



Review

Review of Automatic Processing of Topography and Surface Feature Identification LiDAR Data Using Machine Learning Techniques

Zahra Gharineiat , Fayez Tarsha Kurdi * and Glenn Campbell

School of Surveying and Built Environment, Faculty of Health, Engineering and Sciences,
University of Southern Queensland, Springfield Campus, Springfield, QLD 4300, Australia

* Correspondence: fayez.tarshakurdi@usq.edu.au

Abstract: Machine Learning (ML) applications on Light Detection And Ranging (LiDAR) data have provided promising results and thus this topic has been widely addressed in the literature during the last few years. This paper reviews the essential and the more recent completed studies in the topography and surface feature identification domain. Four areas, with respect to the suggested approaches, have been analyzed and discussed: the input data, the concepts of point cloud structure for applying ML, the ML techniques used, and the applications of ML on LiDAR data. Then, an overview is provided to underline the advantages and the disadvantages of this research axis. Despite the training data labelling problem, the calculation cost, and the undesirable shortcutting due to data downsampling, most of the proposed methods use supervised ML concepts to classify the downsampled LiDAR data. Furthermore, despite the occasional highly accurate results, in most cases the results still require filtering. In fact, a considerable number of adopted approaches use the same data structure concepts employed in image processing to profit from available informatics tools. Knowing that the LiDAR point clouds represent rich 3D data, more effort is needed to develop specialized processing tools.

Keywords: LiDAR; Machine Learning (ML); classification; modelling; point cloud



Citation: Gharineiat, Z.; Tarsha Kurdi, F.; Campbell, G. Review of Automatic Processing of Topography and Surface Feature Identification LiDAR Data Using Machine Learning Techniques. *Remote Sens.* **2022**, *14*, 4685. <https://doi.org/10.3390/rs14194685>

Academic Editor: Luis A. Ruiz

Received: 10 August 2022

Accepted: 17 September 2022

Published: 20 September 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

A Light Detection And Ranging (LiDAR) point cloud (airborne, terrestrial, static or mobile) is a list of 3D points covering the surface of a scanned scene. Topographical data obtained this way are rich in geometric features and lend themselves to the possibilities of automatic processing [1]. There are two major forms of automatic processing operations: automatic classification and automatic modelling [2]. Generally, one scanned scene will consist of classes that have different geometric natures or characteristics, e.g., an urban point cloud represents several classes such as terrain, buildings, vegetation, powerlines, roads, railways, and other artificial objects [3]. As each class in the scanned area will require a different modelling strategy depending on its specific geometric nature, e.g., the vegetation class modelling algorithm will need to be different from the building class modelling algorithm, it is necessary to classify the point cloud before starting the modelling stage.

In the first two decades since LiDAR technology's appearance, most of the suggested automatic processing algorithms belonged to the rule-based family [4]. In truth, a single rule-based algorithm actually consists of a list of procedures connected through a proposed workflow and depends on the physical structure of the point cloud [4]. Recently though, in the domain of topographical LiDAR data processing, the general trend has been to employ Machine Learning (ML) algorithms instead of rule-based ones, and the use of ML techniques has become a popular research topic [5].

Supervised ML algorithms assign observations to data classes previously generated, either manually or automatically, from the use of training data that could sometimes be

generated automatically [6]. Alternatively, unsupervised ML algorithms do not need training data and can be classified into four families: classification tree methods such as the Random Forest (RF) algorithms, grouping and separability methods such as Support Vector Machines (SVM), k-Nearest Neighbors (KNN), and rule application methods such as Convolutional Neural Networks (CNN) [7].

This paper reviews the state-of-the-art ML algorithms developed for topographical LiDAR data processing. The novelty of this paper is the classification and analysis of the ML algorithms according to four different dimensions. First, the methods of point cloud generation for input into ML approaches are analyzed and discussed. Second, the different concepts of point cloud structure that are commonly used are studied and compared. Third, the suggested approaches are classified according to the most employed ML techniques, and then the main ML techniques are summarized. Finally, the most current applications of ML techniques are classified and cited.

2. Input Data

Notwithstanding the quality of the employed laser scanning technology, airborne, terrestrial, static, or mobile, all methods allow the creation of a 3D point cloud that covers the scanned area. A LiDAR point cloud consists of a point list of co-ordinates X , Y , and Z defined in 3D Euclidean space. For each point, in addition to the three coordinates, laser intensity, waveform, and Red Green Blue (RGB) colors can be provided [8]. Furthermore, for the same scanned scene, additional data such as multispectral images, maps, and orthophotos can often be provided. As a result, in the literature, the suggested ML approaches for LiDAR data processing are not just limited to the LiDAR point cloud alone. The following subsections explain the different point cloud generation methods for input into ML algorithms.

