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

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

[1]*Division of Forest and Forest Resources, Norwegian Institute of Bioeconomy Research, Høgskoleveien 8, Ås 1431, Norway*
[2]*Department of Forest and Wood Science, Stellenbosch University, Matieland, 7602, Stellenbosch 7599, South Africa*

*Corresponding author Tel: +47 974 77 985; E-mail: johannes.breidenbach@nibio.no

Wheel ruts, i.e. soil deformations caused by harvesting machines, are considered a negative environmental impact of forest operations and should be avoided or ameliorated. However, the mapping of wheel ruts that would be required to monitor harvesting operations and to plan amelioration measures is a tedious and time-consuming task. Here, we examined whether a combination of drone imagery and algorithms from the field of artificial intelligence can automate the mapping of wheel ruts. 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 cross-validation of the model based on 20 harvested sites resulted in F1 scores of 0.69–0.84 with an average of 0.77, and in total, 79 per cent of wheel ruts were correctly detected. The highest accuracy was obtained for severe wheel ruts (average user's accuracy (UA) = 76 per cent), and the lowest accuracy was obtained for light wheel ruts (average UA = 67 per cent). Considering the nowadays ubiquitous availability of drones, the approach presented in our study has the potential to greatly increase the ability to effectively map and monitor the environmental impact of final felling operations with respect to wheel ruts. The automated mapping of wheel ruts may serve as an important input to soil impact analyses and thereby support measures to restore soil damages.

Introduction

Mechanized harvesting of forests is an integral part of sustainable forest management and is required to supply society with the required timber through efficient and safe forest operations. However, mechanized harvesting operations can cause considerable environmental impacts. Besides the inevitable but temporary loss of habitat for forest-dwelling animals and plant species, harvesting operations can also 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 (Heppelmann *et al.,* 2022). Despite considerable technological developments to reduce the impact on soils, wheel ruts can occur when forest operations are conducted under suitably low to moderate soil moisture or non-frozen conditions (Uusitalo *et al.,* 2015). The porosity of soils affected by wheel ruts may be reduced, resulting in anaerobic processes and reducing the stability and growth of future forests (Cambi *et al.,* 2015). Therefore, wheel ruts should be avoided as far as possible, and forest certification schemes and regulations include 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, Banu *et al.,* 2016, Iglhaut *et al.,* 2019, Kentsch *et al.,* 2020). In the field of forest operations, drone images and derived 3D products provide a valuable source of information to assess the environmental performance of the harvesting operation (Nevalainen *et al.,* 2017; Talbot and Astrup, 2021). In particular, drones have been used to measure wheel rut depth (Pierzchała *et al.,* 2014, 2016; Haas *et al.,* 2016; Talbot *et al.,* 2018; Marra *et al.,* 2021). These studies provided insights into the obtainable accuracy of rut depth measurements from drone imagery but required manual intervention to identify the trail network or localize specific measurement points or profiles for further analysis.

The studies mentioned so far have combined 3D elevation information obtained with photogrammetric approaches and colour images to detect the wheel rut type. However, the production of detailed 3D models is time-consuming and resource-intensive and complicates the prediction process

because it depends on a separate photogrammetric processing chain. Hence, such approaches have limited potential for real-time monitoring capabilities of drones in which wheel ruts would be automatically spotted while the drone is flying.

A more widespread operational deployment of drone based post-harvest assessment would require the partial or full automation of wheel rut detection and measurement (Talbot and Astrup, 2021). Established machine learning models such as support vector machines (Boser et al., 1992) and random forest (Breiman, 2001) typically operate on pixel level and hence, mostly depend on the pixels' spectral information. This makes it challenging to transfer a trained classifier successfully to a new scene since the spectral information may be influenced heavily by illumination conditions during the drone flight. Adding texture metrics and applying spectral normalization algorithms can partly help to address these problems, but it is unlikely that these steps will fully resolve the related issues. On the other hand, certain deep learning (DL) algorithms, such as convolutional neural networks (CNNs) mostly rely on textural information and are hence less dependent on an accurate and comparable spectral signal at the pixel level. DL models automatically learn the features from the data (lazy learning), enabling broad application automation and often outperforming traditional algorithms (Zou et al., 2015; Kattenborn et al., 2019; Wurm et al., 2019; Bhatnagar et al., 2020). The application fields for DL have recently increased drastically, partly due to the increasing availability of sophisticated DL libraries and complementary hardware to process the data. For the extraction of pattern-related information in remote sensing imagery, CNNs have become a popular choice (Ma et al., 2019). CNNs are utilized mainly for classification in the form of image classification, semantic segmentation, object detection and instance segmentation (Kattenborn et al., 2021). In the case of image classification, the output of the CNN is a 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 2D score map, similar to a probability map for each pixel that enables to produce a segmentation map of the same size. Object detection is mainly used to locate individual occurrences of an object within the image, and instance segmentation is a combination of object detection and semantic segmentation (Kattenborn et al., 2021). CNNs consist of several layers, including convolutional layers, which are the main building blocks of CNNs containing a set of filters for feature extraction (Mostafa and Wu, 2021); nonlinearity layers which increase the nonlinear representation ability of the convolutional layers (Yuan et al., 2020); pooling layers that down-sample by keeping the most important information (Albawi et al., 2017); and fully connected layers which define the relationship between the extracted features and the output (Yuan et al., 2020). In semantic segmentation, some fully connected layers are replaced by fully convolutional layers. More about the functionalities of CNNs can be found, for example, in Albawi et al. (2017), Bhatnagar et al. (2020) and Kattenborn et al. (2021). There are several new and off-the-shelf architectures available, which have already been successfully applied in remote sensing for image classification (Hoeser and Kuenzer, 2020; Kattenborn et al., 2021).

While we are not aware of studies on the automated detection of wheel ruts using DL or ML, CNNs have been applied for somewhat related objectives in an urban context (Audebert et al., 2018) to, for example, identify roads (Bayoudh et al., 2021), cracks in surfaces (Ali et al., 2021; Kim et al., 2021) and pavements (Ma et al., 2021), and railway tracks (Giben et al., 2015). In an agricultural context, Paul et al. (2021) used LiDAR data in combination with CNN to detect drainage ditches in Sweden. In addition, studies like Kanakaraddi et al. (2021) and Patil and Jadhav (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. All of these studies indicate that the automatic detection of linear but somewhat fuzzy features seems to be possible with CNN.

This study aims to examine whether the automated detection of wheel ruts caused by cut-to-length harvester and forwarder systems can be realized with a combination of drone RGB images of previously tree-covered sites acquired shortly after final harvests and CNNs. We use the CNN models to separate wheel ruts and unaffected areas in a binary classifier. We cross-validate our results using 20 independent harvested sites with areas between 0.5 and 21.5 ha in south-eastern Norway.

Material and methods

The processing workflow (Figure 1) consisted of five steps (1) Capturing the drone imagery; (2) Manual annotation of wheel ruts in the drone images as polyline vectors; (3) Pre-processing, which includes rasterization of the wheel rut vectors into the two classes wheel rut or unaffected area and splitting the drone imagery for feeding into the DL system; (4) Semantic segmentation to detect wheel and unaffected areas in all images per site; (5) Post-processing of the prediction maps, including mosaicking and applying morphological operations. Steps 4 and 5 were repeated for all sites in a k-fold ($k = 4$) cross-validation.

Study sites and drone data

A total of 20 study sites were surveyed with a drone after clear-cutting (Figure 2). The flights were conducted over a span of 4 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. (2022). All sites were productive forest areas in south-eastern Norway.

