AGILE: GIScience Series, 2, 11, 2021. https://doi.org/10.5194/agile-giss-2-11-2021
Proceedings of the 24th AGILE Conference on Geographic Information Science, 2021.
Editors: Panagiotis Partsinevelos, Phaedon Kyriakidis, and Marinos Kavouras.
This contribution underwent peer review based on a full paper submission.

Extraction of linear structures from digital terrain models using deep learning

Ramish Satari[a], Bashir Kazimi[b] (corresponding author) and Monika Sester[b]

satari@hydromech.uni-hannover.de, kazimi@ikg.uni-hannover.de , sester@ikg.uni-hannover.de

[a]Institute of Fluid Mechanics and Environmental Physics in Civil Engineering, Leibniz Universität Hannover, Germany
[b]Institute of Cartography and Geoinformatics, Leibniz Universität Hannover, Germany

Abstract. This paper explores the role deep convolutional neural networks play in automated extraction of linear structures using semantic segmentation techniques in Digital Terrain Models (DTMs). DTM is a regularly gridded raster created from laser scanning point clouds and represents elevations of the bare earth surface with respect to a reference. Recent advances in Deep Learning (DL) have made it possible to explore the use of semantic segmentation for detection of terrain structures in DTMs. This research examines two novel and practical deep convolutional neural network architectures i.e. an encoder-decoder network named as SegNet and the recent state-of-the-art high-resolution network (HRNet). This paper initially focuses on the pixel-wise binary classification in order to validate the applicability of the proposed approaches. The networks are trained to distinguish between points belonging to linear structures and those belonging to background. In the second step, multi-class segmentation is carried out on the same DTM dataset. The model is trained to not only detect a linear feature, but also to categorize it as one of the classes: hollow ways, roads, forest paths, historical paths, and streams. Results of the experiment in addition to the quantitative and qualitative analysis show the applicability of deep neural networks for detection of terrain structures in DTMs. From the deep learning models utilized, HRNet gives better results.

Keywords: Semantic Segmentation, Digital Terrain Models, GIS: Geographic Information Systems, Deep Learning

1 Introduction

The extraction of information, including linear structures, plays an important role in a wide range of disciplines where topographic features are used in spatial analysis, e.g. hydrological applications and archaeological applications. Remote sensing techniques such as Airborne Laser Scanning (ALS) are used to collect 3D data from large areas simply by measuring the range and reflectivity of objects on their surface. ALS is often referred to as light detection and ranging (LIDAR). Data of this type are stored as point clouds, in order to leverage such data to identify structures on the terrain, a rasterized product named DTM is used. DTM is a filtered version of ALS data which preserves the information on the terrain points, i.e. it contains elevation information and surface characteristics such as ditches, forest paths, hollow ways, path ways, roads, etc. Usually some of the most meaningful features needed to be extracted from DTM data are slope, aspect, surface curvature, roughness of the terrain and distance from water reservoirs (bodies). Traditionally, within the field of feature extraction many deterministic algorithms have been designed to identify specific objects or extract specific linear features leveraging ALS data. Methods used for line extraction from DTMs can be divided into two categories: Physical water flow simulation-based methods (O'Callaghan and Mark, 1984, Jenson and Domingue, 1988, Quinn et al., 1991, Tarboton, 1997) and geometrical morphological analysis-based methods (Chang et al., 1998, Gülgen and Gökgöz, 2004, Zhang et al., 2013, Zou and Weng, 2017, Peucker and Douglas, 1975) as mentioned in (Tsai, 2019). The former conduct running water simulation on terrain surface and the latter, geometric approach identify feature candidates for extracting terrain feature lines.

Easier collection and storage of huge dataset and recent advance in hardware have given researchers the opportunity to exploit Machine Learning (ML) techniques. Initially, classical ML algorithms were used for the task of detecting and identifying objects in DTM data. Hitherto, a great number of ML algorithms has been published in the literature for the sole purpose of detecting and identifying objects in products of ALS data. The probability of palustrine wetland in digital elevation data were predicted by (Maxwell et al., 2016) using Random Forest (RF), moreover (Naghibi et al.,

2015) used Boosted Regression Tree (BRT), Classification and Regression Tree (CART) and RF in digital elevation data for modeling and mapping of groundwater spring potential. Support Vector Machines (SVMs) and RF are also exploited for detection of forested landslide in DTM data (Li et al., 2015, Paw luszek-Filipiak and Borkowski, 2016). However, the performance of the most ML algorithm depends solely on how accurately the features are extracted and identified by the experts. The need for an expert in order to select, extract and hand engineer the features, from the raw data, makes the process rather costly. With such challenges, requirements and vast amount of data acquired daily, the need for a much efficient method has been addressed by researchers all over the world.

Recently, a subfield of ML referred to as Deep Learning (DL) has come into play and shown improved performance when compared to the classical ML methods in many applications such as image classification, localization and detection, speech recognition, machine translation and many advanced assistance systems. The advantage of using DL is that it doesn't require a priori extraction of features i.e. learns and extracts features automatically from the dataset. However, training an accurate and efficient DL model requires a huge amount of data, in order to prevent the model from overfitting (i.e. to prevent model for memorizing the training data). In the literature, many datasets are made public. Some benchmark image datasets used for computer vision application such as, medical imaging technology, face recognition and autonomous driving are Labelme (a large dataset of annotated images) (Russell et al., 2008), ImageNet (Deng et al., 2009), Cityscape (Cordts et al., 2016), LIP (annotated human images) (Gong et al., 2017), PASCAL-Context (Mottaghi et al., 2014), and many more.

The main objective of this study is to exploit novel state-of-the-art techniques in deep learning for Semantic Segmentation, in order to extract linear structures from ALS data, e.g. DTM. Following the preprocessing approach proposed by (Kazimi et al., 2019b), SegNet (Badrinarayanan et al., 2017a, Badrinarayanan et al., 2017b) and HRNet (Ke et al., 2019) architectures are examined for a pixel-wise binary classification in order to validate the applicability of the proposed approaches. In the first step, the networks are trained to distinguish between points associated to the linear structures and those of background. In the second step, multi-class segmentation is carried out on the same DTM dataset, where the model is trained to detect and categorize the linear structures into one of the classes: hollow ways, roads, forest paths, historical paths, and streams.

The rest of the paper is organized as follows. Section 2 outlines related works. Section 3 briefly introduce the proposed methods. Section 4 discusses the architecture and hyperparameters of the model and describes the properties of the dataset used in this paper. Section 5 shows the results of the experiment, and the quantitative and qualitative analysis of the results. Section 6 gives details of Data and Software Availability (DASA), and finally Section 7 concludes this paper and lists possible future research in this direction.