2.1. LiDAR Point Clouds

The 3D point cloud is the primary output of a laser scanning operation (Figure 1). This subsection deals with approaches that use only the point cloud, whereas the approaches that use other additionally acquired data will be discussed in the following subsections. The obvious advantage of approaches that use only the LiDAR point cloud is that they are always available for use in all scanning projects. The point cloud does not just represent a simple list of 3D points in the Euclidian space, it may be used as the input data to create a Digital Surface Model (DSM) [1]. Furthermore, for each point, a list of neighboring points can be defined in 3D space [9–11], where all points included inside a sphere surrounding the focus point are considered, or in 2D space where all points included inside a cylinder surrounding the focus point are considered [5]. After this stage is completed, each point and its neighboring points allow for fitting a mean line or plane to analyze their relative topologic positions through several indicators such as standard deviation, mean square error, eigenvector, and eigenvalues [12]. Additionally, the eigenvector permits the calculation of a list of useful geometric features such as linearity, planarity, sphericity and change of curvature [13,14]. In this context, other approaches are to superimpose the point cloud on an empty 2D grid to allow for the analysis of the topological relationships between neighboring points [15], or assuming that they represent one object, using one LiDAR point and its neighborhood to allow calculation of a list of static moments that help to study some of their geometric characteristics [16]. While this has its uses, it is important to note that the employment of just the point cloud as input data does not produce promising results in the general case, e.g., when identifying roofs in an airborne LiDAR point cloud in an urban area, the range of roof point coordinates may be incorrectly allocated to an incorrect building because of underlying topography of the scanned area. That is why the application of the ML techniques in this case uses the point features instead of the point coordinates as input data [9]. Consequently, a long list of geometric features that can be calculated from the point cloud is needed to create a suitable environment to apply ML. The ML techniques that have been applied to airborne and terrestrial LiDAR point clouds are

shown in the next two subsections, with the use of laser intensity observations discussed in the subsection that follows those.