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 accomplished using a DJI Phantom 2 drone fitted with a GoPro™ Hero 4 12 mega-pixel camera (p2GoPro) (DJI 2013), which was later replaced with a DJI Phantom 4 Pro drone, 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 in most cases, although UgCS software (www.ugcs.com) was used on steeper sites to reduce variation in ground sampling distance (GSD) within the same
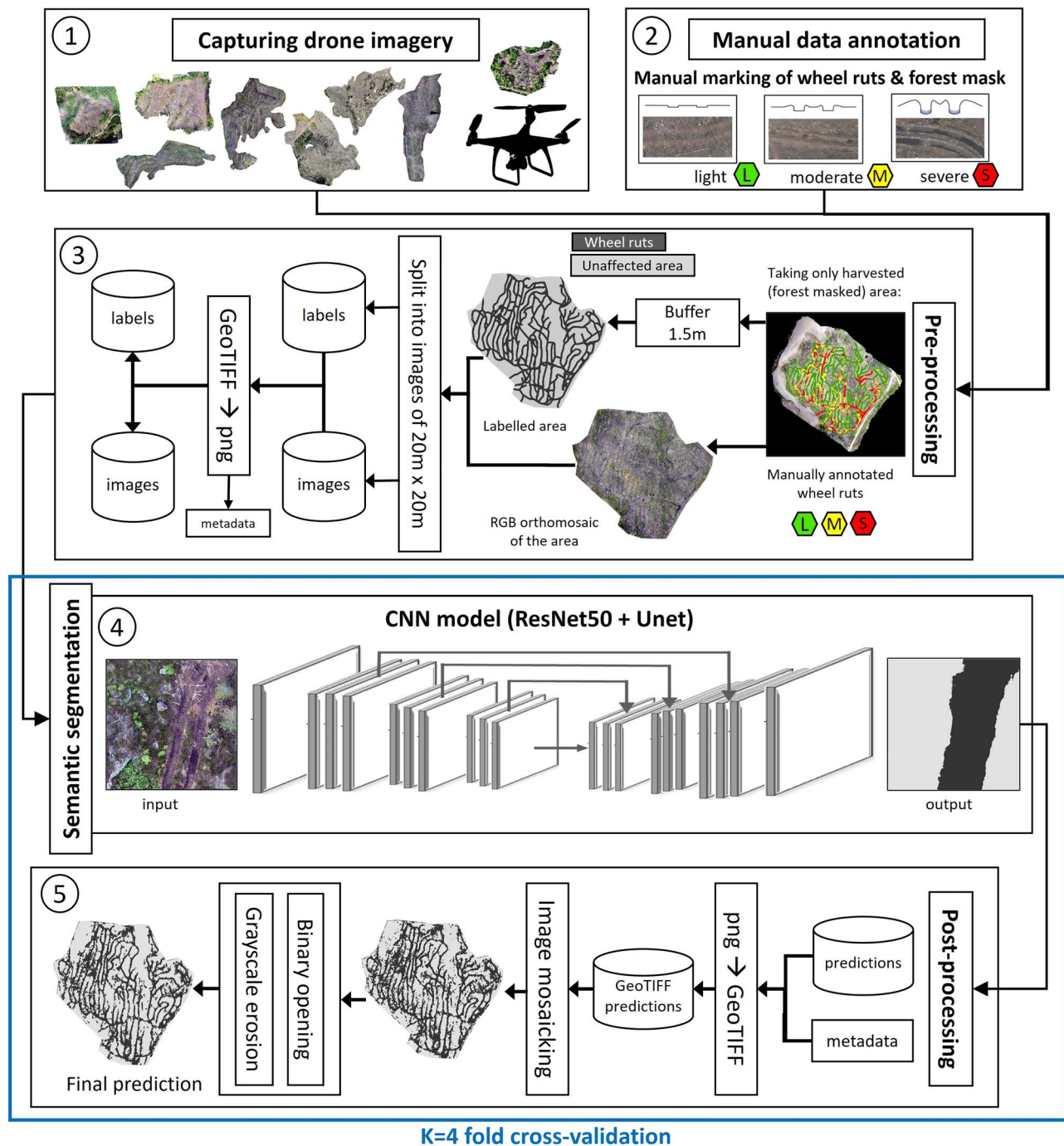
Figure 1 Methodology flowchart for detecting wheel ruts using drone imagery.

acquisition. For the lower resolution p2GoPro, a flight altitude of ~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. A forward overlap of 80 per cent and a lateral overlap of 70 per cent 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 (Agisoft, 2022) was used to generate an orthomosaic for each site that was used for the manual annotation of wheel ruts (next section) and DL.
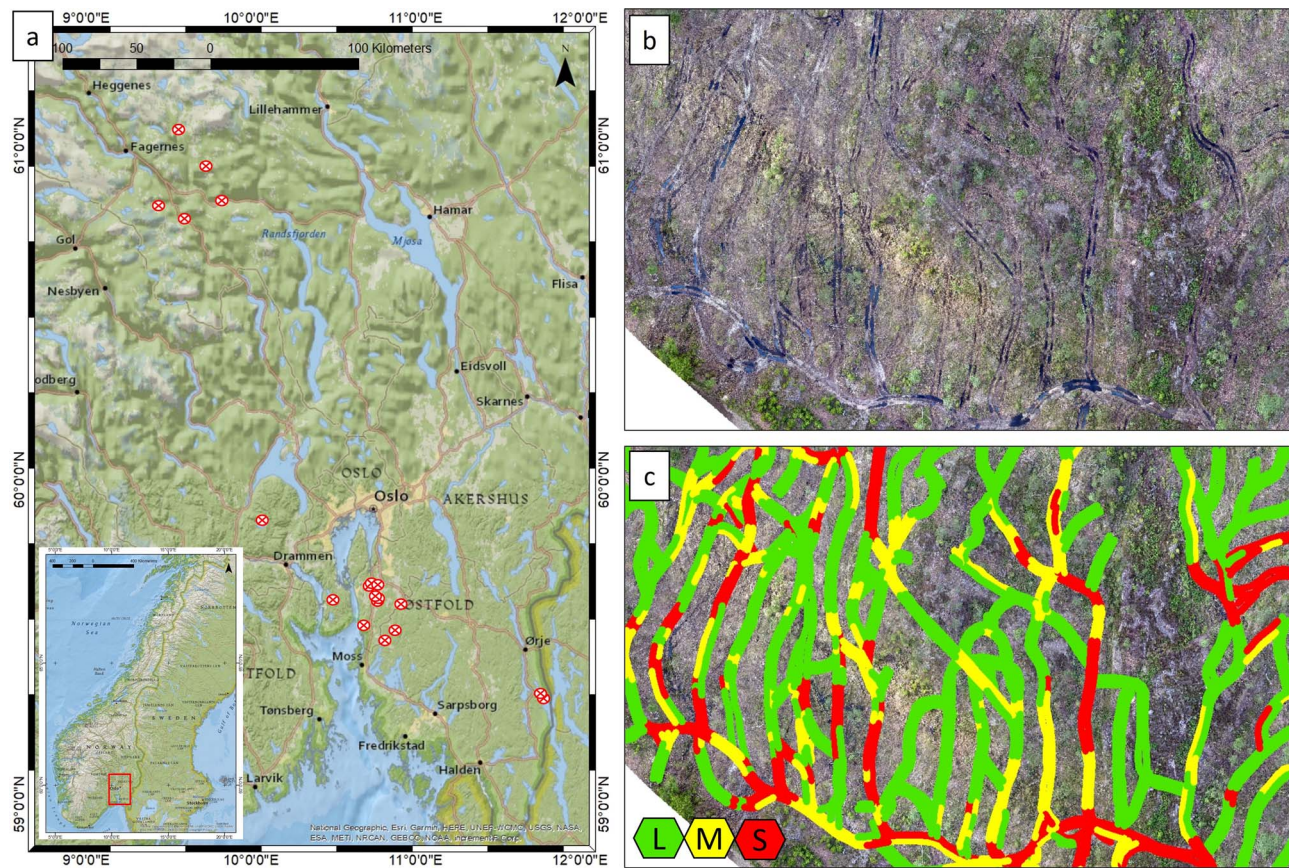
Figure 2 (a) Location of the harvested sites in Norway, (b) drone-image orthomosaic, (c) manually annotated wheel rut coloured by their severity (L = light, M = medium, S = severe) overlaid the orthomosaic.