2 Related Work

Traditionally, in the field of Geographical Information Systems (GIS) many tasks such as, extraction of linear structures (features), skeleton lines (ridge and valley lines) and terrain synthesis (O'Callaghan and Mark, 1984, Jenson and Domingue, 1988, Chang et al., 1998, Zhang et al., 2013) have been conducted using deterministic algorithms. Among them, DTM data are used by mainstream methods such as physical water flow simulation and geometric approaches (Gülgen and Gökgöz, 2004). The former method has many variations e.g. single flow direction or D8 approach (O'Callaghan and Mark, 1984) and multiple flow direction approach, D-infinity approach which are incorporated in many GIS softwares (Quinn et al., 1991, Tarboton, 1997). Although these algorithms have been used by many researchers, they have many drawbacks and disadvantages. Expert knowledge of the topic is always needed, when conducting research and studies using these traditional methods. Moreover, it has become impossible to catch up with the processing of the data. These huge sets of data and recent hardware advances have posed new opportunities for the researchers to utilize the state-of-the-art ML and DL methods.

In the recent time, DL has proven to be successful in many tasks related to the computer vision with applications in search engines, image understanding, drones and self-driving cars among others. Core to these applications are image classification, localization, detection, semantic segmentation and instance segmentation. Different types of model architecture have been proposed for DL, including Feed-forward Neural Networks (FNNs), Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). As the most established architecture among various DL methods, CNNs has recently become a

dominant method in computer vision. There are many prominent DL architectures that make use of CNNs. The most well-known examples are:

- **LeNet**. The first successful applications of CNN developed by LeCun et al. (LeCun et al., 1990) and was used to read zip codes, digits, etc.
- **AlexNet.** The first work that made CNN popular in computer vision and was developed by Krizhevsky et al. (Krizhevsky et al., 2017).
- **GoogLeNet**. The winner of ILSVRC 2014, developed by Szegedy et al. (Szegedy et al., 2015) from Google. Their development of an Inception Module dramatically reduced the number of Parameters in the network (4M when compared to AlexNet with 60M)
- **ResNet.** Residual Networks, the winner of ILSVRC 2015 was developed by Kaiming He et al. (He et al., 2015) and are one of the most commonly used CNN models. The architecture features special skip connections and a heavy use of batch normalization, at the end of the network it is also missing the fully connected layer.
- **Xception**. Inspired by Inception, and developed by François Chollet (Chollet, 2016), this architecture significantly outperforms Inception V3 on larger image classification dataset.

In image segmentation, the current state-of-the-art results are also obtained using the CNN architectures such as, Fully Convolutional Neural Network (FCN) (Long et al., 2015), UNet (Ronneberger et al., 2015), DeepLabv1 (Chen et al., 2018) and SegNet (Badrinarayanan et al., 2017a, Badrinarayanan et al., 2017b) among others. Not long ago, high resolution network (HRNet) (Sun et al., 2019) was proposed that maintains high resolution throughout the whole process and has proved to be superior for tasks such as object detection and semantic segmentation (Sun et al., 2019, Wang et al., 2020).

In the field of remote sensing, DL has also gained popularity in pattern recognition from ALS raster data. In recent times, several CNN-based architectures were proposed in the literature (Marmanis et al., 2015, Marmanis et al., 2016, Hu and Yuan, 2016, Rizaldy et al., 2018, Politz et al., 2018, Kazimi et al., 2018, Torres et al., 2018, Kazimi et al., 2019b, Du et al., 2019, Kazimi et al., 2020), in which Digital Elevation Model (DEM) were used as input data instead of the usual natural images. Marmanis et al. (Marmanis et al., 2015) proposed a deep classification model for detecting objects above-ground using DEM as input data. Their developed network is capable of differentiating between high standing structures e.g. trees and high-rise buildings. Further developing their work using networks pre-trained on regular images, they concluded that coupling DEM data with a real image can produce accurate segmentation masks (Marmanis et al., 2016). Hu et al. (Hu and Yuan, 2016) also developed an architecture based on CNNs to extract relevant data in order to produce DTMs from ALS data. To determine multi-class segmentation Rizaldy et al. (Rizaldy et al., 2018) applied a CNN on loosely processed DTM data. Not long ago, Torres et al. (Torres et al., 2018) applied DL in DEM data to identify mountain summits. For further applications of CNN using ALS data, the interested reader is referred to recent works in (Politz et al., 2018) and Kazimi et al. (Kazimi et al., 2018, Kazimi et al., 2019b, Kazimi et al., 2019a, Kazimi et al., 2020). In order to detect objects and their location in archaeological sites, a two-stage approach were proposed in (Kazimi et al., 2018). In the first stage they developed and trained multiple classifiers with different input sizes and in the second stage they use a sliding window approach to scan large DTMs with each classifier and merge their outputs to generate a heat map for each class. Following their work in (Kazimi et al., 2019b), they generated a segmentation mask directly from labeled DTM data using an encoder-decoder model named DL4DTM. Their proposed model is a modified version of DeepLabv3+, in which the output size is changed to be smaller than that of the input, moreover their proposed preprocessing approach is using a min-max normalization on each input in order to successfully process DTM data. It is worth noting that there also exist many publications which modify and combine different networks for a better performance and less computational effort, for instance, to detect structure in

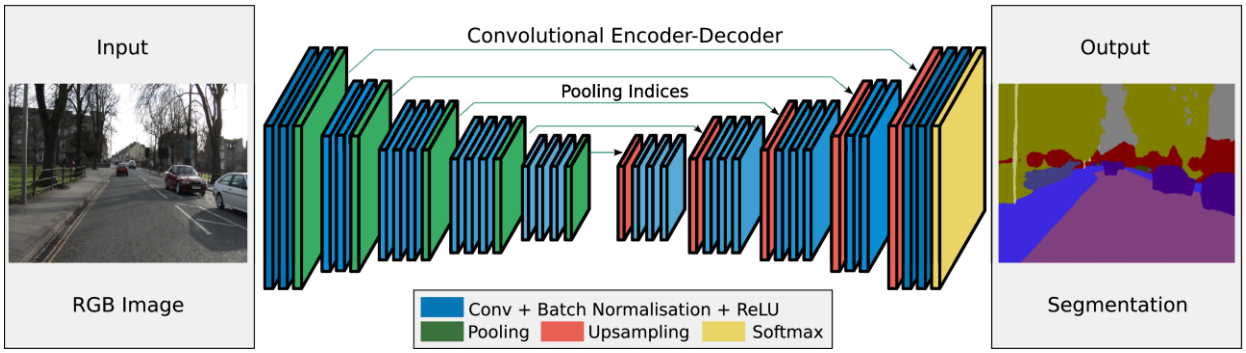


Figure 1: An illustration of a CNN architecture termed SegNet for image segmentation. Figure is adapted from (Badrinarayanan et al., 2017a).