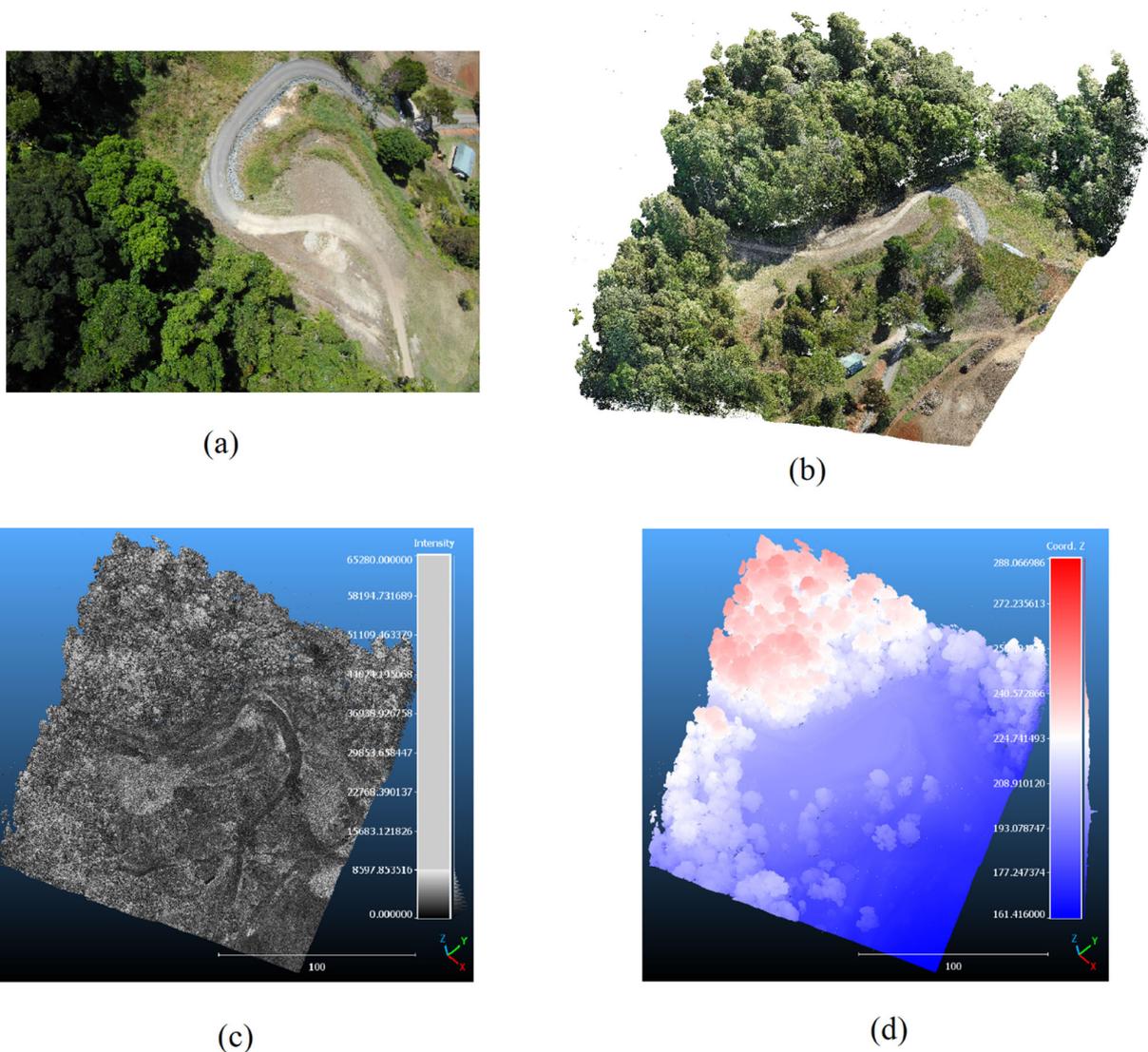


Figure 1. (a) Aerial image of scanned scene; (b–d) 3D LiDAR point cloud visualization (b) using RGB colors; (c) using laser intensity values; (d) using Z coordinate values.

2.1.1. Airborne LiDAR Point Cloud

Airborne LiDAR point clouds provide two obstacles to the applications of ML techniques: variation in point density within the scanned scene [11] and the large number of LiDAR points [17]. Point density plays a vital role in selecting the neighboring points for the calculation of point features [9]. Point density can vary markedly within the same point cloud with the location within the scanning strip, the terrain topography and the geometry, and the orientation of the scanned object with regard to the scan line all having an affect [8]. For a large area, the data volumes can be excessive, meaning the training step will place heavy demands on the computer capacity and processing time [17,18]. Lin et al. [19] and Mao et al. [20] developed approaches to mitigate this problem and classify an urban point cloud into nine classes: powerlines, low vegetation, impervious surfaces, cars, fences, roofs, façades, shrubs, and trees. In this context, Mao et al. [20] developed a Receptive Field Fusion-and-Stratification Network (RFFS-Net). An innovative Dilated Graph Convolution (DGConv) and its extension, the Annular Dilated Convolution (ADConv), are fundamental components of elementary building blocks. The receptive field fusion procedure was applied with the Dilated and Annular Graph Fusion (DAGFusion) component. Thus, the

detection of dilated and annular graphs with numerous receptive zones allows the acquisition of developed multi-receptive field feature implementation to improve classification accuracy. To efficiently extract only one class from the urban point cloud, Ao et al. [21] advised using a presence and background learning algorithm like a backpropagation neural network.

2.1.2. Terrestrial LiDAR Point Cloud

This subsection focuses only on the ML approaches that use a static or mobile terrestrial LiDAR point cloud as input data either indoors or outdoors. An indoor cloud may focus on certain scanned objects such as tables, chairs, decorative statues, and mechanical equipment [22,23] or it may carry out a panoramic scan [24] and use the LiDAR point cloud to then extract the individual objects. An urban outdoor LiDAR point cloud will most likely emphasize artificial or natural objects such as building facades and terrain [25] while a rural scene, like Zou et al. [26] examined, may use a terrestrial LiDAR point cloud of forestry areas to classify the tree species. In fact, most of the suggested approaches that use ML techniques to process terrestrial LiDAR data do not use additional data with the point cloud [17,27–32]. Point density variation has less influence in terrestrial when compared to airborne data. Nevertheless, some authors do use additional data as input, e.g., Xiu et al. [23] suggested a ML algorithm to process indoor point cloud represented by 9 dimensions: X , Y , Z , R , G , B , and normalized location. He et al. [25] developed a SectorGSnet framework for a ground segmentation of terrestrial outdoor LiDAR point clouds. This framework consisted of an encoder in addition to segmentation modules. It introduced a bird's-eye-view segmentation strategy that discretizes the point cloud into segments of different areas. The points within each partition are then fed into a multimodal Point-Net encoder to extract the required features. Li et al. [33] suggested a Rotation Invariant neural Network (RINet) which associated semantic and geometric features to improve the descriptive capacity of scanned objects and classify the terrestrial data into twelve classes.

Terrestrial laser scanning plays a major role in autonomous driving vehicles with Silva et al. [34] developing a Deep Feature Transformation Network (DFT-Net) involving a cascading mixture of edge convolutions and feature transformation layers to capture the local geometric features by conserving topological relationships between points. Alternatively, self-learning algorithms appear as a practical solution to understand the correspondence between adjacent LiDAR scan scenes [35]. Nunes et al. [36] used a momentum encoder network and a feature bank in a self-learning approach [37,38] that aimed to learn the structural context of the scanned scene. This approach applies the contrastive loss over the extracted segments to distinguish between similar and dissimilar objects. Finally, Huang et al. [39] used an unsupervised domain adaptation ML to classify terrestrial LiDAR data and suggested using Generative Adversarial Network (GAN) to calculate synthetic data from the source domain, so the output will be close to the target domain.

2.1.3. Point Cloud and Laser Intensity

In practice, LiDAR systems measure and provide the laser pulse return intensity (Figure 1c). The intensity of emitted laser pulse is greater than the intensity of the reflected laser pulse and with the difference being dependent on the double distance trajectory in addition to the nature of the reflecting surface off which the pulse has returned [40]. Unlike the RGB-measured values of the point cloud, the intensity could be detected regardless of the illumination and can be provided in both airborne and terrestrial LiDAR. Some authors have used the intensity and the 3D point cloud together as input data into their ML algorithms.

