified. When the classifiers tested with field-measured reference data, the colours needed adjustment. If we compare Figure 3 and Figure 6, it can be seen that there is a clear difference in colours, because the examples are taken from different images. To successfully identify pines only brightness needed to be adjusted to the level of training data.

Even if we had no crown diameter information of spruces and birches from the forest test plots, we examined them by giving expected canopy diameters to four spruces and seven birches. This examination revealed that the same brightness correction that worked with pines did not give satisfying results with other tree species. However, after adjusting brightness and stretching the histogram of individual colour layers, it was possible to classify all trees correctly. In practice, we needed to modify only near-infrared colours for spruces and green colours for birches to detect them correctly. This indicates that simulated data is a feasible way to get large training and testing data sets, but when a new aerial image is applied, colours need to be adjusted to meet the expected colour levels. This needs to be done separately to each colour layer and it might need both brightness adjustment and histogram stretching. Ideally, colour values should be changed to something that is invariant to changes in photographs. An alternative to simulation is data augmentation, if some training data is available. However, this is a topic for future research.

Converting feature vectors into feature images and applying convolutional neural networks for classification is an unconventional approach. In other words, by converting feature vectors into feature images, we enable YOLO to detect and classify a tree in focus. Actually, we would not need all the properties of convolutional neural networks, since localization at the image plane could be avoided. After all, the feature information is always located at the same place in feature images. In this experiment, we utilized only five features, and each feature filled a small area in a feature image. Therefore, convolution operations did not dismiss any features. If each feature would correspond to only one pixel in a feature image, it would be expected that the order of features might have influence to results, and the significance of features could vary. However, this is a topic for further research. In addition, in the future it would be interesting to compare the results of our way to utilize YOLO with the results of traditional fully connected multilayer neural networks.

We selected features that can be extracted from dense laser point clouds and photogrammetric images in practice. NVDI was added mainly to get larger pattern area in feature images. Since NVDI is derived from colour values, it most likely makes no difference to random forest classification. However, the effect for YOLO classification needs further research. The good performance of both classifying methods indicate that such simulation approach has potential to work in a practical forest inventory. However, we believe that the presented method can be further developed. We expect that simulation data can be improved by utilizing more species-wise knowledge about variation of the selected features. In addition, it might be possible to fine-tune training data for a specific classifier to improve the performance. For example, the convolutional neural network classifier might benefit if the training data set was larger than we applied. Our crown parameters came from field data, and it is a topic for future research how the method works with crown parameters extracted from laser scanning or photogrammetric data.

In our case, the use of a novel convolutional neural network classification method gave very similar outcome than the random forest classifier. The convolutional neural network classifier is expected to be scalable to handle many feature elements in a feature image, if needed. Theoretically, a feature image of the same size than we applied can hold 9216 features. Optimizing the structure of YOLO for operating with feature images is a good topic for further research. As an example, the current implementation of a full YOLO searches targets with different scales. However, the size of our feature image did not change. Therefore, a convolutional neural network could be modified to operate only with an optimal scale. In addition, the size of our feature images might not be optimal. However, this is not expected to affect to the classification accuracy, but merely to the computation speed.

## 5. CONCLUSIONS

The aim of our research was to examine whether simulated forest data can be utilized to train supervised classifiers. We created a simulator that can produce realistic tree feature vectors. These features were directly applied to a random forest classifier. In addition, we converted feature vectors into feature images suitable for a YOLO convolutional neural network. The random forest classifier and convolutional neural network classifier performed similarly both with simulated data and field-measured reference data. As a result, both methods were able to identify

correctly 97.5% of the field-measured reference trees. This indicates that simulation is a feasible method to train classifiers. Simulated data allows much larger training data than could be feasible from field measurements. However, the colour levels of images need to be adjusted to expected levels when a new image is applied. Using feature images and convolutional neural networks should scale well and the method is not limited to a single application.

## ACKNOWLEDGEMENTS

The authors would like to thank the Academy of Finland for support through the projects Profi5 Autonomous systems (No. 326246), Quality4Roads (No. 323783) and the Strategic Research Council project COMBAT – Competence-Based Growth Through Integrated Disruptive Technologies of 3D Digitalization, Robotics, Geospatial Information and Image Processing/Computing – Point Cloud Ecosystem (No. 293389, and 314312).