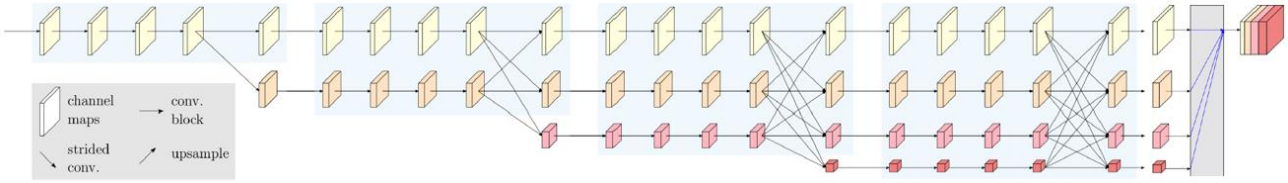


Figure #2: An illustration of HRNet. There are 4 stages. The 1st stage consists of high resolution convolutions and the rest repeats two-, three- and four-resolution blocks, respectively. Figure is adapted from (Sun et al., 2019)

DEM data Kazimi et al. (Kazimi et al., 2020) recently proposed a Multi-Modal High Resolution network named MM-HR which is based on HRNet (Ke et al., 2019) and multi-modal deep learning approach (MM) (Du et al., 2019). Their proposed architecture with fewer parameter outperformed the MM architecture on the dataset of archaeological mining structures from Harz.

In this project, based on literature analysis and our preliminary experiments with an intent to contribute to the field of remote sensing following recent works in (Kazimi et al., 2019b, Kazimi et al., 2020), we explore the use of semantic segmentation by doing experiments using two different CNN architectures, an encoder-decoder network named SegNet (Badrinarayanan et al., 2017a), and a high resolution network called HRNet (Sun et al., 2019) for the sole purpose of extraction of linear structures in DTMs.

3 Method

The main contribution of this research is proposing and confirming the use and efficiency of state-of-the-art models referred to as SegNet (Badrinarayanan et al., 2017) and HRNet (Sun et al., 2019). Additionally, we have also conducted a classical approach using ArcGIS, in order to create a ground for comparison with the proposed methods and also give a glimpse of what it takes to conduct these tasks using a classical method. Details of the proposed classical method and deep learning models are given in the following sections.

3.1 Classical Approach

Classical approaches leverage deterministic algorithms, to extract linear structures from ALS data such as DEM and DTM. In this research we are using ArcGIS spatial analyst tools for conducting experiments on our dataset. Following steps are necessary, in order to extract meaningful results from the DTM.

- **Run the Fill tool** - the Fill tool removes the imperfections from DTM data by filling the sinks in a surface raster.
- **Run the Flow direction tool** - this tool creates a raster flow direction from each cell to the steepest downslope neighbor.
- **Run the Flow accumulation tool** - it calculates accumulated flow as the accumulated weight of all cells.

After a successful execution of the aforementioned tools on the DTM data, the obtained results can be manipulated for a better visualization, the color of the accumulation lines can be changed. Furthermore, if we want to have polylines then another tool called, Raster to Polyline tool needs to be executed.

Please note that this process identifies fluvial structures in a DTM, therefor we expect a good performance of the

operators with respect to those objects – as opposed to other objects such as roads.

3.2 Encoder-Decoder Architecture

As an encoder-decoder architecture, a novel and practical deep fully convolutional neural network architecture for semantic pixel-wise segmentation named SegNet (Badrinarayanan et al., 2017) is chosen. An encoder-decoder architecture is typically used to improve generality and computational efficiency. In such architectures, an encoder compresses the data into a highly generalizable state and decoder upsamples it in order to produce a segmentation mask (output of the model) of the same size as that of the input. The function of downsampling is to not only reduce the size of the output but also to reduce the number of parameters to be calculated at each successive convolutional layer. As the most common downsampling technique, max-pooling extracts the maximum values from the portions of the image and saves its location. To obtain the high resolution segmentation mask, the output must be upsampled, and this can be done using a decoder. In order to achieve this upsampling a decoder uses a combination of max-unpooling, residual connections and convolutions. The combination allows the network to maintain information from multiple stages throughout upsampling process. The previously memorized locations or indices of the corresponding feature map is then used by decoder in order to upsample its input feature map.

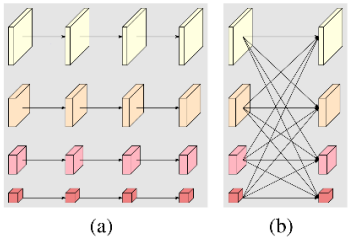
3.2.1 SegNet

SegNet (Badrinarayanan et al., 2017) is a fully convolutional neural network that consists of two distinct parts. The first part is convolution and downsampling (encoding) and the second part is the convolution and upsampling (decoding). These parts namely, convolution, downsampling and upsampling are the most important parts of CNN for semantic segmentation task. An illustration of this architecture is depicted in Fig. 1, in which it is shown that each encoder layer has a corresponding decoder layer. In simple words, SegNet first downsamples the image by an encoder network and then upsamples it by a decoder network. The output of the final decoder is then fed into a multi-class Softmax classifier in order to generate class probabilities for each pixel independently. The encoder part of this network has 13 Convolutional layers and hence the decoder part also has 13 layers. The fully connected layer is discarded for the sole purpose of retaining high resolution feature maps at the deepest encoder output. This in fact results in a significant reduction of parameters in the SegNet encoder network when compared with other architectures. Each layer in the encoder network performs convolution with a filter bank to generate a set of feature maps which are then batch normalized. After normalization an element-wise rectified linear unit (ReLU) is used. Following that a max-pooling with a 2×2 window and stride 2 is performed, thus the resulting output is sub-sampled by a factor of 2.

3.3 High Resolution Network (HRNet)

HRNet, or High Resolution Network, is a general purpose convolutional neural network which is essentially considered for position-sensitive vision problems, e.g. human pose estimation, object detection and semantic segmentation. Unlike existing state-of-the-art frameworks, which recover high resolution representations from low resolution representations outputted by a network (e.g. ResNet (He et al., 2015)) and optionally intermediate medium-resolution representations, e.g. SegNet (Badrinarayanan et al., 2017a, Badrinarayanan et al., 2017b), HRNet maintains high resolution representation through the whole process. This can be done by connecting high to low resolution convolution streams in parallel and by repeatedly exchanging the information across parallel convolutions. Hence, the resulting representation is semantically richer and spatially more precise. High resolution representation was initially developed for human pose estimation (Sun et al., 2019), and soon due to their state-of-the-art performance, it became popular in many applications including semantic segmentation (Wang et al., 2020, Ke et al., 2019). In semantic segmentation, the proposed HRNet (Ke et al., 2019) has achieved state-of-the-art results with similar model sizes and less computational effort on many different datasets (e.g. PASCAL Context, Cityscapes and LIP).

The architecture of HRNet is depicted in Fig. 2. It uses four stages, where the first stage consists of high resolution convolutions and the remaining stages are formed by repeating modularized multi-resolution blocks. This block consists of a multi-resolution group convolution and a multi-resolution convolution as illustrated in Fig. 3.



(a) (b)

Figure 3: Multi-resolution block: (a) multi-resolution group convolution and (b) multi-resolution convolution. Figure is adapted from (Sun et al., 2019).

In semantic segmentation this architecture was used on two different datasets (Pascal Context and Cityscapes). To measure their performance, the mean of class-wise intersection over union (mIoU) were adopted as the evaluation metric.