In this regard, Wen et al. [41] proposed a Directionally constrained fully Convolutional Neural network (D-FCN) where the input data were the original 3D point cloud in addition to the LiDAR intensity. Since road line markings have a higher reflectance, and hence higher intensity value than the surrounding ground, Fang et al. [42] considered the 3D LiDAR

point cloud and the laser intensity as input data to their ML algorithm. Wang et al. [29] employed the intensity component in semantic outdoor 3D terrestrial dataset to achieve the cloud segmentation using Graph Attention Convolution (GAC) and Murray et al. [43] calculated a 2D image from the intensity component of LiDAR data. This image was used as input data for the CNN algorithm and then for the SVM.

2.2. Point Cloud and Imagery

In the image processing domain, many algorithms for feature extraction from images have been implemented where the image's spatial and textural features were extracted using mathematical descriptors, such as histograms of oriented gradients and SVMs [44]. The combination of LiDAR data with high-resolution images can provide highly relevant data for the analysis of scanned scene characteristics [45]. Indeed, numerous authors develop classification ML networks using LiDAR point clouds in addition to digital images as input data. Nahhas et al. [46] employed orthophotos in addition to airborne LiDAR point clouds to recognize the building class by using an autoencoder-based dimensionality reduction to convert low-level features into compressed features. Similarly, Vayghan et al. [3] used aerial images and LiDAR data to extract building and tree footprints in urban areas while Zhang et al. [47] fused the LiDAR data and a point cloud calculated from the aerial images to improve the accuracy of a ML building extraction algorithm. Shi et al. [48] suggested the use of an enhanced lightweight deep neural network with knowledge refinement to detect local features from LiDAR data and imagery while preserving solid robustness for day-night visual localization.

2.3. Multispectral LiDAR Data

Multispectral images have layers that represent the reflectance in a few wide and disconnected spectral bands within given specified spectral intervals [49]. In the case of airborne LiDAR data, some authors have used multispectral images in addition to the LiDAR point cloud as input data for ML algorithms, because most objects on the Earth's surface have indicative absorption features in certain discrete spectral bands which can help to create an accurate classification of the scanned scene [49]. Though the multispectral data are not always available, where they are, they can be an asset for processing efficacy. In this context, Marrs and Ni-Meister, [50] used LiDAR, hyperspectral, and thermal images on experimental forests and found that the combination of these two data can help improve the classification of tree species. Yu et al. [51] used multispectral LiDAR data for individual tree extraction and tree species recognition. Zhao et al. [52] used a FR-GCNet network to increase the classification accuracy of multispectral LiDAR point clouds, whereas Zhou et al. [53] applied an RF algorithm on a combination of hyperspectral images and LiDAR data for monitoring insects. Peng et al. [54] suggested a MultiView Hierarchical Network (MVHN) could be used to segment hyper spectral images and LiDAR point cloud together. For this purpose, the hyper spectral images were divided into multiple groups with the same number of bands to extract spectral features. Thereafter, ResNet framework was implanted to detect the spectral-spatial information of the merged features.

2.4. Full-Waveform Representation and Point Cloud

Some airborne laser systems, called full waveform, can record the complete power spectrum of the returned pulse. The different surface characteristics can influence the reflected signal, so analysis of the laser pulse full waveform has been used to improve the extraction of surface features [55] especially in forested areas. Five parameters are calculated from the waveform of the return pulse, e.g., the amplitude of the highest peak, the total energy, the full-width half-maximum return width, and the length of the sequence. In this context, Guan et al. [56] constructed a geometric tree model based on the full-waveform representation. Afterward, in order to classify the tree species, they applied a deep learning algorithm to the last model to extract the high-level features. Blomley et al. [57] classified

the tree species using the RF algorithm based on the geometric features calculated from the full-waveform analysis.

Similarly, by means of an integrated system that acquired hyperspectral images, LiDAR waveforms, and point clouds, Yang et al. [58] classified tree species after systematic pixel-wised investigation of different features. For this purpose, the Canopy Height Model (CHM) was extracted from the LiDAR data, and multiple features from the hyperspectral images, including Gabor textural features. Shinohara et al. [59] suggested a semantic classification algorithm named Full-Waveform Network (FWNet) based on PointNet-based architecture [27], which extracted the local and global features of the input waveform data. The classifier in this case consisted of 1D convolutional operational layers. Due to the sensitivity of border points to the multi return difference value, to achieve the cloud segmentation, Shin et al. [60] used multiple returns in addition to the point cloud as training data using the PointNet++ network [61].

2.5. Different Other Data

Sometimes other data, not mentioned previously, may be used in addition to the LiDAR point cloud. For example, Zhang et al. [62] used the interaction of the high-resolution L-band repeat-pass Polarimetric Synthetic Aperture Radar Interferometry (PolInSAR) and low-resolution large-footprint full-waveform LiDAR data to estimate forest height. Park and Guldmann [63] utilized a city LiDAR point cloud in addition to building footprint data to extract building class before applying an RF algorithm and Feng and Guo [64] suggested a segment-based parameter learning approach that fuses a 2D land map and 3D point cloud together.