Data annotation

The orthomosaics for all 20 sites were manually annotated in a GIS environment to register the location and severity of wheel ruts (Figure 1, step 2), as described in detail by Heppelmann *et al.* (2022). The annotation was carried out using visual interpretation of the drone imagery without additional field visits. Due to the removal of most trees during the harvesting operations (i.e. clear cut), the wheel ruts were clearly visible in the RGB orthomosaics. They were digitized as lines in the centre of the two parallel wheel ruts, which were further classified into the following three severity classes (Figures 2 and 6).

- Light: visible wheel ruts with no identifiable soil displacement or rut-formation; 63.6 per cent of the total annotated wheel ruts were light.
- Moderate: showed rutting with minor soil displacement and deeper indentations but no visible loss of water drainage functions; 24.6 per cent of the total annotated wheel ruts were moderate.
- Severe: all wheel ruts with either substantial soil displacement, deep indentations, loss of water drainage functions, or a combination of various of these factors; 11.8 per cent of the total annotated wheel ruts were severe.

Preparation of the annotated data for deep learning

The detection of wheel ruts is prerequisite for determining their severity in a more in-depth analysis, concentrated to a smaller area. Therefore, we segmented wheel ruts and unaffected areas as a binary class independent of the wheel-rut severity category (Figure 1, 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. A visual assessment of the drone images revealed that, on average, ∼1.5 m on both sides of the centerline of the wheel ruts were affected by the harvesting operation (Figure 3). The input reference image was therefore generated by applying a 1.5-m buffer from the centre of the wheel rut on both sides of the annotated polylines, dissolving the results and converting the resulting polygon into a binary raster with a value of one in correspondence to wheel ruts and zero to the unaffected area (Figure 3b–d). Additionally, any non-forest area was masked out.

There are two types of image splitting that can happen based on either image size (pixels) or image extend (coverage). The inputs to the deep learning model are RGB images in portable network graphics (png) format with an extent of 20 × 20 m. Therefore, the RGB orthomosaic GeoTIFF raster and the annotated data were split into tiles of 20 × 20 m and converted

Table 1 Details about the sites under consideration.

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	1540
B	13.08°E, 82.53°N	p2GoPro	2.0	08.05.2016	13:47	1.2	1698
C	10.92°E, 85.88°N	p4pro	3.3	11.07.2019	18:24	1.4	2092
D	10.93°E, 85.88°N	p4pro	3.1	11.07.2019	18:00	1.6	1720
E	12.13°E, 83.16°N	p4pro	2.0	11.07.2018	09:11	2.2	1519
F	13.09°E, 82.52°N	p2GoPro	2.0	11.05.2016	19:36	2.4	1507
G	11.68°E, 83.18°N	p4pro	1.5	08.09.2017	15:00	2.5	1164
H	11.93°E, 83.26°N	p2GoPro	1.0	26.05.2016	20:20	2.7	2693
I	11.97°E, 83.28°N	p4pro	2.9	27.04.2019	14:06	3.0	1782
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	1288
L	11.96°E, 83.21°N	p4pro	2.0	01.05.2017	18:57	3.9	1698
M	11.19°E, 83.71°N	p4pro	1.0	09.07.2019	12:15	4.3	1579
N	11.98°E, 83.19°N	p4pro	7.1	23.04.2017	13:27	4.5	1887
O	11.93°E, 83.27°N	p4pro	2.0	30.08.2016	12:00	6.8	2149
P	10.64°E, 86.36°N	p4pro	2.9	12.07.2019	16:37	6.8	2120
Q	11.97°E, 83.19°N	p4pro	3.5	01.05.2017	19:07	7.9	1583
R	12.10°E, 82.97°N	p2GoPro	1.9	29.05.2016	18:13	9.5	2118
S	12.02°E, 82.90°N	p2GoPro	1.4	28.05.2016	18:04	11.0	1436
T	10.48°E, 85.85°N	p4pro	2.4	19.05.2018	15:00	21.5	1141

Drone and camera type: p4pro = DJI Phantom 4 Pro with DJI camera (20 mega-pixel), p2GoPro = DJI Phantom 2 with GoPro™ Hero 4 camera (12 mega-pixel).

to png format. The metadata containing the geotags of the tiles was stored and used later for mosaicking the prediction maps. Different sites under consideration had different spatial resolutions (Table 1), meaning that even though the image tiles were resampled to a fixed pixel size of 1024 × 1024 pixels when training the CNN model (see Semantic segmentation using CNN section), the model was exposed to a variation in GSD.

The orthomosaics of the 20 sites were split into a total of 2597 images with a size of 20 x 20 m. The sites were randomly divided into four groups for training and testing, i.e. the training was done using 15 sites, and the model was tested on the rest of the five sites in a $k = 4$ cross-validation. Group specifications are as follows:

1. Group 1
   a. Sites: D, G, K, Q, T
   b. Training images = 1734; Testing images = 863

2. Group 2
   a. Sites: B, C, H, N, S
   b. Training images = 2096; Testing images = 501

3. Group 3
   a. Sites: A, E, F, J, R
   b. Training images = 2074; Testing images = 523

4. Group 4
   a. Sites: I, L, M, O, P
   b. Training images = 1887; Testing images = 710

Semantic segmentation using CNN

For semantic segmentation (Figure 1, 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). See Hamedianfar *et al.* (2022) for an overview of various deep learning methods used for forestry applications. Preliminary analysis on a subset of this study's data revealed that using a popular machine learning algorithm (Random Forest), only poor results for detecting wheel ruts could be obtained (Appendix A1). A brief description along with a comparison of several DL methods is provided in Appendix A1, Table 4. Based on some preliminary analyses in which we examined several combinations of network architectures (see Appendix A1), we found that the combination of ResNet50 and UNet provided the best accuracy, and we thus selected this combination for further detailed analysis.