3.4 Models for Extraction of Linear Features

The complexity of DTM data are relatively low when compared with other image data which contain complex features and patterns. Therefore, deliberate modifications to the proposed complex models such as SegNet (Badrinarayanan et al., 2017) and HRNet (Sun et al., 2019) are performed. These modifications are necessary since these models are designed to generate complex segmentation masks and are probably more complex and may result to a lengthy training time and other issues such as overfitting. Hence, smaller versions of the models are used. Our first model, SegNet as stated before is consisted of many conv-blocks and each block has many layers, each layer is consisted of a Conv, batch normalization and ReLU see Fig. 1. In this paper, we keep only one such layer of each conv-block. On the other hand, for the 2nd model (HRNet), the employed modifications are rather simpler. We only reduced the number of blocks at each stage from four to two blocks only.

4 Experiments

In this section, we aim to use a simpler version of SegNet and HRNet, to do semantic segmentation on DTM data. The following sections give details of the model architecture, the dataset used, and the experimental setup for both cases, binary classification and multi-class segmentation.

4.1 Binary Classification

4.1.1 Dataset

The DTM dataset used in the binary classification task is acquired from the Harz Region and has a resolution of 0.5 meters per pixel. They are labeled into two different classes: background and linear features. Fig. 4 depicts a tile of training region. Each example has a size of 128×128 and the entire dataset is then divided into training, validation and test subsets.

(a) RGB (b) Ground-truth

Figure 4: Binary Classification – A 128x128 pixel patch of training region where linear structures are colored blue and the background is black.

4.1.2 Experimental Setup

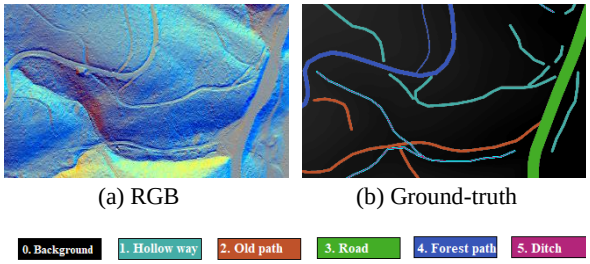
The architectures used in the binary classification task are SegNet (Badrinarayanan et al., 2017) and HRNet (Sun et al., 2019). Both models are trained and evaluated on the same dataset using the same settings. The models are trained with an input size of 128×128 for 50 epochs and a batch size of 20. As a loss function, binary cross-entropy is chosen, where it is then minimized using an optimizer called Adam (Kingma, Ba, 2014). The main metric for evaluation of the models performance is F1-Score. At each epoch the model is saved to disk, and the best model is then used to scan the test region and produce pixel-level predictions. The quantitative and qualitative analysis of the experiments are detailed in the results and discussion section.

4.2 Multi Class Segmentation

4.2.1 Dataset

The objective of the second part of the paper is to categorize the detected linear features in DTMs as one of the classes: hollow ways, roads, forest paths, historical paths, streams and background. The dataset used here is the same data used in the binary classification task, however they are labelled differently. Fig. 5 illustrates the DTM and its ground truth label for linear features. Each color in the ground truth label represents a distinct class. Apart from the labelling, the

datasets undergo the same procedures as explained in the binary classification section.



(a) RGB (b) Ground-truth

(c) Ground-truth labels

Figure 5: Multi-Class Segmentation – A portion of training region.

4.2.2 Experimental Setup

After a profound qualitative and quantitative analysis and experiments on the binary classification task of this paper which is given in Section results and discussion, it is decided to perform the multi-class segmentation task, using only the state-of-the-art high-resolution network (HRNet). The architecture of our network for multi-class task is mostly identical to that of binary classification, however the only difference appears in the final layer. In the binary case, the final layer has a feature map size of 1 ($128 \times 128 \times 1$) with Sigmoid activation function for binary cross-entropy (BCE). On the other hand, for multi-class segmentation task, the final layer has a feature map with size equal to the number of categories which is 6 ($128 \times 128 \times 6$) with a Softmax activation function. The model is then trained to minimize categorical cross-entropy targeting the maximum categorical accuracy, and the final layer uses a Softmax function producing class probabilities for a given example (image). As for the evaluation of the performance of the model, metrics such as mIoU, F1-score, accuracy among others are chosen.

5. Results and Discussion

In this section, we first conduct a binary classification task where we evaluate and compare the performances of both classical and deep learning methods on the same dataset. Additionally, a multi-class segmentation task using only high resolution network (HRNet) is also conducted, in order to categorize the detected linear features.

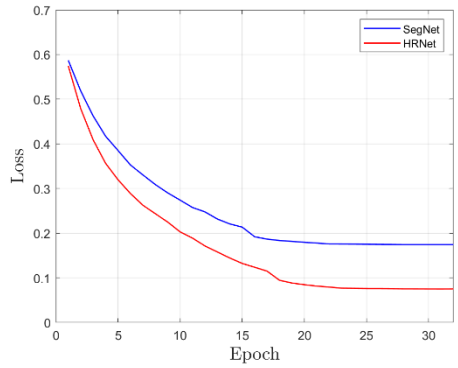
5.1 Binary Classification

As discussed in the previous sections of this project, the goal of binary-classification task is to detect the linear features from DTM data using both traditional and DL methods. As for classical method, ArcGIS software is used to extract some useful linear features. As for the proposed CNN architectures, we evaluate and compare the performances of both models: SegNet and HRNet on the same dataset. Moreover, their predictions on unseen data are also analyzed. The evaluation metrics of both models on the same dataset can be seen in Tab. 1, where different metrics such as accuracy, F1-score, precision and recall, are listed.

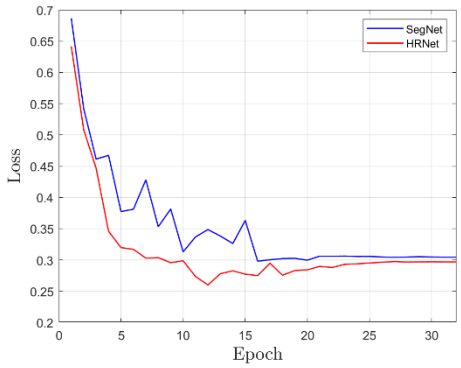
Name	SegNet	HRNet
Accuracy	0.8789	0.9065
Precision	0.7678	0.8183
Recall	0.6417	0.7353
F1-Score	0.6983	**0.7738**
Loss	0.3088	0.2582

Table 1: Binary Classification - Results of the evaluation metrics for both models on the same test region.

For imbalanced classes such as this one where the distribution of examples in the training dataset across the classes are not equal i.e. number of background classes are orders of magnitude higher than pixels belonging to linear features, accuracy becomes an unreliable metric, to measure the model performance. For example, in the case of spam detection in emails, where there are 90 spam emails and 10 not-spam, if a model only predicts not-spam for any input email, without learning anything, an accuracy of 90% can be achieved, which is completely misleading. The quantitative analysis show not only that HRNet performs better than SegNet, but it also gives better results, as listed in Tab. 1. It is important to consider F1-score as our main metric given we have a 128×128 input tile where most of region has a value of zero. F1-score is the harmonic mean of precision and recall. The greater the score, the better is the performance of the model. Thus high-resolution network (HRNet) with a score of 77% outperforms the encoder-decoder architecture (SegNet). Additionally, we plot the results of training the two models in the experiments from Section 4 in Fig. 6. The trained models are then used to perform segmentation on large DTMs of test region using a sliding window approach. Example test regions, and the predictions by ArcGIS and our two models are shown in Fig. 7, Fig. 8 and Fig. 9. These regions and patches are extreme cases, cases where ground-truth is perfect but predictions are not and cases where ground-truth is incomplete but predictions are better.



