Mapping wheel-ruts from timber harvesting operations using deep learning techniques in drone imagery

Saheba Bhatnagar[1]*, Stefano Puliti[1], Bruce Talbot[2], Joachim Heppelmann[1], Johannes Breidenbach[1]*, Rasmus Astrup[1]

[1] *Division of forest and forest resources, Norwegian Institute of Bioeconomy Research, Norway*

[2] *Department of forest and wood science, Stellenbosch University, South Africa*

*corresponding authors: saheba.bhatnagar@nibio.no; johannes.breidenbach@nibio.no

Mapping wheel-ruts from timber harvesting operations using deep learning techniques in drone imagery

Wheel ruts, i.e. soil deformations caused by harvesting machines, are especially harmful because they can impact growth conditions of future forest generations and should therefore be avoided or ameliorated. However, the mapping of wheel ruts that would be required in monitoring harvesting operations and planning amelioration measures is a tedious and time-consuming task. We used a deep-learning image-segmentation method (ResNet50 + UNet architecture) that was trained on drone imagery acquired shortly after harvests in Norway, where more than 160 km of wheel ruts were manually digitized. The leave-one-out cross-validation of 20 harvested sites resulted in F1-scores of 0.45-0.83 with an average of 0.67. The highest accuracy was obtained for severe wheel ruts (average user's accuracy (UA) = 0.74), and the lowest accuracy was obtained for light wheel ruts (average UA = 0.62). Besides rut severity, the accuracy was also affected by the spatial resolution and noise present at the site. In combination with the ubiquitous availability of drones, the results of our study have the potential to greatly reduce the environmental impact of final felling operations by enabling the automated mapping of wheel ruts.

Keywords: UAV, soil displacement, image segmentation, CNN, Neural networks

Introduction

Mechanized harvesting of forests is an integral part of sustantiable forest management and required to supply society with the required timber through efficient and safe forest operations. However, mechanized harvesting operations can constitute considerable environmental impacts. Besides the inevitable but temporary loss of habitat for forest-dwelling animal and plant species, harvesting operations can result in soil damage (Ampoorter et al. 2010). Wheel ruts are a form of soil displacement caused by wheels or crawler tracks of forest machines that compress and shear upper soil layers. Despite considerable technological developments to reduce the impact on soils, wheel ruts can occur when forest operations cannot be conducted under suitably low soil moisture conditions. The porosity of soils affected by wheel ruts is may be reduced, resulting in anaerobic processes, and reducing the stability and growth of future forests. Therefore, wheel ruts should be avoided as far as possible, and forest certification schemes and regulations includes thresholds for acceptable levels of wheel rutting. Hence, it is important to have efficient ways to monitor the amounts of wheel rutting following harvests.

Because of the large areas involved, rugged terrain, and presence of harvest residues, it is challenging to map wheel ruts in the field, and the use of remotely sensed data may present a feasible alternative (Talbot and Astrup 2021). Drones or unmanned aerial vehicles (UAVs) have become popular for capturing images in many forest-related applications (Puliti et al. 2015, Ighlaut et al. 2019, Kentsch et al. 2020, Banu et al. 2016). In the field of forest operations, drone images and derived 3D products provide a useful source of information to assess the environmental performance of the harvesting operation (Talbot and Astrup 2021). In particular, drones have been used to measure wheel rut depth (Pierzchala,Talbot and Astrup 2014, 2016, Haas et al. 2016, Talbot et al. 2018, Marra et al. 2021). These studies provided insights in the obtainable accuracy of the rut depth measurements from drone imagery.

Nevertheless, all the above studies required manual intervention in either identifying the trail network or localizing specific measurement points or profiles for further analysis. In an effort to automate such measurements, Nevalainen et al. (2017) developed a method to delineate wheel ruts and measure their depth based on UAV imagery. While providing a certain degree of automation, such a method is limited since it heavily relies on the presence of trees around the wheel ruts, which are used to mask out the track area. Such conditions are rarely met in clear cut areas, characterized by a mix of open ground and harvest residuals.

A more widespread operational deployment of UAV based post-harvest assessment would require the partial or full automation of wheel rut detection and measurement (Talbot and Astrup, 2021). Popular techniques for the analysis of drone data comprise support vector machines (SVM) (Boser et al. 1992) and random forest (Breiman 2001), but these also require human intervention for feature extraction and therefore rely heavily on domain knowledge (Liu & Lang 2019). Deep learning (DL) algorithms automatically learn the features from the data (lazy learning), enabling automation for broad applications, and often outperform traditional algorithms (Zou et al. 2015; Wurm et al. 2019; Bhatnagar et al. 2020). This is also aided by more sophisticated DL libraries and complementary hardware to process the data. For remote sensing applications, convolution neural networks (CNNs) have become a popular choice (Ma et al. 2019). CNNs are utilized mainly for classification in two ways – scene annotation and semantic segmentation. In the case of scene annotation, the output of the CNN is the 1-dimensional (1D) vector defining the probability of the image belonging to a particular label. Whereas in semantic segmentation, every pixel is labelled, i.e., the output is not a 1D probability but a 2-dimensional (2D) score map. Therefore, in semantic segmentation, some fully connected layers are replaced by fully convolutional layers. There are multiple new and off-the-shelf architectures present, which have been successfully applied in remote sensing for image classification already (Shin et al. 2016).

While we are not aware of studies on the automated detection of wheel ruts using DL, CNNs have been applied for urban mapping (Audebert et al. 2018), like the identification of roads (Bayoudh et al. 2021), cracks in surfaces (Kim et al. 2021; Ali et al. 2021), tracks (Giben et al. 2015), and pavements (Ma et al. 2021). In addition, studies like Kanakaraddi et al. (2021), Patil & Jadhav (2021), Sofia et al. (2021) have depicted the usage of CNNs to detect roads using satellite imagery. Zhang et al. (2018) describe the benefit of combining ResNet with UNet to extract roads from aerial images.

This study aims to automate the detection and segmentation of wheel ruts caused by cut-to-length harvesters and forwarders in drone images of previously tree-covered sites acquired shortly after final harvests. We use CNN models to segment wheel ruts and cross-validate our results using 20 independent harvested sites with areas between 0.5-21.5 ha in south-eastern Norway.

Material and methods

The processing workflow (Figure1) consisted of the five steps 1) Capturing the drone imagery. 2) Manual annotating the drone images (digitized as polyline vectors). 3) Pre-processing of the images, which includes rasterization of the wheel ruts vectors, and splitting the imagery for feeding into the DL system. 4) Semantic segmentation to detect wheel and non-wheel rut areas in all images per site. 5) Post-processing of the predicted results, including mosaicking the images and applying morphological operations on the mosaicked images. Steps 4 and 5 were repeated for all sites in the leave one out cross-validation format.

Study sites and drone data

A total of 20 study sites were available that were imaged from a drone after clear cutting (Figure 2a). The flights were conducted over a span of four years (2016 – 2019) as part of a long-term effort to monitor the environmental performance of modern harvesting practices. A full description of the manually annotated data is provided by Heppelmann et al. 2021. All sites were productive forest areas in south-eastern Norway. The availability of the sites was determined in cooperation with forest owners commissioning harvests in several research projects for which the drone data were collected.

The drone data acquisitions varied with respect to several parameters, including the camera used for the image acquisition, flight altitude, season, date and time of the day (Table 1).

The sites captured in the initial part of the acquisition period were done so using a DJI Phantom 2 UAV fitted with a GoPro^{TM} Hero 4 12 mega-pixel camera (p2GoPro) (DJI 2013), which was later replaced with a DJI Phantom 4 Pro UAV, with DJI's factory fitted 20 mega-pixel camera (p4pro) (DJI 2020). Survey flights with given altitudes and overlap were conducted using DJI's Ground Station Pro software (DJI 2021) in most cases, although UgCS software (www.ugcs.com) was used on steeper sites to reduce variation in GSD within the same model. UgCS allows for flight paths to be set to follow the terrain form from a given digital terrain model resulting in a constant flight height above ground. For the lower resolution p2GoPro, a flight altitude of approximately 50 m above the ground was targeted, while this was increased to between 60 and 100 m on the p4pro, both depending on terrain and obstacles. Despite only low-to-ground objects being of interest, a high forward overlap of 80% and lateral overlap of 70% was targeted in the flight plan. On each site, 5-7 ground control points (GCPs) were installed before image capturing. The GCP position was recorded at centimetre accuracy using a TopCon GR-5 real-time kinematic (RTK) GNSS with live correction via the GSM network.