## REFERENCES

- Afridi, M.J., Ross, A., Shapiro, E.M., 2018. On automated source selection for transfer learning in convolutional neural networks. *Pattern recognition*, 73, pp. 65–75.
- Allken, V., Handegard, N.O., Rosen, S., Schreyeck, T., Mahiout, T., Malde, K., 2019. Fish species identification using a convolutional neural network trained on synthetic data. *ICES Journal of Marine Science*, 76(1), pp. 342–349.
- Breiman, L., 2001. Random forests. *Machine Learning*, 45(1), pp. 5–32.
- Colgan, M. S., Baldeck, C. A., Féret, J. B., Asner, G. P., 2012. Mapping savanna tree species at ecosystem scales using support vector machine classification and BRDF correction on airborne hyperspectral and LiDAR data. *Remote Sensing*, 4, pp. 3462–3480.
- Deng, S., Katoh, M., Yu, X., Hyypä, J., Gao, T., 2016. Comparison of Tree Species Classifications at the Individual Tree Level by Combining ALS Data and RGB Images Using Different Algorithms. *Remote Sensing* 8(12), 1034, 22 pages.
- Dhillon, A., Verma, G.K., 2020. Convolutional neural network: a review of models, methodologies and applications to object detection. *Progress in Artificial Intelligence*, 9, pp. 85–112.
- Dreossi, T., Ghosh, S., Sangiovanni-Vincentelli, A., Seshia, S. A., 2017. Systematic testing of convolutional neural networks for autonomous driving. *arXiv preprint arXiv:1708.03309*, 5 pages.
- Ferreira, M.P., de Almeida, D.R.A., de Almeida Papa, D., Minervino, J.B.S., Veras, H.F.P., Formighieri, A., Santos, C.A.N., Ferreira, M.A.D., Figueiredo, E.O. and Ferreira, E.J.L., 2020. Individual tree detection and species classification of Amazonian palms using UAV images and deep learning. *Forest Ecology and Management*, 475, 118397, 11 pages.
- Franklin, S. E., 2018. Pixel- and object-based multispectral classification of forest tree species from small unmanned aerial vehicles. *Journal of Unmanned Vehicle Systems*. 6(4), pp. 195–211.