For detecting individual trees, Schmohl et al. [65] used an orthophoto to colorize the point cloud for additional spectral features along with laser intensity and the number of returns utilized as additional input. Kogut et al. [66] improved the classification accuracy of seabed laser scanning (bathymetry data) by using the Synthetic Minority Oversampling Technique (SMOTE) algorithm to evaluate the input data. Then, a Multi-Layer Perceptron (MLP) neural workflow was applied to classify the point cloud. Barbarella et al. [67] applied a ML network that trained a model able to classify a particular gravity-driven coastal hillslope geomorphic model (slope-over-wall) including most of the soft rocks. However, they used only geometric data which are morphometric feature maps computed from a Digital Terrain Model (DTM) calculated from the LiDAR point cloud.

Finally, Duran et al. [68] compared nine ML methods: logistic regression, linear discriminant analysis, K-NN, decision tree, Gaussian Naïve Bayes, MLP, adaboost, RF, and SVM to classify LiDAR and colored photogrammetric point clouds into four classes: buildings, ground, low and high vegetation with the highest accuracy being attained with MPL. For more details about these ML techniques, please see Mohammed et al. [69] and Kim, [70].

3. Concepts of Point Cloud Structure for Applying ML Algorithms

The 3D point cloud consists of a large number of 3D points covering the scanned area. These points are normally distributed in an irregular way depending on the scanning system quality and the scanned area geometric characteristics. In any event, to process, classify, and model the LiDAR data using ML techniques, most of the suggested approaches try to define a mathematical model that allows for the management, reduction, pooling, and convolution of these data [71]. Consequently, most ML approaches consist of two main steps, firstly preprocessing and then ML algorithm application. In this paper, the mathematical model in addition to all operations realized on it before applying the ML technique is named the data adaptation step (Figure 2). The data adaptation procedures may play several roles. Some ML informatics tools for imagery data processing or other data kinds, require the transformation of point cloud into novel data forms such as 2D and 3D matrices before they can be used. As informatics tools for processing LiDAR data require high time processing cost, two solutions are employed: either designing new ML tools that correspond to the LiDAR data concept or, more commonly, reducing the LiDAR

data. At this stage, it is important to refer that the interpolation or reduction of LiDAR data is not always a preferable solution from the geomatics industrial viewpoint.

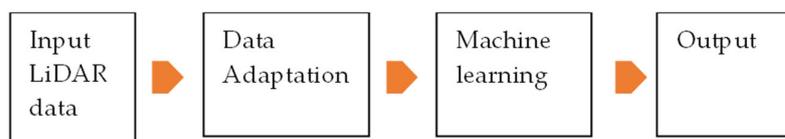


Figure 2. Structure of ML algorithm of LiDAR data processing.

In the next subsections, the main concepts of LiDAR data adaptation will be revealed and discussed.

3.1. Voxelization

Voxelization, a 3D matrixial representation, may sometimes solve the issue of the irregular distribution of the 3D point cloud [56]. In practice, the LiDAR points are distributed on the scanned surfaces which leads to a considerable number of empty voxels which cause additional calculation costs. Moreover, coarse spatial resolution (large voxel size) may cause the loss of information which will reduce the accuracy of data processing. Conversely, if the spatial resolution is too small, that may increase the calculation cost, and the memory usage [17].

In the literature, many authors suggest voxelizing the LiDAR point clouds. In this context, Maturana and Scherer [72] developed the VoxNet network using the occupancy grid algorithm. They divided the point cloud into many 3D grids and then normalized each grid unit to enter the volume build layers and maximum pooling layers. Gargoum et al. [73] suggested a voxel-based approach to classify the light poles of roads while Zou et al. [26] proposed a voxel-based deep learning method to identify tree species in a three-dimensional map. They extracted individual trees through point cloud density and used voxel rasterization to obtain features. Guan et al. [56] used a voxel-based upward growth algorithm to remove the ground point cloud and then segment a single tree species by European clustering and a voxel-based normalization algorithm. Shuang et al. [74] developed an Adaptive Feature Enhanced Convolutional Neural Network (AFERCNN) for 3D object detection. This algorithm is a point-voxel integrated network, where voxel features are extracted through the 3D voxel convolutional neural network. These features are projected to the 2D bird's eye view and the relationship between the features in both spatial dimension and channel dimension is learned. Wijaya et al. [75] applied a voxel-based 3D object detection deep neural network on terrestrial LiDAR data where they minimized the features from a 3D into a 2D bird-eye view map before generating object proposals to save processing time.

However, voxelization tries to conserve the LiDAR point cloud 3D structure by defining a spatial matrixial form that enables improved management of the point cloud. Hence, the form will be limited by the available, the used memory, and the requested processing time may represent the main limitations.