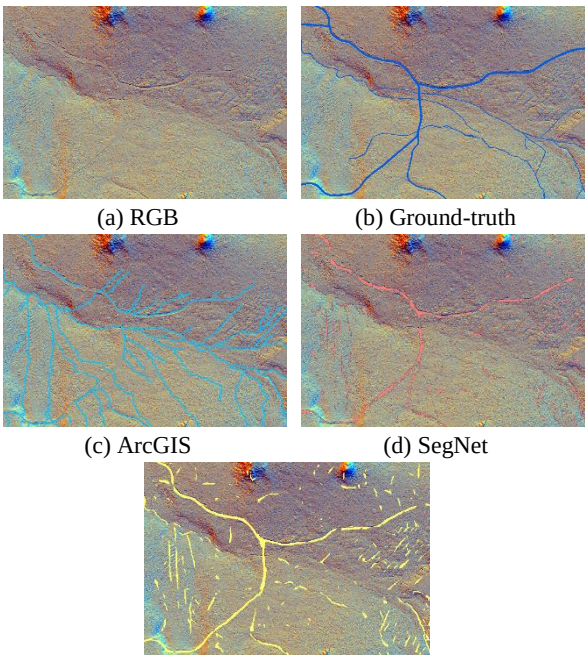
(a) Training Loss



(b) Validation Loss

Figure 6: Performance of SegNet and HRNet models on our dataset.

Fig. 7 illustrates a larger portion of DTM used in the test region with overlay of linear features predicted by ArcGIS, SegNet and HRNet on RGB. It can be observed from our predictions when compared with the ground truth labels, that our models did a good job of extracting the major linear structures. Even though ArcGIS has created these linear structures in a more continuous and consistent manner, the size of linear structures predicted by HRNet is more precise compared to ArcGIS, additionally HRNet results are better than SegNet. Moreover, there are small discontinuous structures predicted by our models which are not covered in the ground-truth labels, this in fact shows that ground-truth labels are incomplete. These structures that are not covered in the ground-truth and are predicted by our models have tremendously affected the quantitative evaluation results. As expected the ArcGIS method tries to find fluvial patterns in the DTM – which does not necessarily correspond to our ground truth.

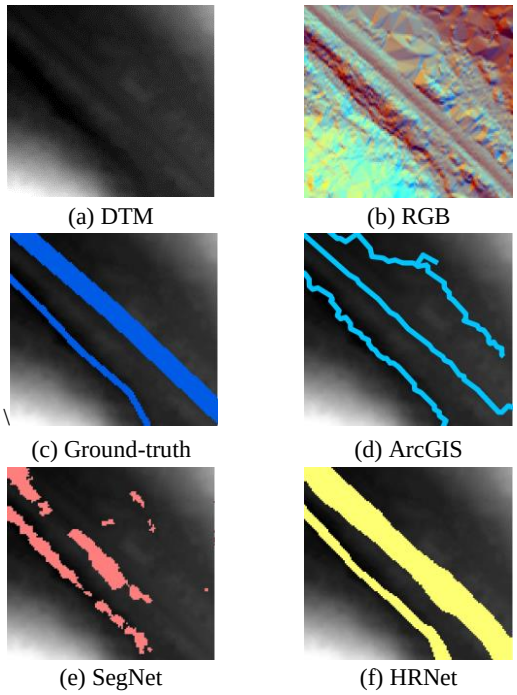


(a) RGB (b) Ground-truth

(c) ArcGIS (d) SegNet

(e) HRNet

Figure 7: Binary Classification – A large portion of test region. In #b blue color indicates the ground-truth labeling.



(a) DTM (b) RGB

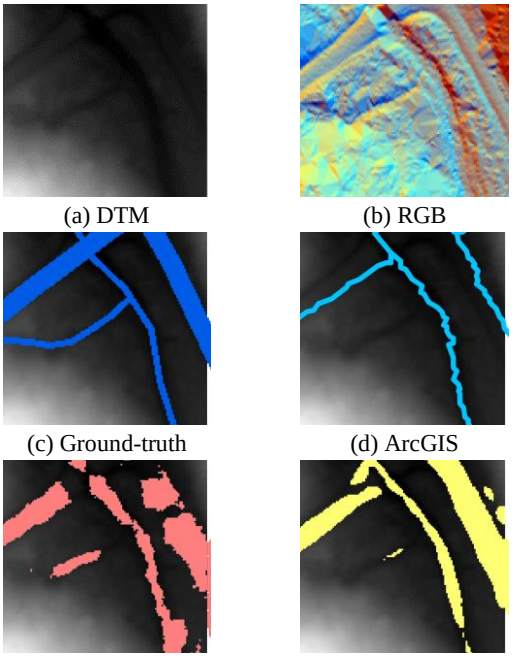
(c) Ground-truth (d) ArcGIS

(e) SegNet (f) HRNet

Figure 8: Binary Classification – A 128x128 pixel patch of test region.

Individual patches are also evaluated and are shown in Fig. 8 and Fig. 9. In Fig. 8 it can be observed, HRNet has predicted the linear structures not only in a continuous manner but also accurately predicted their size. Nonetheless, there are regions where our models didn't perform well, as a result some discontinuities in

the predicted segments can be seen see Fig. 9. Therefore, in the second part of this research which we will categorize these detected linear structures into classes, we use HRNet only.



(a) DTM (b) RGB (c) Ground-truth (d) ArcGIS (e) SegNet (f) HRNet

Figure 9: Binary Classification – A 128x128 pixel patch of test region.

5.2 Multi Class Segmentation

The goal of 2nd part of this paper is to categorize the detected linear features into classes: background, hollow way, old path, road, forest path and ditch. After a detailed study of the dataset and conducting experiments using binary labels, it was concluded that HRNet outperforms other approaches used in this research. Therefore, HRNet is employed in multi-class segmentation task. Tab. 2 lists the evaluation metrics of HRNet. The calculated mIoU score may not be very accurate measure of performance, given that most of our region has a value of zero. Therefore, F1-score is also computed to measure the performance.

mIoU	Precision	Recall	F1-Score	Loss
0.4174	0.8837	0.8963	0.8874	0.4201

Table 2: Multi-class Segmentation - Results of the evaluation metrics for high-resolution network.