Figure 4 shows the DL architecture used in this study. The ResNet50 architecture is resilient to overfitting due to its residual learning concept (Yang *et al.,* 2020), which states that each layer will feed to the next layer and the activation layer directly (He *et al.* 2016). The layers are considered residual blocks to facilitate the network's training (Ardakani *et al.,* 2020). For decoding the information from ResNet50, the UNet architecture is used (Ronneberger *et al.,* 2015). The UNet retains the spatial information while upsampling to circulate context from a lower to a higher resolution layer (Alam *et al.,* 2021). Due to limited data availability, a transfer of pre-trained weights from ImageNet (Russakovsky *et al.,* 2015) was applied on ResNet50, while
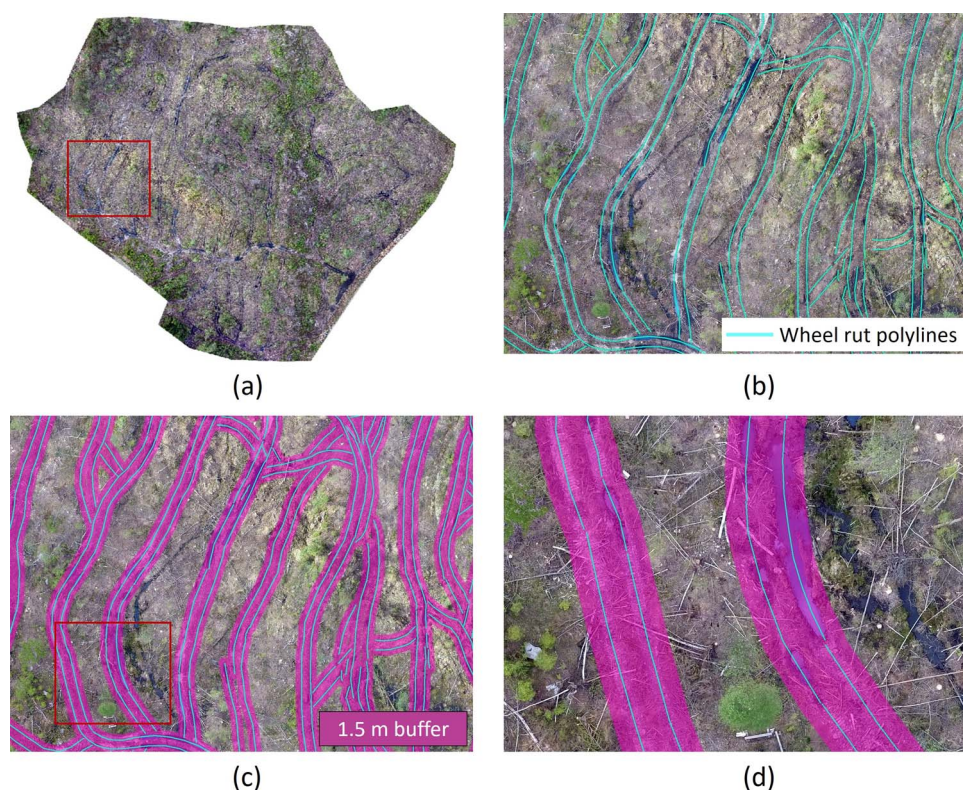
Figure 3 (a) Orthomosaic of study site S; (b) zoomed-in view with manually annotated wheel ruts as polylines; (c) 1.5 m buffer around wheel ruts; (d) zoomed-in view of wheel rut buffers.

the UNet (decoder) was trained from scratch. Transfer learning enables the transfer of information of a pre-trained network (trained on a big dataset, for example, ImageNet) to a new network which is then fine-tuned for the required application. In a somewhat simplified language, one could say one uses a pre-trained network that has already learned 'how to see' and then teaches this network the specific problem it has to solve in the new images. This step enhances the model's performance and makes the convergence faster (Nigam *et al.,* 2018; Kentsch *et al.,* 2020).

The first part of the ResNet50 workflow is an image preprocessing step, resampling all the input images to 1024 × 1024 pixels, which was the required input matrix size of the model. This process changes the pixel size of the original images. An initial test with images consisting of 1024 × 1024 pixels, which results in input images of different extent in metres but maintains the original resolution (i.e. pixel size), was also tried but led to poor model performance (Appendix A2). Furthermore, image augmentation (flipping, rotation) was used to increase the number of input images.

ResNet50 is a deep network having 50 layers, including batch normalization layers (He *et al.* 2016). 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 a matrix multiplication over the images using a filter of size 3 × 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 and Arıca, 2018). ReLu removes all the negative parts from the input (image *f*), as described in equation (1).

$$
\begin{aligned}
A(f) &= 0; f < 0 \\
A(f) &= f; f \geq 0
\end{aligned}
\tag{1}
$$

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. A pooling layer was used to extract the most important features (e.g. sharp and smooth features). Here, we use the 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 hyperparameters also plays a vital role in the performance of the CNN model. 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) regularization method was used. An initial learning rate (the rate at which the model is updated) of 0.01 was selected by the hit and trial method. The batch size was set to 20. For upsampling, the UNet architecture (Figure 4) was trained from scratch. UNet uses both transpose convolution and
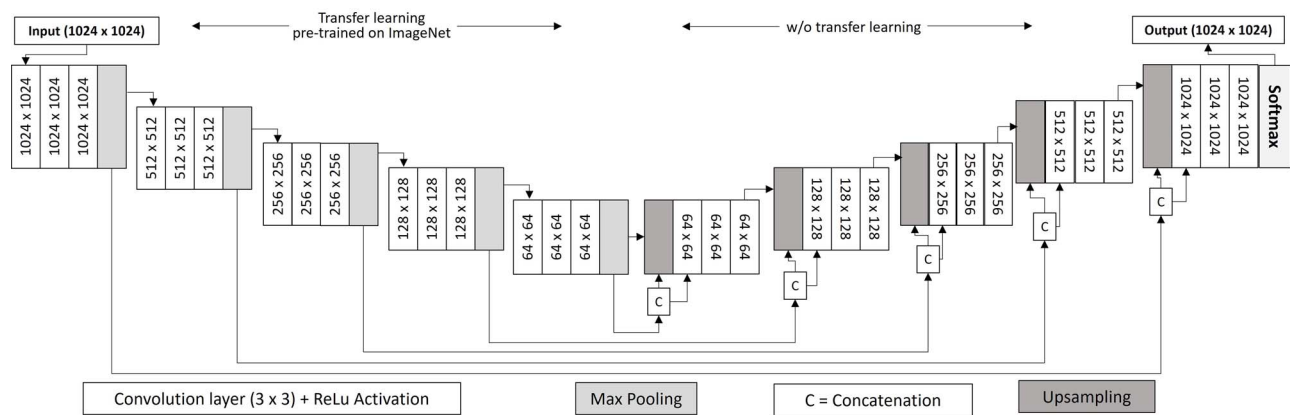
Figure 4 ResNet50 + UNet architecture for automatic detection of wheel ruts using drone imagery.

skip connections to sync the feature maps from the encoder and decoder.

The model was run for 20 epochs for each of the four cross-validation groups as an initial analysis showed that an increase in the number of epochs did not markedly improve the precision. The programming code defining the model is available at Bhatnagar *et al.* (2021) for full transparency. Our implementation is based on the repository by Divamgupta (2019).

Post-processing

The predictions (binary masks indicating presences and absences of wheel ruts in each pixel) were further enhanced to locate the wheel ruts using multiple morphological operations. The noise elements were removed in the post-processing without distorting the original results. First, a binary opening was performed (equation (2)). Opening in mathematical morphology is defined as an erosion followed by dilation using the same structuring element (SE) or kernel on the image (f). The aim was to enhance the wheel rut detection. As a result, spurious regions smaller than ~5 m^2 were removed.

$$F_1 = (f \ominus SE_1) \oplus SE_1 \tag{2}$$

where F_1 was the opened-binary image, $\ominus$ is erosion and $\oplus$ is dilation (Serra, 1979). SE_1 was a circular disc with a radius of 2 m.