3.2. Graphic Structure

Using graphic structure to transform the 3D point cloud into a 2D regular grid has the main advantage of transforming the point cloud classification question into the general image processing one. Simonovsky and Komodakis [76] used edge labels to calculate Edge Conditional Convolution (ECC) in the neighborhood of regular grids. Then, an asymmetric edge operation was used to calculate the relationship between neighboring points. Wang et al. [77] developed a SpecGCN network where the maximum pooling was replaced with a recursive clustering. The nearest neighbor was applied to calculate a graph regular grid. Thereafter, they combined a spectral graph convolution using a local graph, with a pooling strategy. Nahhas et al. [46] suggested a deep learning approach based on using an interpolated LiDAR point cloud and orthophotos simultaneously. This approach employed object-based analysis to create objects, a feature-level fusion. Li et al. [78] devel-

oped a deep learning network named Attentive Graph Geometric Moments Convolution (AGGM Convolution) network to classify the LiDAR point cloud into four classes: trees, grass, roads, and buildings. The Dynamic Graph Convolution Neural Network (DGCNN), suggested by Wang et al. [28], built the directed graph in both the Euclidean space and the feature space, and dynamically updated the feature layers. A similar approach suggested by Wang et al. [29] employed the attention mechanism in the graph-based methods. The extended approach is named Graph Attention Convolution Network (GACNet) for semantic point cloud segmentation.

In the same context, Wen et al. [79] presented a global-local Graph Attention Convolution Neural Network (GACNN) that could be directly applied to airborne LiDAR data. The graph attention convolution module includes two types of attention mechanisms: a local attention module that combines edge attention and density attention, and a global attention module. The local edge attention module is designed to dynamically learn convolution weights using the spatial relationships of neighboring points; thus, the receptive field of the convolution kernel can dynamically adjust to the structure of the point cloud. Zhao et al. [52] used a Feature Reasoning-based Graph Convolution Network (FR-GCNet) to increase the classification accuracy of airborne multispectral LiDAR data. Jing et al. [80] proposed a Graph-based neural Network with an Attention pooling strategy (AGNet) where the local features were extracted through the point topological structure. Chen et al. [81] improved the descriptiveness in the network ChebyNet [82] by increasing the width of input to avert the above drawbacks. The suggested network, named WGNet, is inspired by the image processing dilated convolution. This network is based on two modules, the local dilated connecting and context information awareness. Wan et al. [83] developed a Dilated Graph Attention-based Network (DGANet) for local feature extraction on 3D point clouds. It was based on the dilated graph attention modules which allow the network to learn the neighborhood representation by using the long-range dependencies given by the calculated dilated graph-like region for each point.

To conclude, the use of graphic structure facilitates the point cloud processing duty tasks by using image processing functions, but unfortunately at the cost of minimizing the 3D structure advantages.

3.3. Kernel-Based Convolution

The geometric structure of a point cloud can be defined through the Kernel correlation layer [41]. The kernel size value can be suggested according to a different number of neighboring points in the convolution layer. Points within the kernel can contribute to their center point [84]. At this stage, Klokov et al. [85] proposed a K-NN algorithm that uses the Euclidean metric to return the closest points inside the kernel. The kernel is defined by two parameters: the inner and the outer radius to ensure that the closest and unique points will be detected in each ring kernel. In the context of ML applications, Song et al. [86] employed the kernel correlation learning block approach to recognize the local and global features at different layers thus enhancing the network perception capacity. Zhang et al. [31] suggested a Local k-NNs Pattern in Omni-Direction Graph Convolution Neural Network named LKPO-GNN to capture both the global and local point cloud spatial layout. This approach converts the point cloud into an ordered 1D sequence, to feed the input data into a neural network and reduce the processing cost.

In fact, this approach allows applying all operations directly on the point cloud, but it still requires an optimized neighborhood searching procedure.

3.4. Reducing of Point Cloud Density (Downsampling)

Most ML approaches applied to LiDAR data try to reduce data density and keep the processing time within accepted limits. The successful use of the convolutional technique within the image processing field has encouraged authors to use the same approach in reducing LiDAR data and thus to solve the processing time issue. Although the most used point cloud structures apply the idea of point cloud reduction, the suggested approaches in

this subsection conserve the point cloud structure and reduce the point density. However, the application of ML techniques is still in its infancy, and a lot of advancement is expected in future research.

In the context of point cloud reduction, Wen et al. [41] developed a D-FCN network architecture that included both downsampling and upsampling paths to enable multiscale point feature learning. Several authors, Hu et al. [30], Wei et al. [17], and Du et al. [22] used random downsampling to reduce the point cloud in the context of applying the ML algorithm such as developing consecutively feed-forward MLPsRandLA-Net and encoder–decoder structure (BushNet and ResDLPS-Net). Mao et al. [20] suggested three downsampling layers to classify the LiDAR data.

Though the downsampling reduces the data volume, it loses an important information quantity that may be useful to object recognition and modeling.