Agisoft Photoscan was used in processing all UAV based image data (Agisoft, 2019). This process includes feature detection, image alignment, depth mapping, the generation of sparse and dense point clouds, a textured mesh, and finally, digital elevation model (DEM) and ortho-mosaic creation. Depth mapping and the accuracy of that is carried out using structure-from-motion (SfM), as described by Iglhaut et al. (2019). The GCPs, which were clearly visible in the imagery, were used in optimizing camera pose estimates during the SfM procedure in Agisoft, resulting in DEMs with an RMSE typically < 5 cm on average. The finest common resolution on each site was used in generating the orthomosaics and DEMs, which was typically the same as the GSD of 1-3 cm.

Data annotation

The ortho-mosaics for all 20 sites were manually annotated in a GIS environment to register the location, frequency, and severity of wheel ruts (Figure1, step 2). Due to the removal of most trees (i.e., clear cut) during the harvesting operations, the tracks were clearly visible in the RGB ortho-mosaics. In addition to the RGB information, the UAV DEM were also used to aid the annotation of the UAV data (Heppelmann et al. 2021). The tracks were digitized as polylines, and each polyline was classified into the following three severity classes (Figure *2*c).

- Light: visible tracks with no identifiable soil displacement or rut-formation.
- Moderate: tracks that showed rutting with minor soil displacement and deeper indentations but no visible loss of water drainage functions.
- Severe: all tracks with either substantial soil displacement, deep indentations, loss of water drainage functions, or a combination of various of these factors.

Preparation of the annotated data for deep learning

The aim of the study was the segmentation of wheel ruts independent of their severity category (Figure1, step 3). The input reference observation for modelling was thus a binary annotated image with wheel ruts as one class and unaffected area as a second class. Such an image was generated by applying a three meters buffer around the annotated line shapefile, dissolving the results, and converting the resulting polygon into a binary raster with a value of one in correspondence to tracks and zero to non-tracks (Figure 3 b,c). The three-meter buffer around the annotated line shapefile was selected after a visual assessment of the affected area. Additionally, any non-forest area was masked out from the further analysis.

The input to the deep learning model (next section) is RGB images in portable network graphics (png) format with a size of 210 × 210 pixels. Therefore, the RGB GeoTIFF raster and the annotated data were split into tiles of 210 × 210 pixels and converted to png format. The metadata containing the geotags of the tiles were stored and used later for mosaicking the predictions. All images were manually checked for quality before feeding the images into the model, and completely blurred or distorted images were removed. That is, from the total of 1909 images, 12 were removed such that a total of 1897 images were available for model training. However, for testing the model, all the images were used, resulting in a total of 1909 predicted images. This was done such that continuous mosaicked can be formed without any gaps. Depending on the size of the sites being used for training and testing, the amount of training data varied. The smallest site (site A) consisted of 96 images, and 1801 images were available for training. The biggest site (site T) consisted of 423 images, and 1474 images were available for training (Table 1, Section 3.2).

Semantic segmentation using CNN

For semantic segmentation (Figure1, step 4), the choice of architecture is generally application-specific, and each architecture has advantages and disadvantages in accuracy, memory consumption, operation counts, inference time and parameter utilization (Canziani et al. 2016). Preliminary analysis on a subset of this study’s data revealed that among several combinations of network architectures, the combination of

ResNet50 and UNet provided the best result, and we thus selected this combination for further analysis.

Figure 4 shows the architecture used in this study. The ResNet50 architecture is well resilient to overfitting due to its residual learning concept (Yang et al. 2020), which states that each layer will feed to next layer and also to the activation layer directly. The layers are considered residual blocks to facilitate the network's training (Ardakani et al. 2020). For decoding the information from ResNet50, UNet architecture is used. The UNet retains information while upsampling to circulate context from lower to a higher resolution layer (Alam et al. 2021). Due to the availability of limited data, a transfer of pre-trained weights from ImageNet was applied on ResNet50, and UNet (decoder) was trained from scratch. Image augmentation (flipping, rotation) was also used to increase the number of input images.

The ResNet50 is a deep network having 50 layers, including batch normalization layers. Such layers normalize the nodes before inputting them into the following activation function. The architecture uses skip connections to impart information between the layers. The convolutional layer is matrix multiplication over the images using a filter of size 3 x 3 with stride = 2 (Figure 4).

The activation function (A) is used to introduce nonlinearity in the input images, which is done to make the model more expressive and sensitive to distinguish minute features. The Rectified Linear unit (ReLu) activation function we used in this study is one of the most used activation functions due to its high computational effectiveness and computing speed (Lu et al. 2017, Bircanoğlu & Arıca 2018). ReLu removes all the negative parts from the input (image f), as described in ‘Equation (1)’.

$$A(f) = 0;\ f < 0 \qquad (1)$$

$$A(f) = f;\ f \geq 0$$

where A(f) is a picture element of f. Apart from ReLu, for classification, a Softmax classifier for calculating probabilities was used as a top layer for pixel-wise prediction. To extract the most important features (e.g. sharp and smooth features), pooling layer was used. Here, we use max pooling layer, where only the local maxima of the region under the filter was carried forward.

Apart from the choice of architecture, the choice of hyper-parameters also plays a vital role in the performance of the CNN model. Only a subset of data was used for tuning the model for efficient and faster processing. The aim of optimization is to minimize the cross-entropy loss this was done using adaptive momentum (ADAM) optimization. To ensure augmentation does not change the quality of the test results, the L2 (or ridge) regularisation method was used. An initial learning rate (the rate at which weights are updated) of 0.01 was selected by the hit and trial method. The batch size was set to 20. For upsampling, the UNet architecture trained from scratch was applied. UNet uses both transpose convolution and skip connections to sync the feature maps from the encoder and decoder.

The proposed methodology was run using a 16 GB Precision 5820 Tower Workstation with Intel® Xeon™ processer and Ubuntu® Linux operating system. The deep learning model was run in a leave-one-out cross-validation scheme for all 20 sites by omitting one site at a time to test. The model training took approximately 50 hours for 20 epochs per site. Hence, the entire process for 20 sites was finished in approx. 45 days. An increase in the number of epochs did not markedly improve the precision (see Section 3).

Post-processing

The predictions were made for all the images from all the sites, including the images which were discarded for training the model, such that the ortho-mosaic can be calculated for the entire area. The predictions were further enhanced to locate the wheel ruts using multiple morphological operations. In the post-processing, the noise elements were removed without distorting the original results. First, a binary opening was performed. Opening in mathematical morphology is defined as an erosion followed by dilation using the same structuring element (SE) or kernel on the image (f), shown in 'Equation (2)'. The aim was to enhance detected wheel ruts. As a result, all the spurious regions smaller than ~5 m^2 were removed.

$$F(f, SE_1) = (f \ominus SE_1) \oplus SE_1 \qquad (2)$$

where F was the opened-binary image, $\ominus$ is erosion, and $\oplus$ dilation (Serra 1979). The SE_1 was a circular disk of a radius of 2 m.

Second, grayscale erosion was performed after binary opening. This step replaces each pixel with the local minimum of the defined SE (SE2) around the pixel. This was done to define the wheel ruts accurately and remove overestimation along the boundaries of the ruts. The circular disk also uses erosion in this study was set to 20 cm. For the example of a spatial resolution of 1cm, SE2 was a matrix of size 10 x 10.

Validation

We used leave-one-out cross-validation, and in each iteration, the following steps were implemented:

(1) Splitting the data into training (number of sites minus one = 19) and testing data (one site).

(2) Training the model on the training data.

(3) Applying the model to the testing site to classify wheel ruts.

(4) Computing the confusion matrix of the manually annotated vs the model predictions.

Additionally, the training overall accuracy (OA - accuracy of CNN model applied on training data; ‘Equation (3)’), testing OA (accuracy of CNN model applied on the testing data), F1-score and proportion of wheel rut detected ‘Equation (4)’ user’s accuracy (UA) and producer’s accuracy (PA) were calculated ‘Equation (5)-(6)’.

$$OA = \frac{TP+TN}{TP+FP+FN+TN} \tag{3}$$

$$F1 = \frac{2TP}{2TP+FP+FN} \tag{4}$$

$$UA = \frac{TP}{TP+FN} \tag{5}$$

$$PA = \frac{TP}{TP+FP} \tag{6}$$

where, TP = true positives, TN = true negatives, FP = false positives, FN = false negatives. An overall confusion matrix was calculated by adding the TP, TN, FP and FN pixels from every site. Relative confusion matrices were calculated by dividing each cell (TP, TN, FP, FN) by the total number of pixels.

Results and discussion

A comprehensive post-harvest assessment of soil disturbance for compliance with management objectives is a resource-demanding exercise, hardly justifiable under current economic or regulatory conditions. Talbot et al. (2018) show how manual in-field measurement would require between 10 and 20 transects of 50 m each (depending on rut prevalence) per hectare to estimate rut lengths with a mean error below 10%.