Second, grayscale erosion was performed to define the wheel ruts accurately and remove overestimation along the boundaries of the segments. This step replaces each pixel with the local minimum of the defined SE (SE_2) around the pixel

$$F_2 = F_1 \ominus SE_2 \tag{3}$$

where F_2 is the eroded image, and SE_2 was a circular disc with a radius of 20 cm that facilitates the erosion operation. For the example of a drone image with a pixel size of 1 cm, SE_1 and SE_2 were matrices of size 100×100 and 10×10, respectively. An example of the post-processing steps is shown in the Results section (Figure 5).
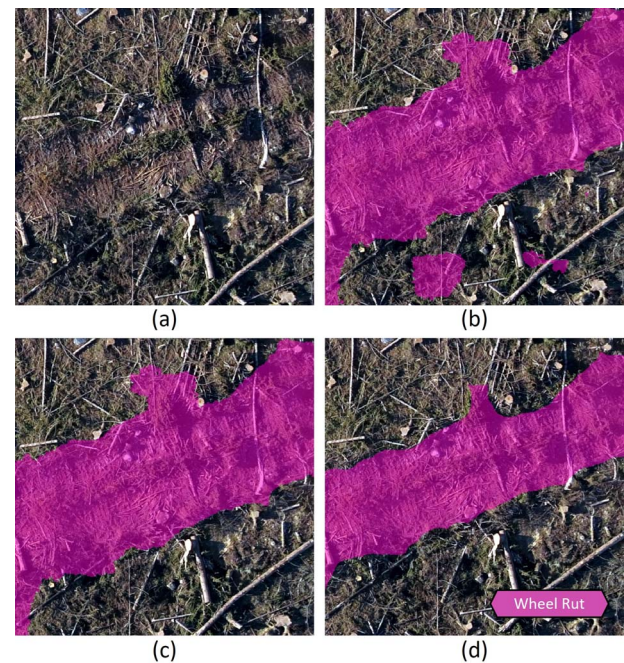

Figure 5 Result of the post-processing steps: (a) original wheel rut image, (b) original prediction map of wheel rut not post-processed, (c) prediction map after step 1 – area opening, closing to remove islands of pixels, (d) prediction map after step 2 – the erosion of wheel ruts, final post-processed result.

Validation

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

1. Splitting the data into training (number of sites minus 5 = 15) and testing data (remaining five sites).
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 (based on pixels) of the manually annotated input images vs the model prediction maps.

Table 2 Confusion matrix for wheel rut detection overall 20 harvested sites after post-processing.

		Reference				
		Wheel rut	Unaffected area	Total	User's Accuracy	F1 score
Prediction	Wheel rut	38%	10%	48%	78%	0.77
	Unaffected area	11%	41%	52%	81%	0.80
	Total	49%	51%	100%		
	Producer's accuracy	79%	80%		OA = 79.5%	

All values except for the F1 score are given in percent of the total number of pixels. A confusion matrix with the absolute number of pixels is given in Appendix A3.

Additionally, testing OA (accuracy of CNN model applied on the testing data, equation (4)), F1 score (equation (5)), user's accuracy (UA) and producer's accuracy (PA) where calculated (equations (6) and (7)).

$$OA = (TP + TN) / (TP + FP + FN + TN) \qquad (4)$$

$$F1 = 2TP/(2TP + FP + FN) \qquad (5)$$

$$PA = TP/(TP + FN) \qquad (6)$$

$$UA = TP/(TP + FP) \qquad (7)$$

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

We developed a five-step procedure to detect wheel ruts in drone images using a CNN model over 20 harvested sites in Norway. The model was evaluated in a fourfold cross-validation where the model was fit using 15 sites, while five sites were available as independent test data. The CNN model resulted in an average OA over all sites of 77.4 per cent. Post-processing with morphological operators resulted in the final wheel rut map with a slightly improved OA of 79.5 per cent (see also Appendix A3). The improvement by post-processing is visualized for an example area in Figure 5. The average F1 score, PA, and UA of the final wheel rut map were 0.77, 78.9 and 73.5, respectively. The overall confusion matrix (containing data from all pixels in all of the 20 sites) is given in Table 2. Confusion matrices for all sites individually are shown in Appendix A3, and accuracy metrics for all sites are collectively presented in Table 3, including accuracies by severity class.

The drone data were collected using two different sensors (Table 1). However, despite a difference in the training data size (75 per cent p4pro, 15 per cent p2GoPro) and sensor properties, there was no notable difference in the accuracy for detecting wheel ruts for the sites from either sensor. The average OA for p4pro and p2GoPro were 79.6 per cent and 78.9 per cent, respectively. The accuracy metrics were also relatively robust, given other acquisition characteristics such as flying height and time of flight (Tables 1 and 3).

Examples of wheel rut predictions for each severity class are shown in Figure 6. Severe wheel ruts were best detected with an average UA of 76 per cent (range = 65–96 per cent). Moderate and light wheel ruts were more often confused with the unaffected area, which resulted in average UAs of 69 per cent (range = 52–92 per cent) and 67 per cent (range = 55–91 per cent), respectively. This can also be seen in Figure 6d, where a patch of severe wheel rut was identified amidst unidentified light wheel ruts. The PA is not provided for severity classes because predictions were not made for severity classes, i.e. the omission error represented by FN is not available in the reference. Site-specific accuracies and confusion matrices can be found in Appendix A4.

A visual inspection of the wheel rut prediction maps showed that interferences in the form of residual logs, branches, harvest residues and shadows could increase the FP leading to over-estimation of wheel ruts. Figure 7 shows two cases, site O being one of the sites with the best accuracies (F1 score of 0.84) and site J with one of the poorest accuracies (F1 score of 0.69). The reason for the relatively poor model performance of site J was that 95 per cent of the total wheel ruts were in the light severity class that is less well detected, and there were no severe wheel ruts present on this site. Also, it can be seen that a blurred patch (upper left) is present for site J, which hampers the detection of light wheel ruts in this part of the image.

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. One key-attribute related to soil conditions after harvesting is wheel ruts since they serve as an indicator of the intensity of soil disturbance in the area. In this study, we suggest a drone-based workflow to automatically detect wheel ruts using CNN. For this, a ResNet50 + UNet architecture was applied, and the study was verified for 20 sites in Norway.

The testing was done using $k = 4$-fold cross-validation technique, where all the sites were randomly divided into groups of five sites. An initial analysis using leave-one-out-cross-validation was performed, and the results did not differ markedly from k-fold cross-validation despite a considerably smaller number of

Table 3 Wheel rut detection accuracy metrics after post-processing based on cross-validated data including wheel rut user's accuracy given severity classes.

Site	OA (%)	F1 score – wheel rut	PA – wheel rut (%)	UA – wheel rut (%)	UA – Light (%)	UA – Moderate (%)	UA – Severe (%)
A	71.2	0.72	85.4	66.0	55.0	55.0	65.0
B	82.7	0.75	86.0	67.0	68.1	58.2	69.3
C	81.3	0.81	85.6	70.0	58.2	70.0	71.0
D	82.1	0.82	77.0	85.2	85.6	80.5	90.9
E	69.3	0.70	73.4	62.1	61.7	63.1	83.5
F	73.3	0.74	67.7	76.6	78.4	52.0	73.0
G	80.5	0.70	74.0	66.0	60.2	73.3	75.7
H	78.3	0.81	92.0	72.3	69.2	78.0	87.3
I	71.2	0.70	60.0	84.0	83.7	86.0	87.6
J	84.9	0.69	68.2	69.0	68.9	56.3	–
K	82.4	0.76	81.3	64.4	66.2	71.8	73.2
L	86.0	0.81	84.8	75.2	74.1	74.4	67.2
M	73.4	0.74	62.5	78.6	62.0	63.0	68.0
N	85.1	0.79	91.4	65.6	54.5	70.2	71.8
O	84.0	0.84	95.0	73.1	69.7	70.1	77.3
P	82.1	0.83	73.2	91.0	90.5	91.5	96.3
Q	75.0	0.79	72.1	74.2	60.0	72.0	74.8
R	76.2	0.80	76.0	79.5	63.0	63.0	69.0
S	84.5	0.81	88.2	68.0	62.0	81.2	77.1
T	86.0	0.83	83.1	82.3	55.6	67.4	67.9