4. Employed ML Techniques

Currently, the advancement of digital technologies and data acquisition techniques in different disciplines can lead to the generation of excessively large data sets. To manage and process the oversized data sets, the questions of data classification and object recognition have become ones of crucial importance. In this context, ML techniques occupy an enviable position because they allow for automatic and efficient solutions. The ML techniques can be classified into four categories according to the required input data (see Mohammed et al. [69]): supervised learning, where labelled data are needed for training, unsupervised learning, where labelled data are not needed, semi-supervised learning that uses a mixture of classified and unclassified data, and reinforcement learning where no data are available. Of these, the supervised and unsupervised techniques may be considered the main two categories. In each one of these two groups, several algorithms are employed, e.g., supervised ML uses algorithms such as decision trees, rule-based classifiers, Naïve Bayesian classification, k-nearest neighbors' classifiers, RF, Neural Networks (NN), linear discriminant analysis, and SVM, whereas unsupervised ML uses k-means clustering, Gaussian mixture model, hidden Markov model, and principal component analysis.

In the LiDAR data-processing domain, the application of ML algorithms represents an emerging research area. Despite the great number of papers published in this area, very few new ML algorithms are employed. In the next subsections, more focused ML algorithms will be introduced and discussed.

4.1. Random Forest (RF) and Support Vector Machine (SVM)

Tarsha Kurdi et al. [87] summarized the applications of RF classifiers for automatic vegetation detection and modelling using LiDAR point clouds. Many authors used RF exclusively on LiDAR data [88], whereas other authors used additional data [89,90]. Yu et al. [91], and Yu et al. [51] estimated tree characteristics such as diameter, height, and stem volume using an RF classifier and Levick et al. [92] connected the DSM and field-measured wood volume using an RF algorithm. Chen et al. [88] used the feature selection method and an RF algorithm for landslide detection under forest canopy, where the DTM and the slope model were constructed for the scanned area, and the features were calculated at the pixel level. The same principle was used by Guan et al. [93] to identify the city classes in urban areas and Ba et al. [94] employed RF for detecting the tree species.

Man et al. [90] applied an RF classifier to calculate a two-dimensional distribution map of urban vegetation. In this study, individual tree segmentation was conducted on a CHM and point cloud data separately to obtain three-dimensional characteristics of urban trees. The results show that both the RF classification and object-based classification could extract urban vegetation accurately, with accuracies above 99%, and the individual tree segmentation based on point cloud data could delineate individual trees in three-dimensional space better than CHM segmentation. Arumäe et al. [95] calculated a model for predicting necessity thinning using the RF technique to retrieve the two indicative parameters for requiring thinning, height percentage and the canopy cover. Park and

Guldmann, [63] used an RF algorithm to classify building point clouds into four classes: rooftop, wall, ground, and high outlier. To overcome the complexity of building geometry of the Ming and Qing Dynasties' Official Architecture style (MQDOAs), Dong et al. [96] employed semantic roof segmentation. This method was composed of two stages. Some geometric features such as the normalized symmetrical distance, relative height, and local height difference are extracted and then the RF algorithm is applied to classify the roof point cloud. Feng and Guo [64] suggested a segment-based parameter learning approach in which a 2D land cover map is chosen to generate labelled samples, and a formalized operation is then implemented to train the RF classifier. Liao et al. [97] fed in point cloud super voxels and their convex connected patches into an RF algorithm. For this purpose, they consider three types of features: point-based, eigen-based, and grid-based.

The SVM algorithm tries to find a hyperplane in high dimensional feature space to classify some linearly correlative point distributions. While there could be many hyperplanes that separate the target classes, the hyperplane that optimizes the boundary between the classes is identified. Aside from just linear classification, SVM can carry out nonlinear classification using the kernel trick by indirectly drawing their inputs into high-dimensional feature spaces [69].

Though the SVM classifier is efficient for data classification when using rather small data, it is also used by Ba et al. [94] to recognize tree species. Murray et al. [43] trained an SVM on the passing and ongoing results of a CNN algorithm through pixel classification and the interpolation result of the intensity vector as input data. Hoang et al. [98] introduced a hybrid approach of a CNN and an SVM for 3D shape recognition, where eight layers of the CNN are utilized for geometric feature extraction and afterward an SVM is applied to classify them. Zhang et al. [99] suggested an object-based approach to classify an urban airborne LiDAR point cloud. First, different point features such as geometry, radiometry, topology, and echo characteristics are extracted and then the SVM classifier algorithm was applied to detect five classes: terrain, vegetation, building, powerlines, and vehicles. To detect powerlines, Shokri et al. [100] eliminated the undesirable points and then apply the SVM after calculating the point geometric features.

In conclusion, RF and SVM are less used in recent years, and both are more basic classification models. Therefore, most modern approaches focus on deep learning techniques.