Therefore, the use of UAV derived data is an attractive option for forest regions applying clear cutting or shelterwood regimes where the ground is partially or fully unobscured, including most of the managed boreal forest and all forms of plantation forestry. However, none of the studies carried out with UAVs to date has presented possibilities for fully automating the data analysis process. Hence, a study using deep learning to detect wheel ruts in a harvested forest automatically was performed. For this, ResNet50+UNet architecture was used, and the study was verified for 20 forest sites in Norway. For every site, training OA, testing OA, and other results achieved at the end of the 20th epoch, are described in Table 2.

On average, the cross-validation test OA was 73%, with an F1 score of 0.67 for the wheel ruts. Morphological operations such as area opening and erosion were performed in the predicted ortho-mosaics, leading to more refined wheel ruts' maps. The post-processed imagery was compared against the original labels. Table 3 gives an overall confusion matrix (containing data from all pixels in all of the 20 sites), and the confusion matrix for all locations is shown in supplementary data (S.1). Compared to original CNN predictions, the testing OA increased by an average of 1.2% after post-processing. Due to the area-opening technique, spurious regions were removed, increasing FN for wheel ruts. The next step, erosion of the wheel ruts, made the wheel-ruts more defined, increasing the TP (for the wheel ruts), leading to an increase in testing OA.

The severe wheel ruts were best detected with an average UA of ~74% (Figure 5). Since the predictions were made only for wheel rut and non-wheel rut, and the severity was not predicted, the PA is not provided. The moderate and light wheel ruts were often confused with the background and were detected with ~67% and 62% average UA, respectively. The light wheel ruts, representing 63.6% of annotated tracks, were mostly shallow and spectrally similar to the background, resulting in poorer detection accuracy. This category poses little harm to the soil, and therefore, a lower detection probability is not alarming.

The moderate wheel ruts consisted of 24.6% of annotated tracks. The wheel ruts are also different in various locations; for example, mainly moderate ruts were present in site J, and no severe. Moderate wheel ruts show no visible water drainage but can still have noticeable soil displacement. Therefore, they are detected better than light wheel ruts. For some sites, like site C, the severe and moderate wheel ruts look similar, increasing FP for moderate wheel ruts (S.2).

The severe wheel ruts generally have reduced water drainage functions and considerable soil displacement, making them unique and identifiable. We found that despite the severe class was representing only 11.8% of the total length of the annotated tracks, it was the one detected with the largest accuracy. Even though severe ruts often represent a small portion of the area in a clear cut, they are key in determining the environmental performance of harvest operations for certification purposes. In this sense, our results are encouraging as they show that the most important class is the one that is best predicted.

The average spatial resolution of the images was ~2 cm, and for sites with finer or coarser resolution, the detection was poorer. For example, only 20% of the wheel rut was detected in site N, with a spatial resolution of 7.1 cm. For the sites with a spatial resolution of approx. 2cm (7 of 20 sites), the wheel ruts were detected with an average accuracy of 71%; whereas, for the sites with a spatial resolution of 1cm (5 of 20 sites), it was 66%.

Figure 6 shows two test cases, site O being one of the best detected sites with an F1 score of 0.83, and site N, one of the poorest, with an F1 score of 0.45. Maps of detected and annotated wheel ruts, including confusion matrices for all sites, can be found in the Supplementary material (S.1 and S.2).

Apart from the actual wheel rut, the noise present in the form of residual logs, branches, harvest residues, and shadows can also increase the FP leading to over-estimation of wheel ruts. For example, the site I and E has low testing OA (66% and 54% respectively) mostly due to the noise present on the site.

There are also two types of sensors that were used to capture the drone images. Although the images from p4pro and p2GoPro had differences in the technical specifications (example: shutter mechanism, sensor size, focal length) and the size of the training data (75% p4pro, 15% p2GoPro), there was no notable difference in the accuracy of detecting wheel ruts for the sites from either sensor (p4pro testing OA ~64%, p2GoPro testing OA ~73%). The proposed model was also robust and applicable across the different sensors regardless of their different RGB colour profile (S.3) and sensor specifications.

To check if there is any bias due to imbalance of data between cameras, a separate model was run only on images from p4pro. Figure 7 depicts the accuracy metrics for the CNN model using images from only p4pro, tested on site P. The testing site was chosen randomly, and the total number of training images used was ~900. This test was carried out for 50 epochs to see the effect of increasing epochs on the testing OA (Figure 7).

At epoch 20, in comparison to site P (trained using all data), site P trained using only p4pro data has slightly better testing accuracy but a lower wheel rut F1 score (Table 2). At the end of 20th and 50th epochs, the training accuracy was 99.3% and 99.8%, respectively. This process took ten days to run. Compared to the 20th epoch, there is 2% rise in training accuracy and an increase of 0.04 in F1 score for predicting wheel rut. This means, by increasing the number of epochs, there is a slight improvement in the model's performance. However, it is important to consider the time constraint, and therefore running the model for 20 epochs was deemed acceptable.

Lastly, removing the data from p2GoPro made no notable difference in the model's performance. This also implies that, due to the usage of transfer learning, the proposed model works well even with a smaller amount of training data. This is particularly helpful when working in a new area with limited images, which is often the case in practical applications.

Conclusions

In this study, ResNet50+UNet architecture was deployed to identify wheel ruts in drone images of harvested forest sites. The research was conducted on 20 sites, and an average of 73% of testing OA was achieved, with the highest testing OA of 80%. Based on the results obtained, the following conclusions were drawn:

(1) Drone-based photogrammetry and deep learning are valuable techniques for detecting the presence of wheel ruts.

(2) Severe wheel ruts were most accurately identified due to their prominence in appearance compared to light wheel ruts that are homogenous to the surroundings.

(3) The proposed model is robust and can be used to identify wheel ruts from multiple sensors captured at different times.

The application of the proposed method can provide an efficient avenue for monitoring and mapping the environmental impact of harvest operations which in turn may lead to better overall environmental performance of harvest operations. The final CNN trained with all 20 sites and programming code are available at [after acceptance of the paper].

Acknowledgements

This work was supported by the Bio Based Industries Joint Undertaking under the European Union's Horizon 2020 research and innovation program, TECH4EFFECT- Knowledge and Technologies for Effective Wood Procurement [grant number 720757], and The Norwegian Institute for Bioeconomy Research, NIBIO.

References

Alam, M., Wang, J. F., Guangpei, C., Yunrong, L. V., & Chen, Y. (2021). Convolutional Neural Network for the Semantic Segmentation of Remote Sensing Images. Mobile Networks and Applications, 26(1), 200-215.

Ali, L., Alnajjar, F., Jassmi, H. A., Gochoo, M., Khan, W., & Serhani, M. A. (2021). Performance Evaluation of Deep CNN-Based Crack Detection and Localization Techniques for Concrete Structures. Sensors, 21(5), 1688.

Ampoorter, E., Van Nevel, L., De Vos, B., Hermy, M., & Verheyen, K. (2010). Assessing the effects of initial soil characteristics, machine mass and traffic intensity on forest soil compaction. Forest Ecology and Management, 260(10), 1664-1676.

Ardakani, A. A., Kanafi, A. R., Acharya, U. R., Khadem, N., & Mohammadi, A. (2020). Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks. Computers in biology and medicine, 121, 103795.

Astrup, R., Nowell, T., Talbot, B. (2021) A personal laser scanning system for mapping soil disturbance: system development and evaluation against manual and photogrammetry based measurements. International Journal of Forest Engineering. [under review]

Audebert, N., Le Saux, B., & Lefèvre, S. (2018). Beyond RGB: Very high resolution urban remote sensing with multimodal deep networks. ISPRS Journal of Photogrammetry and Remote Sensing, 140, 20-32.

Banu, T. P., Borlea, G. F., & Banu, C. (2016). The use of drones in forestry. Journal of Environmental Science and Engineering B, 5(11), 557-562.

Bayoudh, K., Hamdaoui, F., & Mtibaa, A. (2021). Transfer learning based hybrid 2D-3D CNN for traffic sign recognition and semantic road detection applied in advanced driver assistance systems. Applied Intelligence, 51(1), 124-142.

Bhatnagar, S., Gill, L., & Ghosh, B. (2020). Drone Image Segmentation Using Machine and Deep Learning for Mapping Raised Bog Vegetation Communities. Remote Sensing, 12(16), 2602.

Bircanoğlu, C.; Arıca, N. A comparison of activation functions in artificial neural networks. In Proceedings of the 2018 26th Signal Processing and Communications Applications Conference (SIU), Izimir, Turkey, 2–5 May 2018 Institute of Electrical and Electronics Engineers (IEEE): Piscataway, NJ, USA, 2018; pp. 1–4.

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992, July). A training algorithm for optimal margin classifiers. In Proceedings of the fifth annual workshop on Computational learning theory (pp. 144-152).

Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

Canziani, A., Paszke, A., & Culurciello, E. (2016). An analysis of deep neural network models for practical applications. arXiv preprint arXiv:1605.07678.

Giben, X., Patel, V. M., & Chellappa, R. (2015). Material classification and semantic segmentation of railway track images with deep convolutional neural networks. In 2015 IEEE International Conference on Image Processing (ICIP) (pp. 621-625). IEEE.

Haas, J., Ellhöft, K. H., Schack-Kirchner, H., & Lang, F. (2016). Using photogrammetry to assess rutting caused by a forwarder—A comparison of different tires and bogie tracks. Soil and Tillage Research, 163, 14-20.

Heppelmann J.B., Talbot B., Fernández C., Astrup,R. (2021). Depth-to-water maps as predictors of rut severity in fully mechanized harvesting operations. International Journal of Forest Engineering. [under submission]

Iglhaut, J., Cabo, C., Puliti, S., Piermattei, L., O’Connor, J., & Rosette, J. (2019). Structure from motion photogrammetry in forestry: A review. Current Forestry Reports, 5(3), 155-168.

Kanakaraddi, S. G., Chikaraddi, A. K., Pooja, B. L., & Preeti, T. (2021). Detection of Roads in Satellite Images Using Deep Learning Technique. In ICT Analysis and Applications (pp. 441-451). Springer, Singapore.

Kentsch, S., Lopez Caceres, M. L., Serrano, D., Roure, F., & Diez, Y. (2020). Computer vision and deep learning techniques for the analysis of drone-acquired forest images, a transfer learning study. Remote Sensing, 12(8), 1287.

Kim, B., Yuvaraj, N., Preethaa, K. S., & Pandian, R. A. (2021). Surface crack detection using deep learning with shallow CNN architecture for enhanced computation. Neural Computing and Applications, 1-17.

Liu, H., & Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: A survey. applied sciences, 9(20), 4396.

Lu, H., Fu, X., Liu, C., Li, L. G., He, Y. X., & Li, N. W. (2017). Cultivated land information extraction in UAV imagery based on deep convolutional neural network and transfer learning. Journal of Mountain Science, 14(4), 731-741.

Ma, D., Fang, H., Wang, N., Xue, B., Dong, J., & Wang, F. (2021). A real-time crack detection algorithm for pavement based on CNN with multiple feature layers. Road Materials and Pavement Design, 1-17.

Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G., & Johnson, B. A. (2019). Deep learning in remote sensing applications: A meta-analysis and review. ISPRS journal of photogrammetry and remote sensing, 152, 166-177.

Marra, E., Wictorsson, R., Bohlin, J., Marchi, E., & Nordfjell, T. (2021). Remote measuring of the depth of wheel ruts in forest terrain using a drone. International Journal of Forest Engineering, 1-11.

Nevalainen, P., Salmivaara, A., Ala-Ilomäki, J., Launiainen, S., Hiedanpää, J., Finér, L., & Heikkonen, J. (2017). Estimating the rut depth by UAV photogrammetry. Remote Sensing, 9(12), 1279.

Patil, D., & Jadhav, S. (2021). Road Extraction Techniques from Remote Sensing Images: A Review. Innovative Data Communication Technologies and Application, 663-677.

Pierzchała, M., Talbot, B., & Astrup, R. (2014). Estimating soil displacement from timber extraction trails in steep terrain: application of an unmanned aircraft for 3D modelling. Forests, 5(6), 1212-1223.

Pierzchała, M., Talbot, B., & Astrup, R. (2016). Measuring wheel ruts with close-range photogrammetry. Forestry: An International Journal of Forest Research, 89(4), 383-391.

QGIS Development Team, 2020. QGIS Geographic Information System. Open-Source Geospatial Foundation Project. http://qgis.osgeo.org

Serra, J. (1979). Biomedical image analysis by mathematical morphology (author's transl). Pathologie-biologie, 27(4), 205-207.

Shin, H.C., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D. and 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.

Talbot, B., & Astrup, R. (2021). A review of Sensors, Sensor-Platforms and Methods Used in 3D Modelling of Soil Displacement after Timber Harvesting. Croatian Journal of Forest Engineering: Journal for Theory and Application of Forestry Engineering, 42(1), 149-164.

Talbot, B., Rahlf, J., & Astrup, R. (2018). An operational UAV-based approach for stand-level assessment of soil disturbance after forest harvesting. Scandinavian Journal of Forest Research, 33(4), 387-396.

Wurm, M., Stark, T., Zhu, X. X., Weigand, M., & Taubenböck, H. (2019). Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks. ISPRS journal of photogrammetry and remote sensing, 150, 59-69.

Yang, P., Dong, C., Zhao, X., & Chen, X. (2020). The Surface Damage Identifications of Wind Turbine Blades Based on ResNet50 Algorithm. In 2020 39th Chinese Control Conference (CCC) (pp. 6340-6344). IEEE.

Zhang, Z., Liu, Q., & Wang, Y. (2018). Road extraction by deep residual u-net. IEEE Geoscience and Remote Sensing Letters, 15(5), 749-753.

Zou, Q., Ni, L., Zhang, T., & Wang, Q. (2015). Deep learning based feature selection for remote sensing scene classification. IEEE Geoscience and Remote Sensing Letters, 12(11), 2321-2325.

Table 1. Details about the harvested forest sites under consideration. Drone and camera type: p4pro = DJI phantom 4 Pro with DJI camera (20 megapixel), p2GoPro= DJI Phantom 2 with GoProTM Hero 4 camera (12 megapixel).

Site	Approx. location	Drone and camera type	Spatial Resolution (cm)	Date of acquisition (DD.MM.YYYY)	Time of acquisition (hh:mm)	Area (ha)	Wheel rut length (m/ha)
A	10.69°E, 85.76°N	p4pro	3.2	13.07.2019	16:02	0.5	1,540
B	13.08°E, 82.53°N	p2GoPro	2.0	08.05.2016	13:47	1.2	1,698
C	10.92°E, 85.88°N	p4pro	3.3	11.07.2019	18:24	1.4	2,092
D	10.93°E, 85.88°N	p4pro	3.1	11.07.2019	18:00	1.6	1,720
E	12.13°E, 83.16°N	p4pro	2.0	11.07.2018	09:11	2.2	1,519
F	13.09°E, 82.52°N	p2GoPro	2.0	11.05.2016	19:36	2.4	1,507
G	11.68°E, 83.18°N	p4pro	1.5	08.09.2017	15:00	2.5	1,164
H	11.93°E, 83.26°N	p2GoPro	1.0	26.05.2016	20:20	2.7	2,693
I	11.97°E, 83.28°N	p4pro	2.9	27.04.2019	14:06	3.0	1,782
J	10.82°E, 86.12°N	p4pro	2.7	12.07.2019	19:30	3.4	614
K	11.88°E, 83.02°N	p4pro	1.0	18.05.2018	16:43	3.5	1,288
L	11.96°E, 83.21°N	p4pro	2.0	01.05.2017	18:57	3.9	1,698
M	11.19°E, 83.71°N	p4pro	1.0	09.07.2019	12:15	4.3	1,579
N	11.98°E, 83.19°N	p4pro	7.1	23.04.2017	13:27	4.5	1,887
O	11.93°E, 83.27°N	p4pro	2.0	30.08.2016	12:00	6.8	2,149
P	10.64°E, 86.36°N	p4pro	2.9	12.07.2019	16:37	6.8	2,120
Q	11.97°E, 83.19°N	p4pro	3.5	01.05.2017	19:07	7.9	1,583
R	12.10°E, 82.97°N	p2GoPro	1.9	29.05.2016	18:13	9.5	2,118
S	12.02°E, 82.90°N	p2GoPro	1.4	28.05.2016	18:04	11.0	1,436
T	10.48°E, 85.85°N	p4pro	2.4	19.05.2018	15:00	21.5	1,141

Table 2 Wheel rut detection accuracy metrics. Except for Training OA, all metrics are based on post-processed data.

Testing site	Training OA (%)	Testing OA (%)	F1 score (wheel rut)	Wheel rut detected (%)
A	98.5	70.4	0.71	76.4
B	96.8	78.2	0.70	79.5
C	98.7	80.4	0.75	86.0
D	97.3	78.0	0.72	70.0
E	98.1	58.6	0.64	57.5
F	96.5	73.7	0.67	72.4
G	96.4	73.2	0.56	56.5
H	95.4	70.0	0.71	82.2
I	93.2	66.2	0.52	56.7
J	95.4	75.1	0.47	49.0
K	97.6	76.7	0.65	64.4
L	98.2	72.7	0.52	69.4
M	98.5	67.4	0.53	60.2
N	92.3	64.0	0.45	20.0
O	98.5	80.7	0.83	77.2
P	98.0	79.3	0.73	75.6
Q	97.6	73.0	0.75	66.0
R	98.1	75.6	0.79	74.5
S	94.3	72.0	0.70	59.1
T	97.4	76.0	0.54	66.2

Table 3 Averaged confusion matrix for wheel rut detection of all 20 harvested sites after post-processing

		Reference			
		Wheel rut	**Non wheel rut**	**User’s Accuracy**	**F1 score**
Prediction	**Wheel rut**	27%	13%	68 %	0.67
	Non wheel rut	14%	46%	77 %	0.77
	Producer’s accuracy	66 %	78 %	Testing OA = 73.1%	

Figure1.