images available for model fitting in the case of *k*-fold cross-validation. Therefore, *k*-fold cross-validation was used given the computation and time benefits. On average, ~1900 images were used to train the model, and the testing was done on ~650 images. The results (OA, UA, PA, F1 score) prove that the suggested method enables the transfer of the trained algorithm to new sites and works well on datasets which were not used during the training stage. Studies such as Talbot *et al.* (2018) and Marra *et al.* (2021) have used drone imagery for manually detecting wheel ruts, but to our knowledge, no study so far has classified post-harvest wheel ruts automatically using only RGB images. Nevalainen *et al.* (2017) have reported an accuracy of 65 per cent in identifying wheel ruts with a depth >20 cm using an image matching-based digital terrain model, which was sufficient for post-harvest quality assurance. Other studies aimed at detecting linear structures in remotely sensed data are comparable only to a limited degree. For example, Paul *et al.* (2021) have used LiDAR data in combination with a machine and deep learning classification algorithms to identify ditches in peatland forests with promising results. In the case of the harvested forest as a semi-natural landscape, the interference in the images in terms of wood logs, branches and other harvest residues increases. This makes detecting wheel ruts challenging and different from detecting features from relatively clearly structured human-made surfaces such as pavements and roads. The latter have often been classified with very high accuracies. For example, Bayoudh *et al.* (2021) used a pre-trained network and obtained an overall accuracy of 99 per cent for detecting roads using RGB imagery from the benchmarked datasets. Zhang *et al.* (2018) used ResNet+UNet for road extraction with 90 per

cent accuracy using aerial images. Similarly, Kim *et al.* (2021) and Ali *et al.* (2021) have used instance classification to detect cracks in surfaces like pavements with an accuracy of over 95 per cent. Giben *et al.* (2015) detected railway tracks with 93 per cent accuracy by utilizing the information from the surroundings such as ballast, wood, concrete, lubricator, rail and fastener. In their study, such context information of objects frequently occurring along railway tracks helped identify railway tracks with high accuracy. Such regular or uniform surrounding information is difficult to find in harvested forests which complicates the classification problem and may hence also explain the lower accuracies obtained in our study. A visual examination of the prediction maps showed that the presence of logs, branches, harvest residues and shadows hampered the correct identification of wheel ruts since these natural objects intersected and partly covered the wheel ruts and hence increased the variability of how a wheel rut can look like in the image.

In our study, the applied post-processing steps improved the CNN results. Due to the area-opening technique, spurious regions were removed, decreasing FN and FP for wheel ruts. The next step, erosion of the wheel ruts, made the wheel ruts more defined, increasing the TN for the unaffected area and thus OA.

In this study, the light wheel ruts, representing 63.6 per cent of annotated wheel ruts, were mostly shallow and visually similar to the unaffected area and also with less clear textural features, resulting in poorer detection accuracy. At the same time, light wheel ruts pose little harm to the soil, and therefore, a lower detection probability in this class is less concerning (Heppelmann *et al.,* 2022). The severe wheel ruts generally have reduced water drainage functions and considerable soil displacement, making
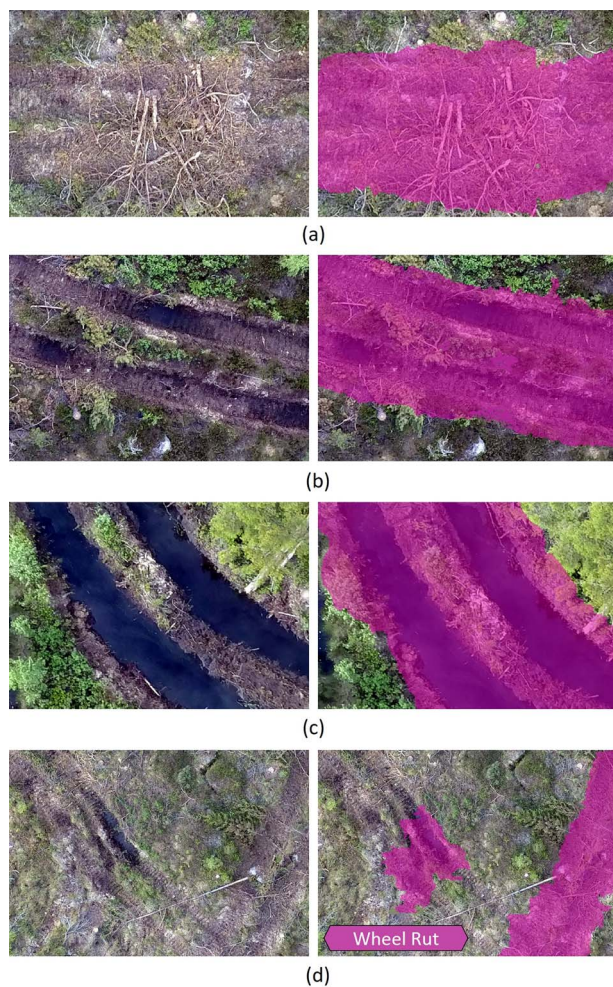
Figure 6 Examples of wheel ruts in site S for different severity classes (a) light, (b) moderate, (c) severe, (d) combination of light and severe. Left column: drone images, right column: drone images overlaid with detected wheel rut after post-processing (pink). Row (d) is an example of an erroneous classification where parts of light wheel ruts were not detected.

them unique and identifiable. We found that despite the severe class representing only 11.8 per cent of the total length of the annotated wheel ruts, it was the one detected with the highest 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 models work best for the most important severity class.

Recommendations and future work

A preliminary analysis (summarized in Appendix A2, Table 5) showed that, on average, the F1 score was improved by 5 per cent, and testing OA by 10 per cent when images with constant image extent in metres (20 m) were used, i.e. the approach described in the Methods Section, as compared to constant pixel size.

The fact that the variations in spatial resolution of the drone imagery did not have a notable effect on the accuracy of the predictions was encouraging as it opens up possibilities to segment wheel-ruts from higher altitudes and thus more efficiently. The procedure to clip the image tiles based on their spatial coverage (20 × 20 m) followed by resampling to the same pixel resolution made the contextual information uniform across the sites irrespective of the original spatial resolution of the imagery. Therefore, we recommend keeping a constant image size in terms of image extent (in metres) while training the DL model. This enhances the ability of the model to learn the wheel rut pattern across imagery from multiple data sources. Also, two types of sensors were used to capture the drone images; namely, p4pro and p2GoPro, with differences in the technical specifications (e.g. shutter mechanism, sensor size, focal length), and it was seen that the proposed DL model was also robust and applicable across the different sensors regardless of their different RGB colour profile (A5) and sensor specifications. Future development of the proposed method should aim at retraining the existing model with additional annotated images covering a broader spectrum of image acquisition parameters (i.e. higher altitudes) and drone sensors (i.e. including newer cameras), potentially leading to models applicable also to traditional aerial photography or even very-high-resolution satellite imagery. Further work should also aim at generalizing the proposed model to a broader range of seasonal, illumination and atmospherical conditions, thus ensuring its applicability with any new drone acquisition.

The results achieved so far are promising and imply that, due to the usage of transfer learning, the proposed approach worked well with ~75 per cent of the whole training dataset, thus reducing the need for extensive data annotation campaigns. This is particularly helpful when working in a new area with non-annotated images, which is often the case in practical applications. Drone data acquisition could thus become an integral part of the harvest documentation in order to comply with certification schemes or statutory provisions. Corresponding maps may help to guide post-harvest restoration of the soil. Since the proposed methodology can be used to detect any kind of linear wheel rut-like features, it may be possible to extend it to find disturbances, for example, in conservation sites. Norwegian nature legislation, for example, restricts motorized traffic outside marked trails which is, however, an increasing problem in the open landscape of the Norwegian mountains with low bush vegetation. An application of our method in this context could map the affected area as a first step in the management of this type of nature use.