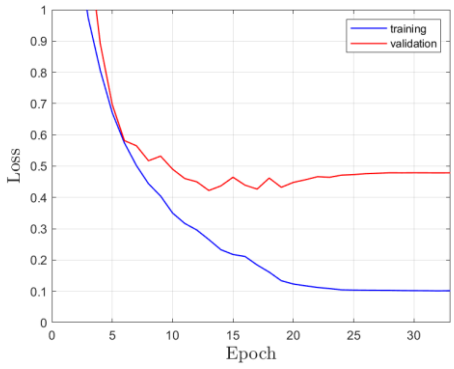
In general, it is important to figure out, on which classes our model performed well. For this purpose, evaluation metrics of individual class are listed in Tab. 3. Overall, our model has done well on class 0 and class 3 with an F1-score of around 95 and 87 percent, respectively. However, the model has poorly performed in categorizing the segments belonging to class 2 (old path). Results also show the model's performance decreases greatly on classes 4, 1 and 5 as well.

Class label	IoU	Precision	Recall	F1-Score
0	0.8952	0.9270	0.9630	0.9446
1	0.2073	0.4009	0.3745	0.3873
2	0.0324	0.3139	0.0373	0.0666
3	0.7648	0.8797	0.8552	0.8673
4	0.4453	0.6953	0.5580	0.6191
5	0.1593	0.2697	0.2787	0.2741

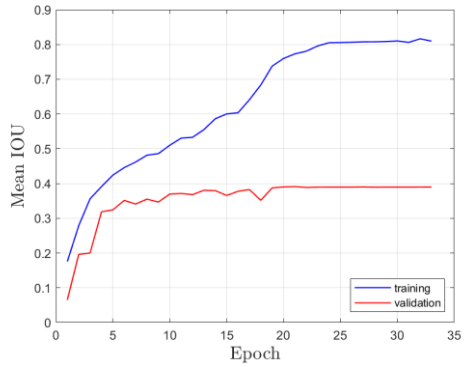
Table 3: Multi-class Segmentation - Results of the evaluation metrics for individual classes.

Additionally, training and validation loss along with the mean IoU for the test region predictions are shown in Fig. 10.

In a practical level, to assess the performance of the model a visual analysis of the predictions is needed. Fig. 11 depicts the same region as illustrated in the binary classification task. However, here the extracted features are categorized into different classes using HRNet only. It can be observed that HRNet successfully differentiates between features of different classes. As pointed out in the Tab. 3, the models perform well only on two classes: background (2) and road (3), its performance is satisfactory on forest path (4) with an F1-score of around 62% and on classes: ditch (5) and hollow way (1) it is below 50%. Unfortunately, the model performed poorly on categorizing old path (2), its F1-score is below 7%, although, its precision is bigger than its recall, meaning that the model returns more relevant results than irrelevant ones.



(a) Loss



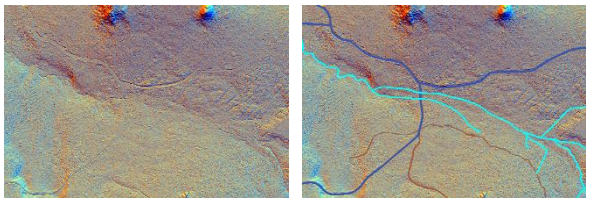
(b) mIoU

Figure 10: Performance of HRNet model on our dataset. (a) Loss during training and (b) Mean IOU

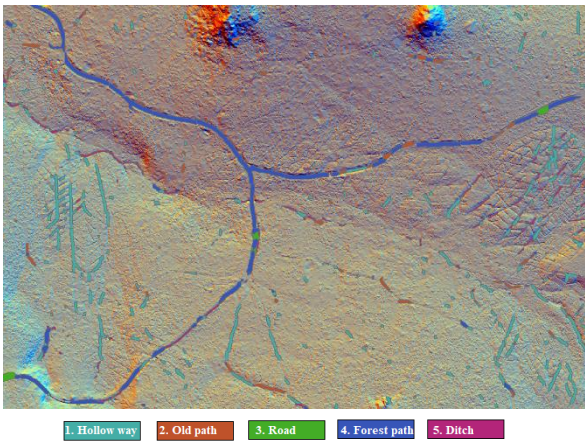
For a thorough interpretation of the predicted results, here we take a closer look into some portion of our region of interest to see, how the model has performed on detecting and categorizing hollow ways, old paths and forest paths.

Fig. 11 and Fig. 12 contain all the classes; it can be observed that our model has categorized accurately most of the detected linear structures into their respective class label. However, classes such as old path (2) and hollow way (1) are not categorized as such. The reason behind could be the similarity in the shape and position of these two classes and also the segments belonging to these classes in the ground-truth are thin in size. Therefore, given their size compared to other classes the model may suffice to learn other classes only. Additionally, in Fig. 12c, our model has detected many discontinuous linear structures in the middle of the region and labeled them as class hollow way. These segments are not covered in the ground-truth label and this can be one reason why our model has such low values in the quantitative evaluations.

Individual patches are also evaluated and our predictions are depicted in Fig. 13, Fig. 14 and Fig. 15.

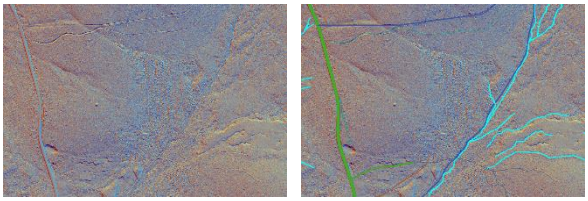


(a) RGB (b) Ground-truth

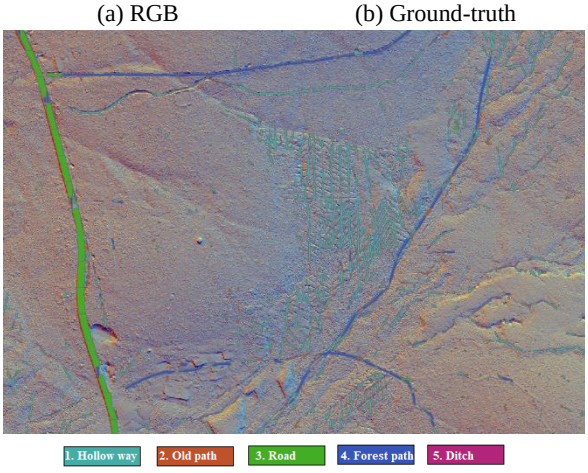


(c) HRNet

Figure 11: Multi-Class Segmentation – A large portion of test region. In #b blue color indicates the ground-truth labeling.