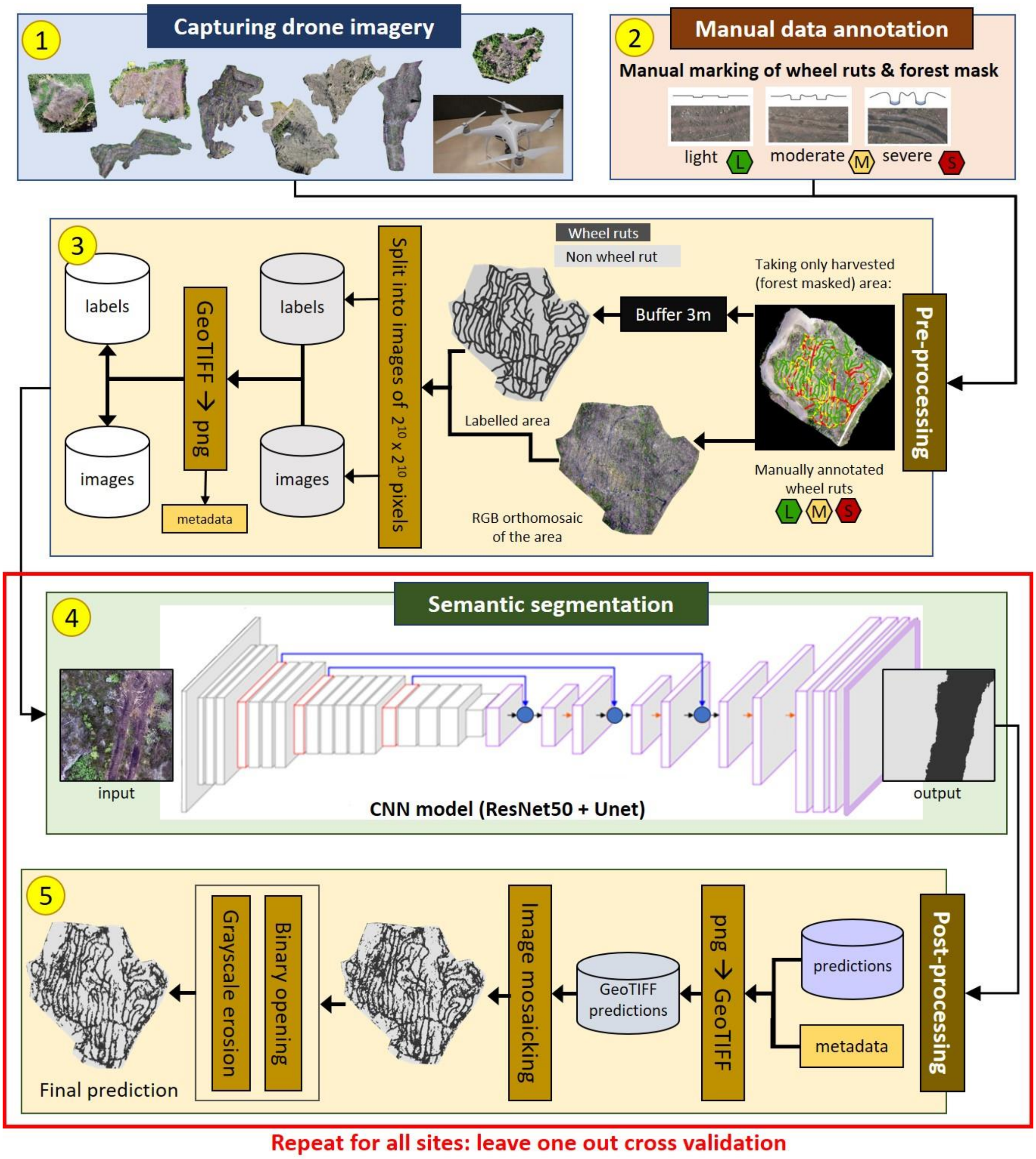

1
Capturing drone imagery
2
Manual data annotation
Manual marking of wheel ruts & forest mask
light L
moderate M
severe S
3
Wheel ruts
Non wheel rut
Taking only harvested (forest masked) area:
Pre-processing
Buffer 3m
labels
images
GeoTIFF → png
metadata
labels
images
Split into images of 2^10 x 2^10 pixels
Labelled area
RGB orthomosaic of the area
Manually annotated wheel ruts
L M S
4
Semantic segmentation
input
CNN model (ResNet50 + Unet)
output
5
Final prediction
Grayscale erosion
Binary opening
Image mosaicking
GeoTIFF predictions
png → GeoTIFF
predictions
metadata
Post-processing
Repeat for all sites: leave one out cross validation

Figure 2.

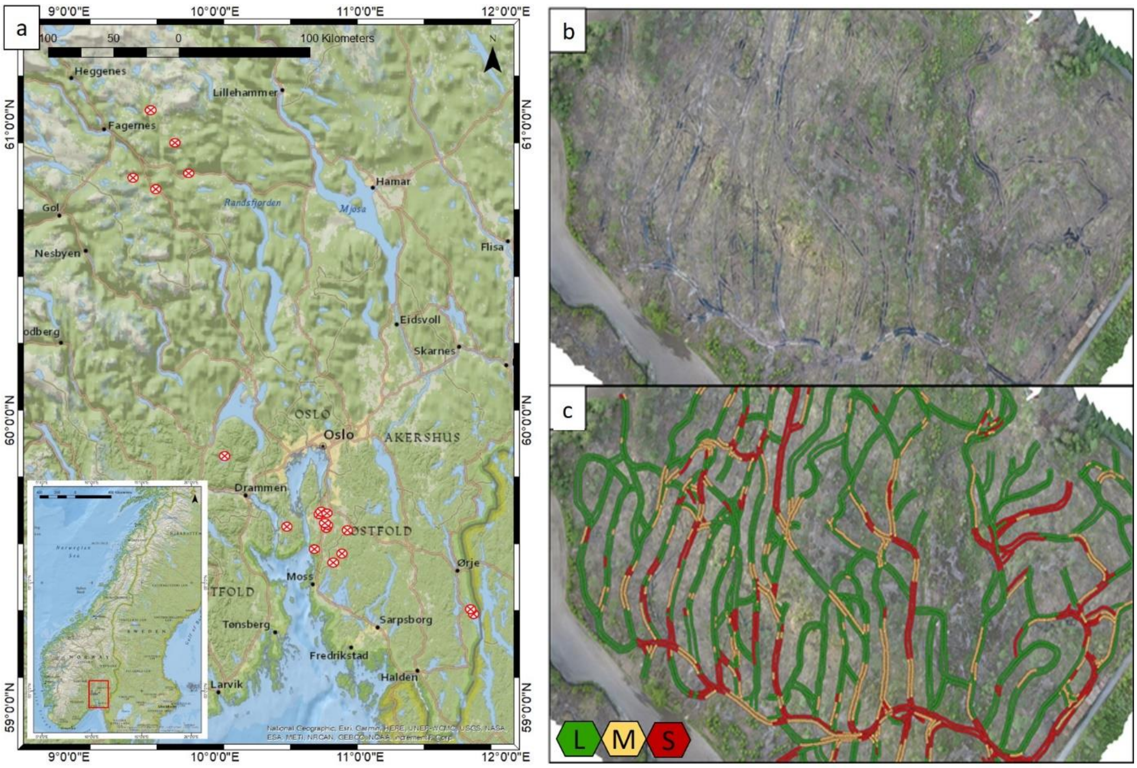


Figure 3.

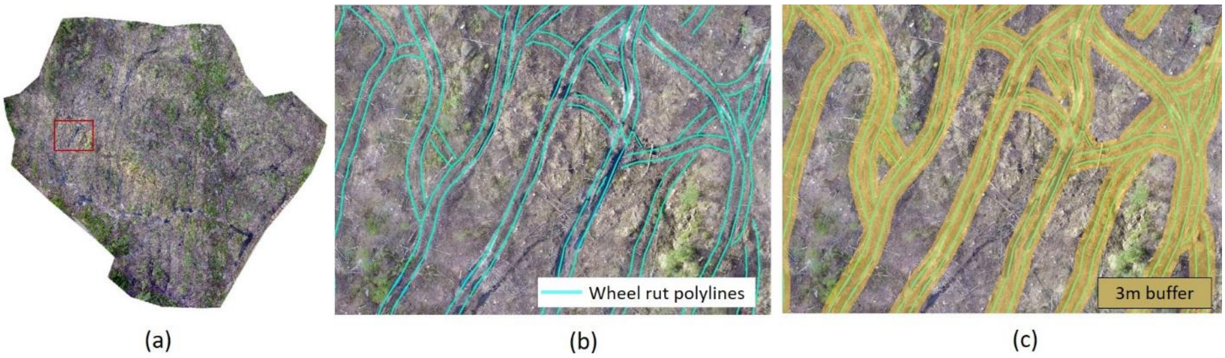


Figure 4.