Conclusion

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 detected due to their prominence in appearance compared to light wheel ruts that are more similar to the surroundings.
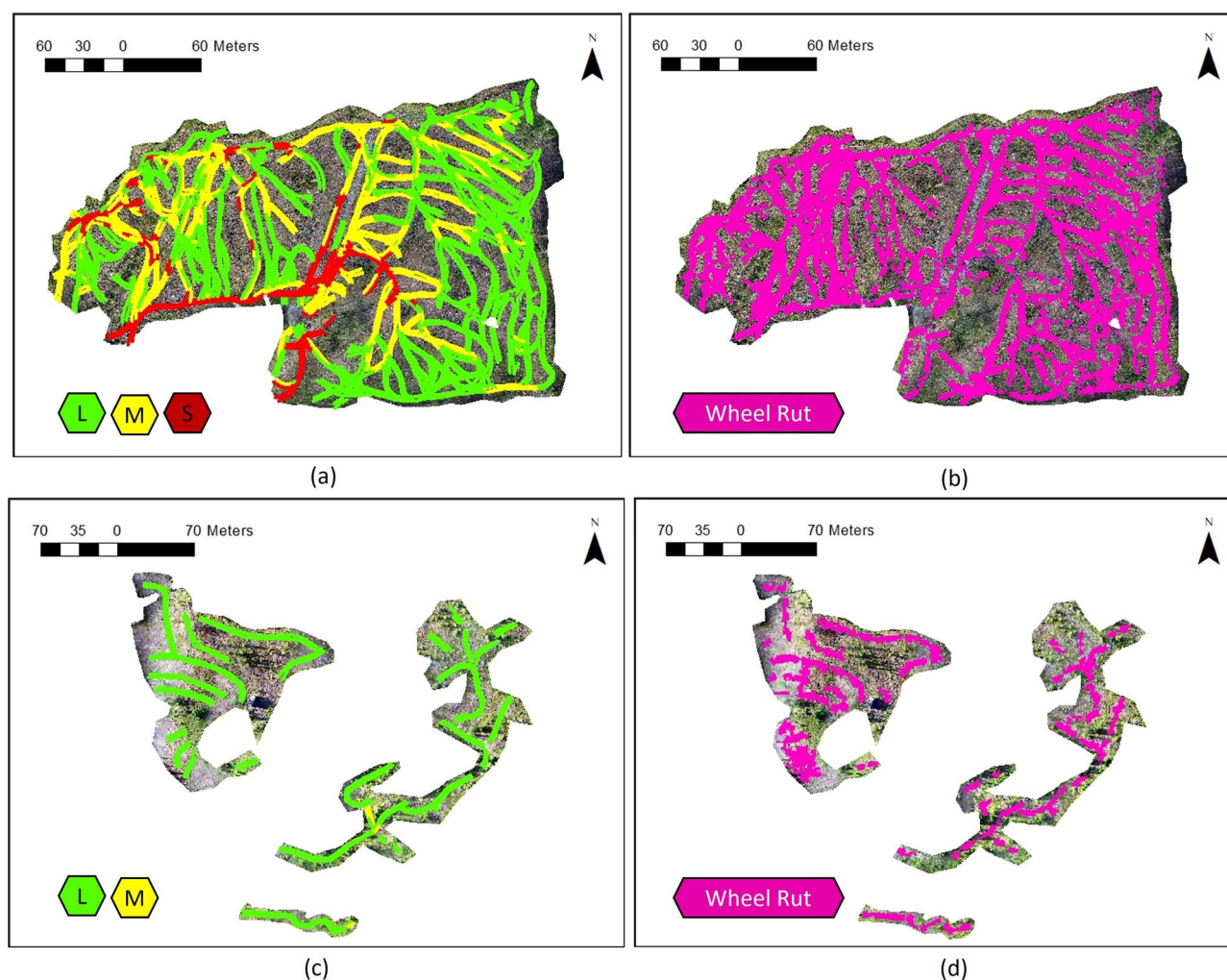
Figure 7 Examples for the site with the highest model accuracy (F1 score = 0.84, a, b) and the lowest model accuracy (F1 score = 0.69 , c, d). Site O with (a) annotated severity labels (L = light, M = medium, S = severe) and (b) predicted wheel ruts; Site J with (c) annotated severity labels and (d) predicted wheel ruts.

3. The proposed model is robust and can be used to detect 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 an improved overall environmental performance of harvest operations with varying acquisition characteristics.

Supplementary data

Supplementary data are available at *Forestry* online.

Data availability

Data and programming code can be accessed upon publication of the article via https://github.com/SmartForest-no/wheelRuts_semanticSegmentation. The final model trained with all 20 sites and programming code is openly available (Bhatnagar *et al.,* 2021).

Conflict of interest statement

None declared.

References

Agisoft 2022 Agisoft Metashape User Manual, Professional Edition, Version 1.8, https://www.agisoft.com/pdf/metashape-pro_1_8_en.pdf (accessed on 19 June 2022).

Alam, M., Wang, J.F., Guangpei, C., Yunrong, L.V. and Chen, Y. 2021 Convolutional neural network for the semantic segmentation of remote sensing images. *Mob. Netw. Appl.* **26**, 200–215.

Albawi, S., Mohammed, T.A. and Al-Zawi, S. 2017 Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*. IEEE, Antalya, Turkey, pp. 1-6.

Ali, L., Alnajjar, F., Jassmi, H.A., Gochoo, M., Khan, W. and Serhani, M.A. 2021 Performance evaluation of deep CNN-based crack detection and localization techniques for concrete structures. *Sensors* **21**, 1688.

Ampoorter, E., Van Nevel, L., De Vos, B., Hermy, M. and Verheyen, K. 2010 Assessing the effects of initial soil characteristics, machine mass and traffic intensity on forest soil compaction. *For. Ecol. Manag.* **260**, 1664–1676.

Ardakani, A.A., Kanafi, A.R., Acharya, U.R., Khadem, N. and 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. *Comput. Biol. Med.* **121**, 103795.

Audebert, N., Le Saux, B. and Lefèvre, S. 2018 Beyond RGB: very high resolution urban remote sensing with multimodal deep networks. *ISPRS J. Photogramm. Remote Sens.* **140**, 20–32.

Banu, T.P., Borlea, G.F. and Banu, C. 2016 The use of drones in forestry. *J. Environ. Sci. Eng. B* **5**, 557–562.

Bayoudh, K., Hamdaoui, F. and Mtibaa, A. 2021 Transfer learning based hybrid 2D-3D CNN for traffic sign recognition and semantic road detection applied in advanced driver assistance systems. *Appl. Intell.* **51**, 124–142.

Bhatnagar, S., Gill, L. and Ghosh, B. 2020 Drone image segmentation using machine and deep learning for mapping raised bog vegetation communities. *Remote Sens.* **12**, 2602.

Bhatnagar, S., Puliti, S., Talbot, B., Heppelmann, J., Breidenbach, J. and Astrup, R. 2021 Mapping wheel-ruts from timber harvesting operations using deep learning techniques in drone imagery – codes. *Zenodo.* . https://doi.org/10.5281/zenodo.5746878.

Bircanoğlu, C. and Arıca, N. 2018 A comparison of activation functions in artificial neural networks. In *2018 26th Signal Processing and Communications Applications Conference (SIU)*. IEEE, Izmir, Turkey, pp. 1-4.

Boser, B.E., Guyon, I.M. and Vapnik, V.N. 1992 A training algorithm for optimal margin classifiers. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, Pittsburgh Pennsylvania, USA. July 27–29, pp. 144-152.

Breiman, L. 2001 Random forests. *Mach. Learn.* **45**, 5–32.

Cambi, M., Certini, G., Neri, F. and Marchi, E. 2015 The impact of heavy traffic on forest soils: a review. *For. Ecol. Manag.* **338**, 124–138.

Canziani, A., Paszke, A. and Culurciello, E. 2016 An analysis of deep neural network models for practical applications. arXiv preprint. https://doi.org/10.48550/arxiv.1605.07678.

Divamgupta. 2019 *Image-Segmentation-Keras*. Available online: https://github.com/divamgupta/image-segmentation-keras.git (accessed on 18 November 2021).