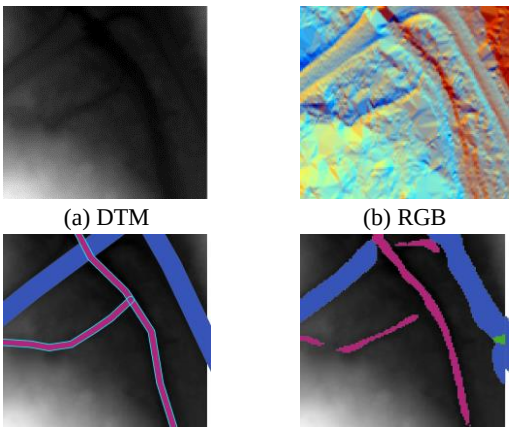


(a) RGB (b) Ground-truth



(c) HRNet

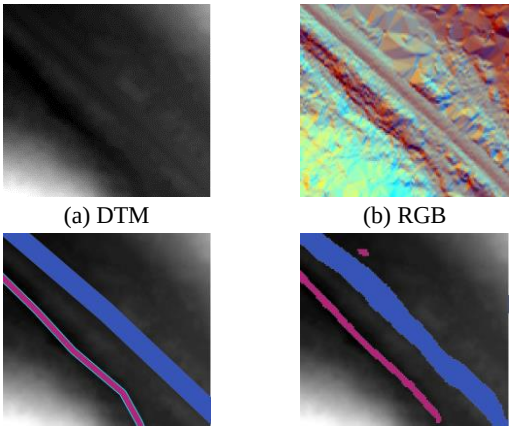
Figure 12: Multi-Class Segmentation – A large portion of test region.



(a) DTM (b) RGB

(c) Ground-truth (d) HRNet

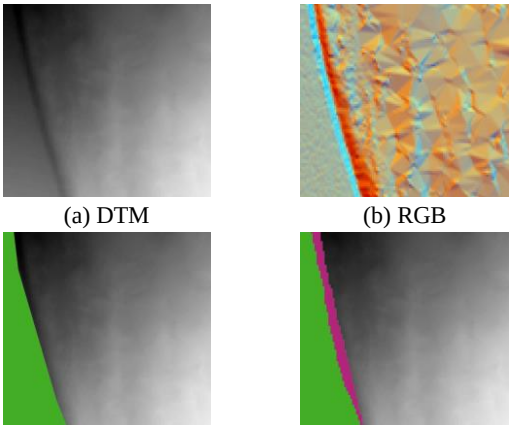
Figure 13: Multi-Class Segmentation – A 128x128 pixel patch of test region.



(a) DTM (b) RGB

(c) Ground-truth (d) HRNet

Figure 14: Multi-Class Segmentation – A 128x128 pixel patch of test region.



(a) DTM (b) RGB

(c) Ground-truth (d) HRNet

Figure 15: Multi-Class Segmentation – A 128x128 pixel patch of test region.

5.3 Limitations

Deep Learning approaches need huge number of dataset, to accurately learn and predict complicated problems. However, our initial objective in this project was not to achieve high accurate results, but to determine the efficiency of CNNs in detecting linear features from ALS point data or more precisely DTMs. To this end, we have confronted many challenges and limitations during this study, here a number of these limitations are discussed. Upon closer inspection of the dataset along with their ground truth labels, it can be observed that there are several regions or tiles in which the ground truth labels are missing, i.e. the dataset is incomplete. Another problem is that the ground truth labels are imprecise, the segments of the same class during labeling using ArcGIS software was buffered with the same length and this is not actually true for all the segments, since segments of the same class have different size. Since the DL approaches are identifying individual object pixels as patches, there is in general a lack of connectivity between them. This, however, is an important characteristic of linear features. Thus those results have to be post-processed (e.g. by connected compound analysis) in order to create consistent and connected objects. Last but not least, number of samples allocated to each classes are not equal i.e. we have and imbalanced class distribution. These aforementioned limitations tremendously affect the performance of our model in general.

6. Data And Software Availability

The training and validation data for this project is private to Lower Saxony State Office for Heritage. However, the source code for this experiments, the trained models and the test dataset to which the reported results correspond to are included in (Satari et al., 2021). The workflow underlying this paper was partially reproduced by an independent reviewer during the AGILE reproducibility review and a reproducibility report was published at https://doi.org/10.17605/osf.io/2sc7g.

7. Conclusion and Outlook

In this research, deep learning techniques were used to conduct experiments on DTM data acquired from the Harz region in Lower Saxony, in order to detect and categorize their linear structures or features. The neural network architectures for semantic segmentation

examined in this project are an encoder-decoder network named as SegNet and High Resolution Network (HRNet). Initially, a binary classification task was conducted, in which the networks were trained to distinguish between points belonging to linear structures and the background. This task was done not only to validate the applicability of the proposed architectures but also to identify the architecture with better overall performance. As a result, in the second part of this project, HRNet was chosen for multi-class segmentation task. The objective of this part of the project beside detection of linear features was their categorization as one of the 6 classes: hollow ways, roads, forest paths, historical paths, streams and background. Results of the experiments in conjunction with the quantitative and qualitative analysis validate the superiority and efficiency of Deep Learning (DL) techniques to extract and categorize linear features in DTM data. It was also concluded that the neural network methods particularly the High Resolution Network (HRNet) can be a better alternative to the classical methods due to its accuracy and straightforward approach. In summary, although we achieved reasonable results using only limited labeled data, with more labeled data, the model is expected to perform better both quantitatively and qualitatively.

Therefore, future research in this direction includes employing an efficient semi-supervised learning method for DNNs also known as pseudo-labeling method proposed by (Lee et al., 2013), in order to confront the challenge of having limited number of labeled data. Image processing algorithms such as Hough Transform and Region Growing can also be used to postprocess the predictions done by our models. Moreover, more weight could be given to classes with thin structures which results into a smaller ratio compared to the other pixel in the input patch.

References

Andrej Karpathy, J. and Li, F.: Lecture notes to cs231n: Convolutional neural networks for visual recognition, 2019.

Badrinarayanan, V., Kendall, A., and Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence., 2017a.

Badrinarayanan, V., Kendall, A., and Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence., 39, 12, 2481–2495, 2017b.

Chang, Y.-C., Song, G.-S., and Hsu, S.-K.: Automatic extraction of ridge and valley axes using the profile recognition and polygon-breaking algorithm, Computers Geosciences., 24, 83–93, 1998.

Chen, L., Papandreou, G., Kokkinos, I., Murphy, K., and Yuille, A. L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, IEEE Transactions on Pattern Analysis and Machine Intelligence., 40, 4, 834–848, 2018.

Chollet, F.: Xception: Deep learning with depthwise separable convolutions, 2016.

Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., and Schiele, B.: The cityscapes dataset for semantic urban scene understanding, In Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)., 2016.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L.: ImageNet: A Large-Scale Hierarchical Image Database, In CVPR09., 2009.

Du, L., You, X., Li, K., Meng, L., Cheng, G., Xiong, L., and Wang, G.: Multi-modal deep learning for landform recognition, ISPRS Journal of Photogrammetry and Remote Sensing., 158, 63–75, 2019.

Ghosh, S., Das, N., Das, I., and Maulik, U.: Understanding deep learning techniques for image segmentation, ACM Computing Surveys (CSUR)., 52, 1 – 35, 2019.

Gong, K., Liang, X., Zhang, D., Shen, X., and Lin, L.: Look into person: Self-supervised structure-sensitive learning and a new benchmark for human parsing, 2017.

Goodfellow, I., Bengio, Y., and Courville, A.: Deep Learning, MIT Press., http://www.deeplearningbook.org, 2016.

Gülgen, F. and Gökgöz, T.: Automatic extraction of terrain skeleton lines from digital elevation models, ISPRS C., 2004.