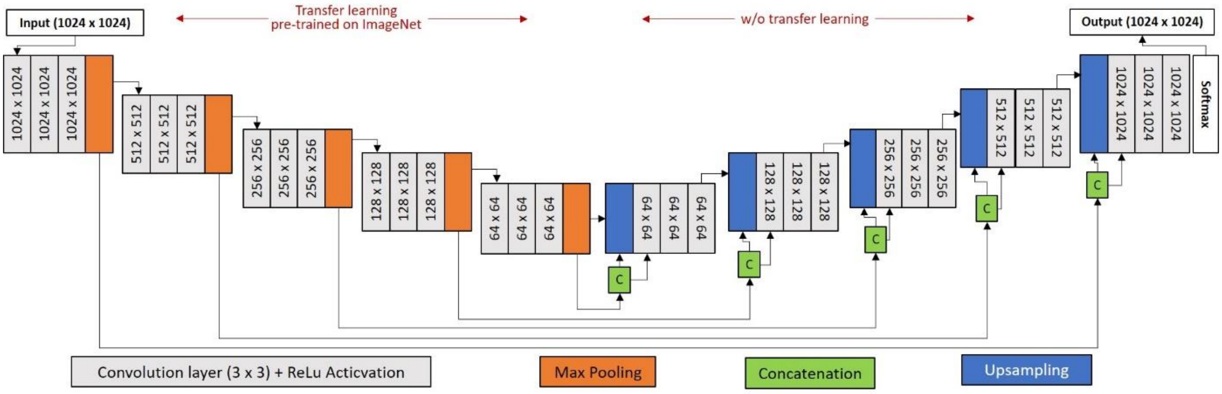


Figure 5.

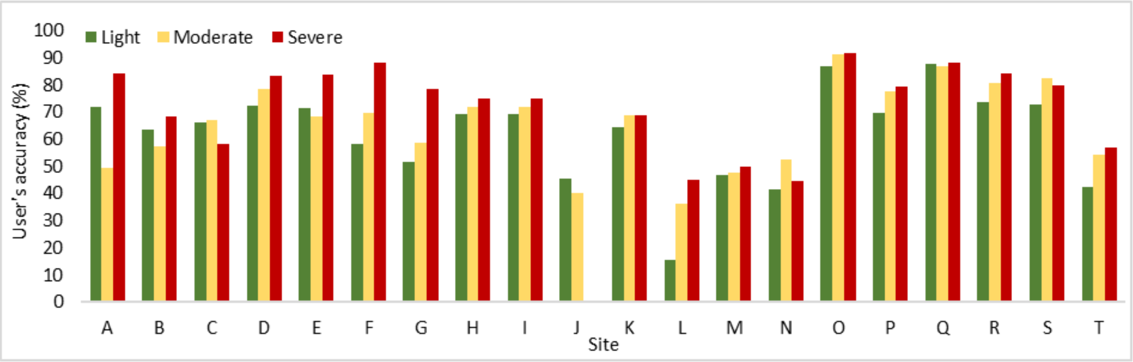
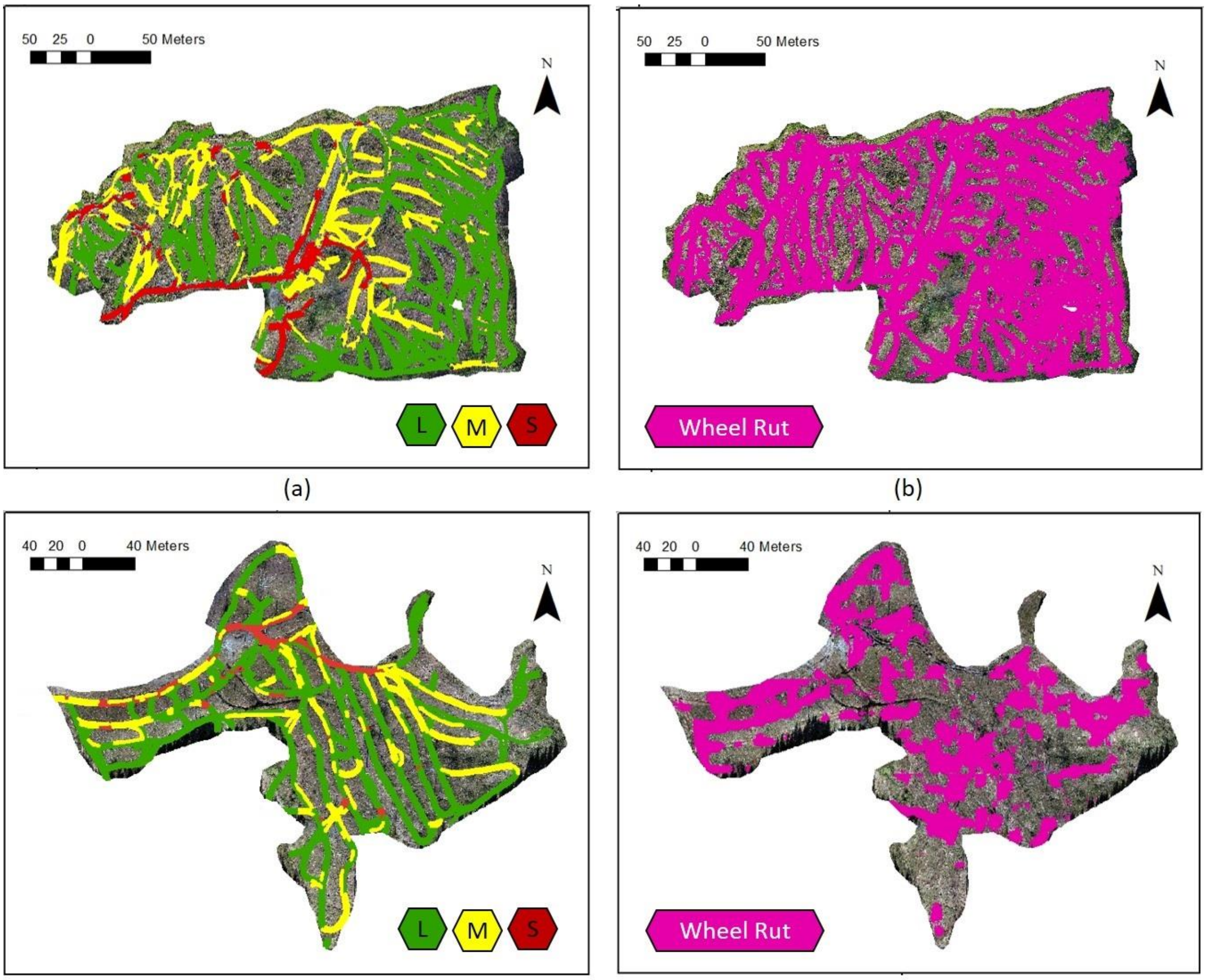


Figure 6.



(a)

(b)

(c)

(d)

Figure 7.