DJI. 2013 https://dl.djicdn.com/downloads/phantom_2/en/Phantom_2_Release_Notes_en.pdf (accessed on 19 June 2022).

DJI. 2020 DJI, https://dl.djicdn.com/downloads/phantom_4_pro/20200803/Phantom_4_Pro_V2.0_Release_Notes_EN.pdf (accessed on 19 June 2022).

Giben, X., Patel, V.M. and 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)*. IEEE, Quebec City, Canada, pp. 621–625.

Haas, J., Ellhöft, K.H., Schack-Kirchner, H. and Lang, F. 2016 Using photogrammetry to assess rutting caused by a forwarder – a comparison of different tires and bogie tracks. *Soil Tillage Res.* **163**, 14–20.

Hamedianfar, A., Mohamedou, C., Kangas, A. and Vauhkonen, J. 2022 Deep learning for forest inventory and planning: a critical review on the remote sensing approaches so far and prospects for further applications. *Int. J. For. Res.* cpac002. https://doi.org/10.1093/forestry/cpac002.

He, K., Zhang, X., Ren, S. and Sun, J. 2016 Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. Las Vegas, NV, USA, pp. 770–778. doi: 10.1109/CVPR.2016.90.

Heppelmann, J.B., Talbot, B., Fernández, C. and Astrup, R. 2022 Depth-to-water maps as predictors of rut severity in fully mechanized harvesting operations. *Int. J. For. Eng.* **33**, 108–118. https://doi.org/10.1080/14942119.2022.2044724.

Hoeser, T. and Kuenzer, C. 2020 Object detection and image segmentation with deep learning on earth observation data: a review-part i: evolution and recent trends. *Remote Sens.* **12**, 1667.

Iglhaut, J., Cabo, C., Puliti, S., Piermattei, L., O'Connor, J. and Rosette, J. 2019 Structure from motion photogrammetry in forestry: a review. *Curr. For. Rep.* **5**, 155–168.

Kanakaraddi, S.G., Chikaraddi, A.K., Pooja, B.L. and Preeti, T. 2021 Detection of roads in satellite images using deep learning technique. In *ICT Analysis and Applications*. Springer, Singapore, pp. 441-451.

Kattenborn, T., Eichel, J. and Fassnacht, F.E. 2019 Convolutional neural networks enable efficient, accurate and fine-grained segmentation of plant species and communities from high-resolution UAV imagery. *Sci. Rep.* **9**, 1–9.

Kattenborn, T., Leitloff, J., Schiefer, F. and Hinz, S. 2021 Review on convolutional neural networks (CNN) in vegetation remote sensing. *ISPRS J. Photogramm. Remote Sens.* **173**, 24–49.

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

Kim, B., Yuvaraj, N., Preethaa, K.S. and Pandian, R.A. 2021 Surface crack detection using deep learning with shallow CNN architecture for enhanced computation. *Neural Comput. Applic.* **33**, 9289–9305.

Lu, H., Fu, X., Liu, C., Li, L.G., He, Y.X. and Li, N.W. 2017 Cultivated land information extraction in UAV imagery based on deep convolutional neural network and transfer learning. *J. Mt. Sci.* **14**, 731–741.

Ma, D., Fang, H., Wang, N., Xue, B., Dong, J. and Wang, F. 2021 A real-time crack detection algorithm for pavement based on CNN with multiple feature layers. *Road Mater. Pavement Des.* https://doi.org/10.1080/14680629.2021.1925578.

Ma, L., Liu, Y., Zhang, X., Ye, Y., Yin, G. and Johnson, B.A. 2019 Deep learning in remote sensing applications: a meta-analysis and review. *ISPRS J. Photogramm. Remote Sens.* **152**, 166–177.

Marra, E., Wictorsson, R., Bohlin, J., Marchi, E. and Nordfjell, T. 2021 Remote measuring of the depth of wheel ruts in forest terrain using a drone. *Int. J. For. Eng.* **32:3**, 224–234.

Mostafa, S. and Wu, F.X. 2021 Chapter 3 - Diagnosis of autism spectrum disorder with convolutional autoencoder and structural MRI images. In Ayman S. El-Baz, Jasjit S. Suri, (eds) *Neural Engineering Techniques for Autism Spectrum Disorder*. Academic Press, pp. 23–38.

Nevalainen, P., Salmivaara, A., Ala-Ilomäki, J., Launiainen, S., Hiedanpää, J., Finér, L. *et al.* 2017 Estimating the rut depth by UAV photogrammetry. *Remote Sens.* **9**, 1279.

Nigam, I., Huang, C. and Ramanan, D. 2018 Ensemble knowledge transfer for semantic segmentation. In *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, Lake Tahoe, NV, USA, pp. 1499-1508.

Patil, D. and Jadhav, S. 2021 Road extraction techniques from remote sensing images: a review. *Innov. Data Commun. Technol. Appl.* **59**, 663–677.

Paul, S.S., Ågren, A.M. and Lidberg, W. 2021 Detection of drainage ditches using high-resolution LIDAR data in the Swedish forest. In *EGU General Assembly Conference Abstracts*. Vienna, Austria, pp. EGU21-EGU2226.

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

Pierzchała, M., Talbot, B. and Astrup, R. 2016 Measuring wheel ruts with close-range photogrammetry. *Forestry Int. J. For. Res.* **89**, 383–391.

Puliti, S.; ørka, H.O.; Gobakken, T.; Næsset, E. 2015 Inventory of Small Forest Areas Using an Unmanned Aerial System. *Remote Sens.* **7**, 9632–9654. https://doi.org/10.3390/rs70809632.

Ronneberger, O., Fischer, P. and Brox, T. 2015 U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*. Springer, Cham, pp. 234-241.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S. *et al.* 2015 Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.* **115**, 211–252.

Serra, J. 1979 Biomedical image analysis by mathematical morphology (author's transl). *Pathol. Biol.* **27**, 205–207.

Talbot, B. and Astrup, R. 2021 A review of sensors, sensor-platforms and methods used in 3D modelling of soil displacement after timber harvesting. *Croat J. For. Eng.* **42**, 149–164.

Talbot, B., Rahlf, J. and Astrup, R. 2018 An operational UAV-based approach for stand-level assessment of soil disturbance after forest harvesting. *Scand. J. For. Res.* **33**, 387–396.

Uusitalo, J., Salomäki, M. and Ala-Ilomäki, J. 2015 The effect of wider logging trails on rut formations in the harvesting of peatland forests. *Croat J. For. Eng.* **36**, 125–130.

Wurm, M., Stark, T., Zhu, X.X., Weigand, M. and Taubenböck, H. 2019 Semantic segmentation of slums in satellite images using transfer learning on fully convolutional neural networks. *ISPRS J. Photogramm. Remote Sens.* **150**, 59–69.

Yang, P., Dong, C., Zhao, X. and Chen, X. 2020 The surface damage identifications of wind turbine blades based on ResNet50 algorithm. In *2020 39th Chinese Control Conference (CCC)*. IEEE, Shenyang, China, pp. 6340-6344.

Yuan, X., Qi, S., Wang, Y. and Xia, H. 2020 A dynamic CNN for nonlinear dynamic feature learning in soft sensor modeling of industrial process data. *Control. Eng. Pract.* **104**, 104614.

Zhang, Z., Liu, Q. and Wang, Y. 2018 Road extraction by deep residual u-net. *IEEE Geosci. Remote Sens. Lett.* **15**, 749–753.

Zou, Q., Ni, L., Zhang, T. and Wang, Q. 2015 Deep learning based feature selection for remote sensing scene classification. *IEEE Geosci. Remote Sens. Lett.* **12**, 2321–2325.