He, K., Zhang, X., Ren, S., and Sun, J.: Deep residual learning for image recognition, CoRR., 2015.

Hirt, C.: Digital terrain models. Encyclopedia of Geodesy, 2016.

Hu, X. and Yuan, Y.: Deep-learning-based classification for dtm extraction from als point cloud, Remote Sensing., 8, 2016.

Jenson, S. and Domingue, J. O.: Extracting topographic structure from digital elevation data for geographic information-system analysis, Photogrammetric Engineering and Remote Sensing., 54:1593–1600, 1988.

Kazimi, B., Thiemann, F., Malek, K., Sester, M., and Khoshelham, K.: Deep learning for archaeological object detection in airborne laser scanning data, 2018.

Kazimi, B., Thiemann, F., and Sester, M.: Object Instance Segmentation in Digital Terrain Models, 488–495, 2019a.

Kazimi, B., Thiemann, F., and Sester, M.: Semantic segmentation of manmade landscape structures in digital terrain models, ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences., 2, 7, 87–94, 2019b.

Kazimi, B., Thiemann, F., and Sester, M.: Detection of terrain structures in airborne laser scanning data using deep learning, ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences., 2, 493–500, 2020.

Ke, S., Zhao, Y., Jiang, B., Cheng, T., Xiao, B., Liu, D., Mu, Y., Wang, X., Liu, W., and Wang, J.: High-resolution representations for labeling pixels and regions, 2019.

Kingma, D. P. and Ba, J.: Adam: A method for stochastic optimization, 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings., 2015.

Krizhevsky, A., Sutskever, I., and Hinton, G. E.: Imagenet classification with deep convolutional neural networks, Commun. ACM., 60, 6, 84–90, 2017.

LeCun, Y., Boser, B., Denker, J., Henderson, D., Howard, R., Hubbard, W., and Jackel, L.: Handwritten digit recognition with a back-propagation network, In Touretzky, D., editor, Advances in Neural Information Processing Systems., 2, 396–404. Morgan-Kaufmann, 1990.

Lee, D.-H.: Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks, ICML 2013 Workshop: Challenges in Representation Learning (WREPL)., 2013.

Li, X., Cheng, X., Chen, W., Chen, G., and Liu, S.: Identification of forested landslides using lidar data, object-based image analysis, and machine learning algorithms, Remote Sensing., 7:9705–9726, 2015.

Long, J., Shelhamer, E., and Darrell, T.: Fully convolutional networks for semantic segmentation, In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)., 3431–3440, 2015.

Marmanis, D., Adam, F., Datcu, M., Esch, T., and Stilla, U.: Deep neural networks for above-ground detection in very high spatial resolution digital elevation models, ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences., 103–110, 2015.

Marmanis, D., Schindler, K., Wegner, J., Galliani, S., Datcu, M., and Stilla, U.: Classification with an edge: Improving semantic image segmentation with boundary detection, ISPRS Journal of Photogrammetry and Remote Sensing., 135, 2016.

Maxwell, A. E., Warner, T. A., and Strager, M. P.: Predicting palustrine wetland probability using random forest machine learning and digital elevation data-derived terrain variables, Photogrammetric Engineering Remote Sensing., 82, 6, 437 – 447, 2016.

Mottaghi, R., Chen, X., Liu, X., Cho, N.-G., Lee, S.-W., Fidler, S., Urtasun, R., and Yuille, A.: The role of context for object detection and semantic segmentation in the wild, In IEEE Conference on Computer Vision and Pattern Recognition (CVPR)., 2014.

Naghibi, S., Pourghasemi, H. R., and Dixon, B.: Gis-based groundwater potential mapping using boosted regression tree, classification and regression tree, and random forest machine learning models in iran, Environmental Monitoring and Assessment., 188, 2015.

O'Callaghan, J. and Mark, D.: The extraction of drainage networks from digital elevation data, Computer Vision, Graphics, and Image Processing., 27, 323–344, 1984.

Paw luszek-Filipiak, K. and Borkowski, A.: Landslides identification using airborne laser scanning data derived topographic terrain attributes and support vector machine classification, ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences., XLI-B8:145–149, 2016.

Peucker, T. and Douglas, D.: Detection of surfacespecific points by local parallel processing of discrete terrain elevation data, Graphical Models /graphical Models and Image Processing /computer Vision, Graphics, and Image Processing – CVGIP., 4, 375–387, 1975.

Politz, F., Kazimi, B., and Sester, M.: Classification of laser scanning data using deep learning, 2018.

Quinn, P., Beven, K., Chevallier, P., and Planchon, O.: The prediction of hillslope flow paths for distributed hydrological modeling using digital terrain model, Hydrological Processes., 5, 1991.

Rizaldy, A., Persello, C., Gevaert, C., Oude Elberink, S., and Vosselman, G.: Ground and multi-class classification of airborne laser scanner point clouds using fully convolutional networks, Remote Sensing., 10:1723, 2018.

Ronneberger, O., Fischer, P., and Brox, T.: U-net: Convolutional networks for biomedical image segmentation, CoRR., abs/1505.04597, 2015.

Russell, B., Torralba, A., Murphy, K., and Freeman, W.: Labelme: A database and web-based tool for image annotation, International Journal of Computer Vision., 77, 2008.

Saha, S.: A comprehensive guide to convolutional neural networks - the eli5way (www document), 2018.

Satari, R., Kazimi, B. and Sester, M.: Code for Extraction of Linear Structures from Digitial Terrain Models Using Deep Learning (Version v1.0.2). Zenodo. http://doi.org/10.5281/zenodo.4730574, 2021.

Simonyan, K. and Zisserman, A.: Very deep convolutional networks for large-scale image recognition, arXiv 1409.1556, 2014.

Sun, K., Xiao, B., Liu, D., and Wang, J.: Deep high-resolution representation learning for human pose estimation, In CVPR., 2019.

Szegedy, C., Wei Liu, Yangqing Jia, Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A.: Going deeper with convolutions, In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)., 1–9, 2015.

Tarboton, D.: A new method for the determination of flow directions and upslope areas in grid digital elevation models, Water Resources Research., 33, 309–319, 1997.

Torres, R. N., Fraternali, P., Milani, F., and Frajberg, D.: A deep learning model for identifying mountain summits in digital elevation model data, 2018 IEEE First International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)., 212–217, 2018.

Tsai, V.: Extraction of topographic structure lines from digital elevation model data, Civil Engineering Research Journal., 7, 2019.

Wang, J., Sun, K., Cheng, T., Jiang, B., Deng, C., Zhao, Y., Liu, D., Mu, Y., Tan, M., Wang, X., Liu, W., and Xiao, B.: Deep high-resolution representation learning for visual recognition, 2020.

Zhang, H., Ma, Z., Liu, Y., He, X., and Ma, Y.: A new skeleton feature extraction method for terrain model using profile recognition and morphological simplification, Mathematical Problems in Engineering., 2013.

Zou, K. and Weng, H.: Terrain feature line extraction by improved gradient-based profile analysis. International Journal of Signal Processing, Image Processing and Pattern Recognition., 10, 1–12, 2017.