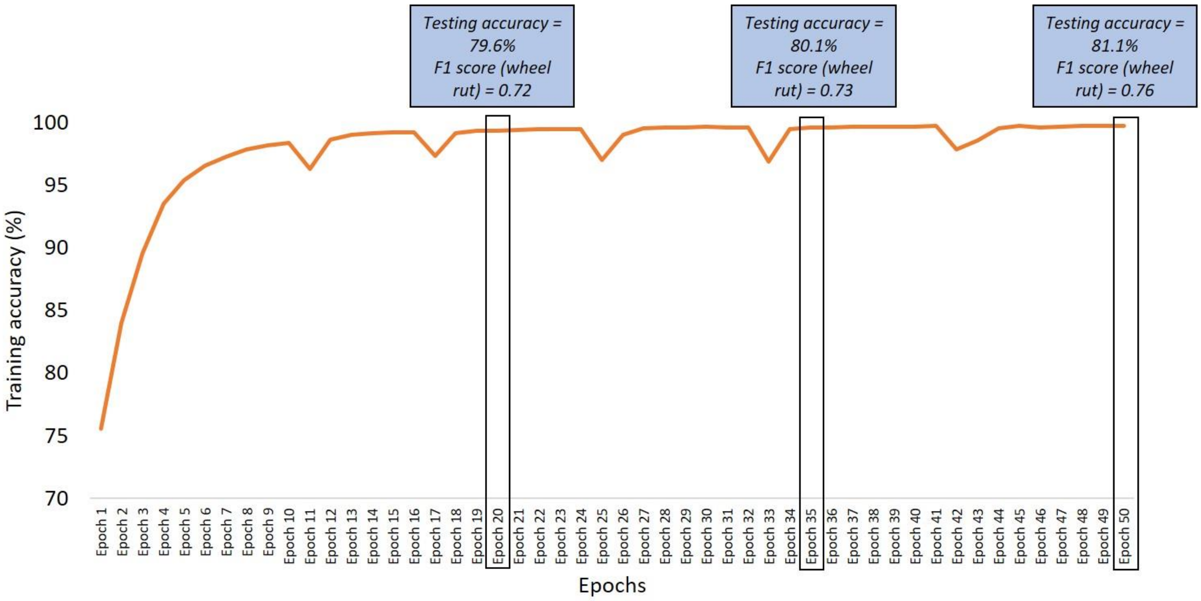


Figure1. Methodology flowchart for detecting wheel ruts using drone imagery

Figure 2. (a) Harvested sites across Norway, (b) drone-image ortho-mosaic of one study site, (c) manual wheel rut annotated shapefile overlaid the ortho-mosaic (Heppelmann et al. 2021).

Figure 3. Study site S (a); Overview (b), highlighted area with manually annotated wheel ruts (polylines); (c) 3 m buffer around wheel ruts

Figure 4. ResNet50 + UNet architecture for automatic detection of wheel ruts using drone imagery.

Figure 5. User's accuracy for light, moderate, and severe wheel ruts

Figure 6. Site O with (a) annotated severity labels and (b) predicted wheel ruts; Site N with (c) annotated severity labels and (d) predicted wheel ruts

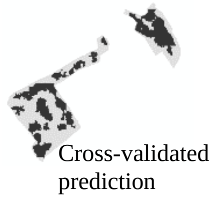
Figure 7. Training and testing accuracy for 50 epochs, trained using p4pro data, tested on site P

Appendices

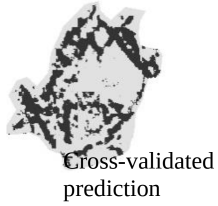
S.1 Confusion Matrices – all sites

The columns in the confusion matrices are the Reference (original) annotated labels, and the rows are the Predicted labels. The values in the confusion matrices are the number of pixels of the ortho-mosaic of that site. The thumbnails depict the actual reference image (training) and the prediction done without using the training image (cross-validated prediction).

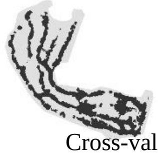
1) A

	Wheel rut	Non-wheel rut	UA (%); F1 score	
Wheel rut	1529845	790154	66.0 ; 0.71	
Non-wheel rut	474217	1478270	75.7 ; 0.70	
PA (%)	76.4	65.2	OA = 70.41%	

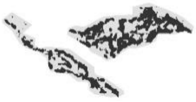
2) B

	Wheel rut	Non-wheel rut	UA (%); F1 score	
Wheel rut	8633868	5487601	61.2 ; 0.70	
Non-wheel rut	2226166	19012069	89.52 ; 0.83	
PA (%)	79.5	77.6	OA = 78.2 %	

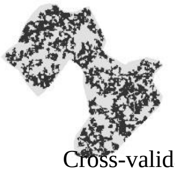
3) C

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	3750840	1833261	67.2 ; 0.75	
Non-wheel rut	610547	6257768	91.1 ; 0.84	 Cross-validated prediction
PA (%)	86	77.3	OA = 80.4 %	

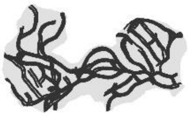
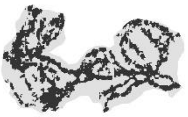
4) D

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	4606508	1523630	75.2 ; 0.72	
Non-wheel rut	1988010	7533521	79.2 ; 0.81	 Cross-validated prediction
PA (%)	70	83.2	OA = 78%	

5) E

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	19611835	7979382	71.1 ; 0.64	
Non-wheel rut	14502883	12235319	45.8 ; 0.52	 Cross-validated prediction
PA (%)	57.5	60.5	OA = 58.6 %	

6) F

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	15850645	9480648	62.6 ; 0.67	
Non-wheel rut	6041919	27698119	82.1 ; 0.97	 Cross-validated prediction
PA (%)	72.4	74.5	OA = 73.7 %	

7) G

	Wheel rut	Non-wheel rut	UA (%); F1 score	training
Wheel rut	19287261	14965715	56.3 ; 0.56	
Non-wheel rut	14894831	62108852	80.6 ; 0.81	Cross-validated prediction
PA (%)	56.5	80.6	OA = 73.2 %	

8) H

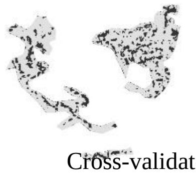
	Wheel rut	Non-wheel rut	UA (%); F1 score	training
Wheel rut	42357673	25668969	62.3 ; 0.71	
Non-wheel rut	9169087	35753791	79.6 ; 0.67	Cross-validated prediction
PA (%)	82.21	58.21	OA = 69.2 %	

9) I

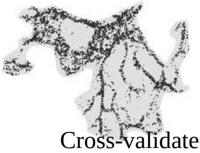
	Wheel rut	Non-wheel rut	UA (%); F1 score	training
Wheel rut	6303886	6996423	47.4 ; 0.52	

Non-wheel rut	4811984	16726515	77.67 ; 0.74	 Cross-validated prediction
PA (%)	56.71	70.51	OA = 66.1 %	

10) J

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	4797929	5744843	46 ; 0.47	
Non-wheel rut	5018338	27682333	84.6 ; 0.84	 Cross-validated prediction
PA (%)	49	83	OA = 75.1 %	

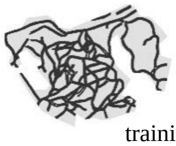
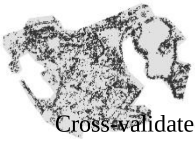
11) K

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	61325832	32726062	65.2 ; 0.65	
Non-wheel rut	33911502	158851440	82.4 ; 0.83	 Cross-validated prediction
PA (%)	64.4	83	OA = 76.7 %	

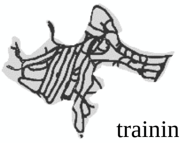
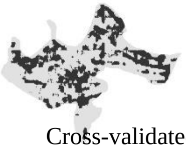
12) L

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	11594722	16091911	42 ; 0.52	
Non-wheel rut	5111268	44931961	90 ; 0.81	 Cross-validated prediction
PA (%)	69.4	73.6	OA = 72.7 %	

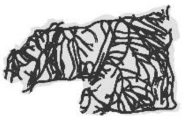
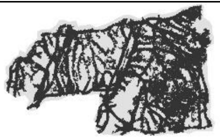
13) M

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	78352959	88907780	46.9 ; 0.53	
Non-wheel rut	51758511	212229730	80.4 ; 0.75	 Cross-validated prediction
PA (%)	60.2	70.5	OA = 67.4 %	

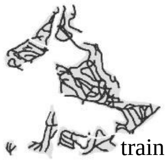
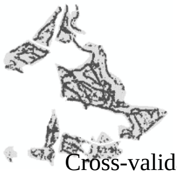
14) N

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	591411	710626	45.5 ; 0.3	
Non-wheel rut	2509780	5155920	67.3 ; 0.76	 Cross-validated prediction
PA (%)	20	87.8	OA = 64%	

15) O

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	79554888	9329993	89.5 ; 0.83	
Non-wheel rut	23528182	58359235	71.3 ; 0.78	 Cross-validated prediction
PA (%)	77.2	86.2	OA = 80.7 %	

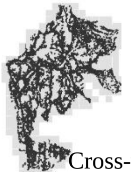
16) P

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	21790020	8805543	71.2 ; 0.73	
Non-wheel rut	7022656	38690763	84.6 ; 0.83	 Cross-validated prediction
PA (%)	75.6	81.4	OA = 79.3 %	

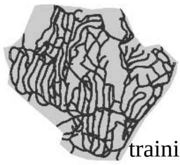
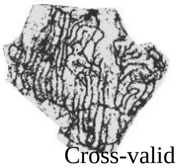
17) Q

	Wheel rut	Non-wheel rut	UA (%); F1 score	training
Wheel rut	25308319	3550036	87.7 ; 0.75	
Non-wheel rut	13104014	19813372	60.2 ; 0.70	Cross-validated prediction
PA (%)	66	84.8	OA = 73.04 %	

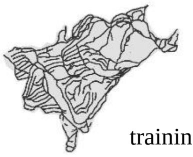
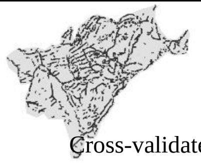
18) R

	Wheel rut	Non-wheel rut	UA (%); F1 score	training
Wheel rut	100139761	25939039	79.43 ; 0.79	
Non-wheel rut	34321776	86245417	71.53 ; 0.71	 Cross-validated prediction
PA (%)	74.47	76.9	OA = 75.6 %	

19) S

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	168829197	29124795	85.3 ; 0.70	
Non-wheel rut	116924142	205397804	63.7 ; 0.74	 Cross-validated prediction
PA (%)	59.1	87.6	OA = 72%	

20) T

	Wheel rut	Non-wheel rut	UA (%); F1 score	 training
Wheel rut	37090310	44173525	45.6 ; 0.54	
Non-wheel rut	18951185	162801876	89.6 ; 0.84	 Cross-validated prediction
PA (%)	66.2	78.6	OA = 76 %	

S.2 S, M, L, Non-wheel rut – Confusion matrix

The columns in the confusion matrices are the Reference (original) annotated labels, and the rows are the Predicted labels. The values in the confusion matrices are the number of pixels of the ortho-mosaic of that site.