- Gopalakrishnan, K., Khaitan, S.K., Choudhary, A., Agrawal, A., 2017. Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection. *Construction and Building Materials*, 157, pp. 322–330.
- 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), pp. 864–873.
- Holmgren, J., Persson, Å, Söderman U., 2008. Species identification of individual trees by combining high resolution LiDAR data with multi-spectral images, *International Journal of Remote Sensing*, 29(5), pp. 1537–1552.
- Ivakhnenko, A. G., 1971. Polynomial theory of complex systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 1(4), pp. 364–378.
- Ji, S., Shen, Y., Lu, M., Zhang, Y. 2019. Building Instance Change Detection from Large-Scale Aerial Images using Convolutional Neural Networks and Simulated Samples. *Remote Sensing*, 11, 1343, 20 pages.
- Korpela, I., Mehtätalo L., Markelin L., Seppänen A., Kangas A., 2014. Tree species identification in aerial image data using directional reflectance signatures. *Silva Fennica* vol. 48 no. 3, article id 1087, 20 pages.
- Krahwinkler, P. and Rossmann, J., 2013. Tree species classification and input data evaluation. *European Journal of Remote Sensing*, 46, pp. 535–549.
- Krizhevsky, A., Sutskever, I., Hinton, G. E., 2012. ImageNet classification with deep convolutional neural networks. In *NIPS: Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, pp. 1097–1105.
- Lecun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11), pp. 2278–2324.
- McCulloch, W.S., Pitts, W. A. 1943. Logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics* 5, pp. 115–133.
- Mubin, N.A., Nadarajoo, E., Shafri, H.Z.M., Hamedianfar, A., 2019. Young and mature oil palm tree detection and counting using convolutional neural network deep learning method. *International Journal of Remote Sensing*, 40(19), pp. 7500–7515.
- Nair, A.A., Gubbi, M.R., Tran, T.D., Reiter, A., Bell, M.A.L., 2018. A fully convolutional neural network for beamforming ultrasound images. In 2018 IEEE International Ultrasonics Symposium (IUS), pp. 1–4.
- Näslund, M., 1936: Skogsforsö ksastaltens gallringsforsök i tallskog. *Meddelanden från Statens Skogsforsöksanstalt*, 29, 169 pages.
- Natesan, S., Armenakis, C., Vepakomma, U., 2019. Resnet-based tree species classification using UAV images. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume XLII-2/W13, 2019 ISPRS Geospatial Week, 10–14 June 2019, Enschede, The Netherlands*, pp. 475–481.
- Othmani, A., Lomenie, N., Piboule, A., 2013. Region-based segmentation on depth images from a 3D reference surface for tree species recognition. In *IEEE International Conference on Image Processing*, September 2013, Melbourne, Australia. pp. 3399–3402.
- Perdios, D., Vonlanthen, M., Besson, A., Martinez, F., Ardit, M., Thiran, J. 2018. Deep Convolutional Neural Network for Ultrasound Image Enhancement. In 2018 IEEE International Ultrasonics Symposium (IUS), Kobe, Japan, pp. 1–4.
- Nezami, S., Khoramshahi, E., Nevalainen, O., Pölonen, I., Honkavaara, E., 2020. Tree species classification of drone hyperspectral and RGB imagery with deep learning convolutional neural networks. *Remote Sensing*, 12 (7), 1070, 20 pages.
- Raczko, E., Zagajewski, B., 2017. Comparison of support vector machine, random forest and neural network classifiers for tree species classification on airborne hyperspectral APEX images. *European Journal of Remote Sensing*, 50(1), pp. 144–154.
- Ramalho de Oliveira, L.F., Lassiter, H. A., Wilkinson, B., Whitley, T., Ifju, P., Logan, Stephen R.L., Peter, G.F., Vogel, J.G., Martin, T.A., 2021. Moving to Automated Tree Inventory: Comparison of UAS-Derived Lidar and Photogrammetric Data with Manual Ground Estimates. *Remote Sensing* 13(1), 72, 16 pages.
- Redmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: unified, real-time object detection, In *CVPR*, 2016, arXiv preprint, arXiv: 1506.02640, 10 pages.
- Serrano, I., Deniz, O., Espinosa-Aranda, J.L., Bueno, G., 2018. Fight recognition in video using hough forests and 2D convolutional neural network. *IEEE Transactions on Image Processing*, 27(10), pp.4787–4797.
- Shin, H.C., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., Summers, R.M., 2016. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, 35(5), pp.1285–1298.
- Siipilehto, J., Kangas, A. 2015. Näslundin pituuskäyry ja siihen perustuvia malleja läpimitan ja pituuden välisestä riippuvuudesta suomalaisissa talousmetsissä *Metsätieteen aikakauskirja* 4, pp. 215–236.
- Sothe, C., Dalponte, M., de Almeida C.M., Schimalski M.B., Lima C.L., Liesenberg V., Miyoshi G.T., Tommaselli A.M.G., 2019. Tree Species Classification in a Highly Diverse Subtropical Forest Integrating UAV-Based Photogrammetric Point Cloud and Hyperspectral Data. *Remote Sensing*. 11(11), 1338, 24 pages.
- van Oort, C.M., Xu, D., Offner, S.S., Gutermuth, R.A., 2019. Casi: A convolutional neural network approach for shell identification. *The Astrophysical Journal*, 880(2), 83, 14 pages.
- van Schayck, J.P., van Genderen, E., Maddox, E., Roussel, L., Boulanger, H., Fröjd, E., Abrahams, J.P., Peters, P.J., Ravelli, R.B., 2020. Sub-pixel electron detection using a convolutional neural network. *Ultramicroscopy*, 218, 113091, 10 pages.
- Weinstein, B.G., Marconi, S., Bohlman, S., Zare, A., White, E., 2019. Individual tree-crown detection in RGB imagery using

semi-supervised deep learning neural networks. *Remote Sensing*, 11, 1309, 13 pages.

Wu, Y., Zhang, X., 2020. Object-Based Tree Species Classification Using Airborne Hyperspectral Images and LiDAR Data. *Forests*, 11(1), 32, 25 pages.

Zhang, J., Zhao, X., Chen, Z., Lu, Z., 2019. A review of deep learning-based semantic segmentation for point cloud. *IEEE Access*, 7, pp.179118–179133.

Ødegaard, N., Knapskog, A.O., Cochin, C., Louvigne, J., 2016. Classification of ships using real and simulated data in a convolutional neural network. In 2016 IEEE Radar Conference (RadarConf), Philadelphia, United States. pp. 1–6.