1) A

	L	M	S	Non-wheel rut
Wheel rut	1207829	309829	142758	474217
Non-wheel rut	466467	316831	26481	1478270
UA (%)	72.1	49.5	84.4	75.7

2) B

	L	M	S	Non-wheel rut
Wheel rut	7800787	1333896	303153	2226166
Non-wheel rut	4506159	985817	136955	19012069
UA (%)	63.4	57.5	68.6	89.5

3) C

	L	M	S	Non-wheel rut
Wheel rut	2044998	1175297	107285	611072
Non-wheel rut	1051918	571360	77260	6276676
UA (%)	66.0	67.3	58.13	91.1

4) D

	L	M	S	Non-wheel rut
Wheel rut	2665291	1215696	222632	1988013
Non-wheel rut	1018070	331117	44220	7533528
UA (%)	72.36	78.6	83.4	79.12

5) E

	L	M	S	Non-wheel rut
Wheel rut	18220567	303153	201333	14502883
Non-wheel rut	7315663	136955	38386	12235319
UA (%)	71.3	68.6	83.9	45.8

6) F

	L	M	S	Non-wheel rut
Wheel rut	10496071	699339	3813203	3539030
Non-wheel rut	7535111	304986	499004	30174009
UA (%)	58.21	69.6	88.43	89.5

7) G

	L	M	S	Non-wheel rut
Wheel rut	9194854	9707118	2587032	14894831
Non-wheel rut	8577155	6825464	707321	62108852
UA (%)	51.73	58.72	78.52	80.65

8) H

	L	M	S	Non-wheel rut
Wheel rut	4950434	1260315	387141	4811984

Non-wheel rut	4705005	2245939	867003	16726515
UA (%)	69.5	71.8	75.1	77.66

9) I

	L	M	S	Non-wheel rut
Wheel rut	4950434	1260315	387141	4811984
Non-wheel rut	4705005	2245939	867003	16726515
UA (%)	69.5	71.8	75.1	77.66

10) J

	L	M	S	Non-wheel rut
Wheel rut	4712009	61955		5018338
Non-wheel rut	5618240	92006		27682337
UA (%)	45.61	40.24		84.65

11) K

	L	M	S	Non-wheel rut
Wheel rut	44293173	22961188	1744765	33911502
Non-wheel rut	24362024	10345849	787594	158851440
UA (%)	64.52	68.94	68.89	82.41

12) L

	L	M	S	Non-wheel rut
Wheel rut	167434	2730205	7037546	5111599
Non-wheel rut	915665	4838275	8627794	44944177
UA (%)	15.45	36.07	44.92	89.78

13) M

	L	M	S	Non-wheel rut
Wheel rut	55499849	21877304	10167519	51758511
Non-wheel rut	62977836	24064443	10217040	212229730
	46.84	47.62	49.87	80.4

14) N

	L	M	S	Non-wheel rut
Wheel rut	1127039	549638	183860	1479249
Non-wheel rut	1582496	498139	226291	3793158
UA (%)	41.60	52.45	44.82	71.94

15) O

	L	M	S	Non-wheel rut
Wheel rut	41665439	21667942	6847033	23822737
Non-wheel rut	6176954	2031039	606572	58366327
UA (%)	87.1	91.4	91.8	71.0

16) P

	L	M	S	Non-wheel rut
Wheel rut	17546420	3107604	77737	7071099
Non-wheel rut	7675275	903071	20050	38700101
UA (%)	69.6	77.5	79.5	84.5

17) Q

	L	M	S	Non-wheel rut
Wheel rut	16619224	5658734	799157	13653553
Non-wheel rut	2334265	846169	106264	19278209
UA (%)	87.6	87	88.2	59

18) R

	L	M	S	Non-wheel rut
Wheel rut	30822943	29045706	40261112	34321776
Non-wheel rut	11084846	12084546	24321670	86245417
UA (%)	73.55	80.76	84.09	71.53

19) S

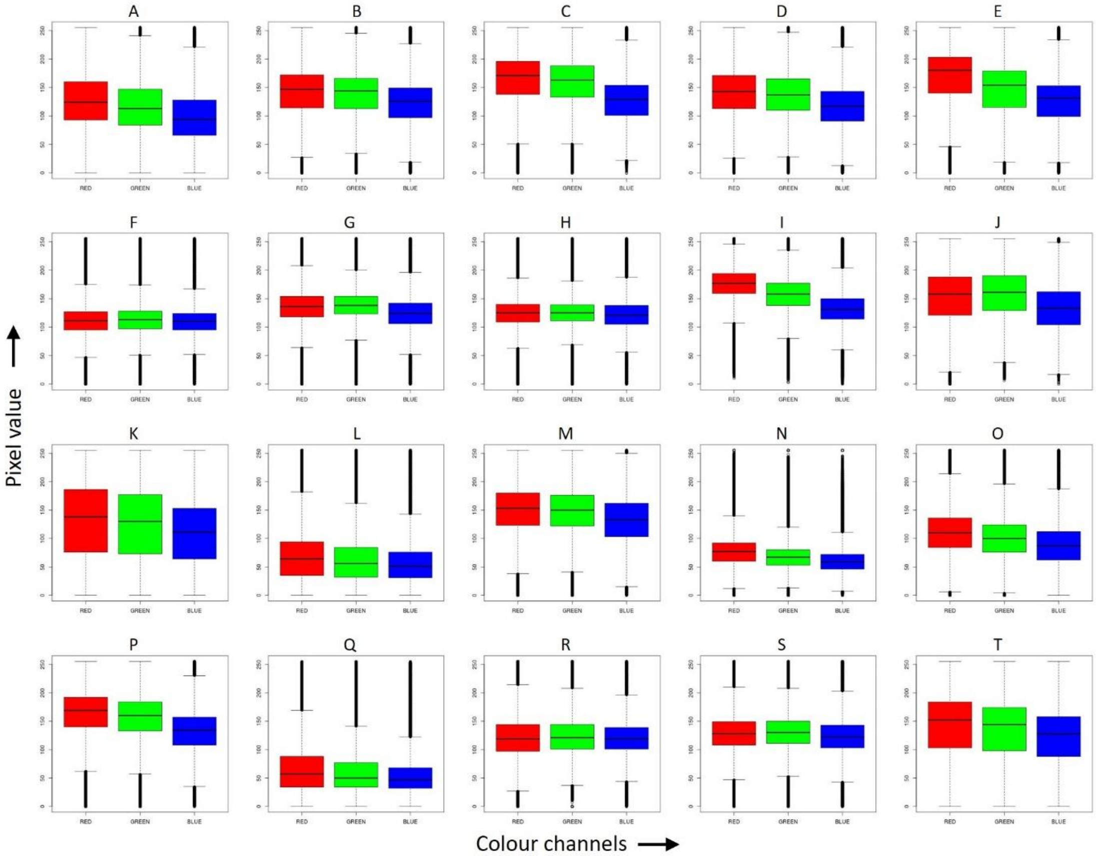
	L	M	S	Non-wheel rut
Wheel rut	91680578	42426956	40906071	45169390
Non-wheel rut	34414254	9025241	10452102	280503298
UA (%)	72.71	82.4	79.7	86.1

20) T

	L	M	S	Non-wheel rut
Wheel rut	25309113	8871241	5607622	18951185
Non-wheel rut	18951185	7452074	4218113	162801876
UA (%)	42.62	54.34	57.07	89.57

S.3 Pattern between sites and parameters

Figure 8. RGB profile for all sites (boxplots)


A
B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
RED
GREEN
BLUE
Pixel value
Colour channels