4.2. Neural Network and Deep Learning

Deep learning represents a sort of ML, and it can be defined as a ML technique that employs a deep neural network such as the MLP neural network that contains two or more hidden layers [70]. A Perceptron Neural network consists of single neurons that have multiple inputs and generate a single output using an activation function. Figure 3 illustrates a deep learning algorithm functionality where the available data consist of two sections: labelled and unlabeled data. The labelled data will be used in training the suggested MLP neural network to correct the assumed weight values which will then be used in the same neural network to label the unlabeled data. For more information about deep learning techniques, please see Kim [70].

In the LiDAR data processing area, deep learning algorithms are widely applied especially for data classification. Zou et al. [26] used a low-level feature representation through voxel-based structure, and then classified tree species by using a deep learning model. In regard to Generative Adversarial Networks (GAN), Goodfellow et al. [101] have achieved a notable performance on pan-sharpening in the image processing domain. Zhang et al. [62] developed a PolGAN deep learning network to determine the forest tree heights. When applying a deep learning classification algorithm, Lin et al. [19] improved the labelling stage to produce training data because the data labelling procedure for generating training data consumes considerable time and effort. In this context, they suggested using weak labelling that needs little annotation effort. The pseudo labels are then considered as the input of a classification network [102]. Thereafter, an overlap region loss and an

elevation attention unit are introduced for the classification network to obtain more accurate pseudo labels.

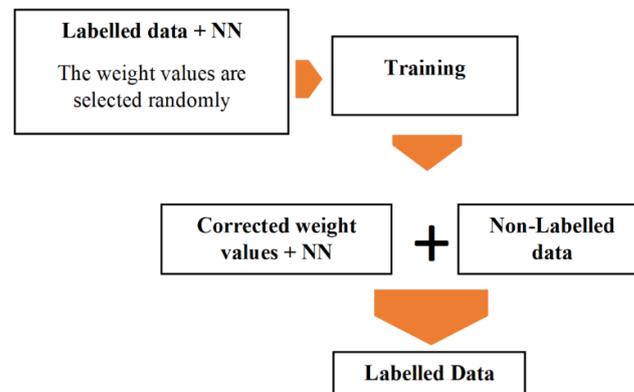


Figure 3. Deep learning functionality; NN is a Neural network.

Zhao et al. [52] used a Feature Reasoning-based Graph Convolution Network (FR-GCNet) to increase the classification accuracy of urban point clouds. Semantic labels were assigned to pixels using global and local features. Based on the graph convolution network, a global reasoning unit is embedded to find the global contextual features, while a local reasoning unit is added to learn edge features with attention weights in each local graph. Li et al. [103] compared three deep learning algorithms for classifying LiDAR point clouds, these algorithms are PointNet++ [61], SparseCNN [104] and KPConv [105]. They found that SparseCNN carries out a better classification accuracy than the other two approaches.

Where there are variations of point cloud density, Théodose et al. [106] suggested adapting an object detection deep learning approach. For this purpose, some data layers are randomly dismissed during the training step to grow the variability of the processed data. Sheikh et al. [32] proposed a Deep Feature Transformation Network (DFT-Net) to classify terrestrial LiDAR data. The suggested algorithm is based on graph analysis in which the edges are dynamically extracted for each layer. Hoang et al. [107] extracted and associated both global and regional features through Gaussian SuperVector and enhancing region illustration deep learning Network (GSV-NET) for 3D point cloud classification. Chen et al. [108] developed a Dynamic Point Feature Aggregation deep learning Network (DPFA-Net) by selectively performing the neighborhood feature aggregation, dynamic pooling, and an attention mechanism. In this semantic classification of the LiDAR point cloud framework, the features of the dynamic point neighborhood are aggregated via a self-attention mechanism. Finally, Song et al. [109] developed, in the context of automatic LiDAR data classification, a 2D and 3D Hough Network (2D&3DHNet) by linking 3D global Hough features and 2D local Hough features with a classification deep learning network.

4.3. Encoder–Decoder Structure

In the encoder–decoder structure, the network consists mainly of two subnetworks: the encoder sub-network and the decoder sub-network [110]. In the encoder part, consecutive downsampling procedures increase the receptivity of the extracted features but unfortunately, that reduces the point cloud resolution. In the decoder part, upsampling and convolution operations are employed for resolution recapture and feature combination.

In laser scanning, several authors developed an encoder–decoder algorithms to classify LiDAR data. Wen et al. [79] created an end-to-end encoder–decoder network named GACNN that is based on the graph attention convolution module and used it for detecting multiscale features of the LiDAR data and achieving point cloud classification. Wei et al. [17] proposed a network point cloud segmentation named BushNet which is the classic encoder–decoder structure. In this context, a minimum probability random sampling module is used for reducing the processing time and improving the convergence speed. Thereafter, the local multi-dimensional feature fusion module is applied to make the

network more sensitive to bush point cloud features. Thus, the employed multi-channel attention module may improve the training efficiency.

Medina and Paffenroth [111] applied an encoder–decoder classifier for reduced LiDAR data by feeding the network with features calculated from the point neighborhood, which showed high efficiency in distinguishing the non-linear features. Mao et al. [20] developed an encoder–decoder architecture for point cloud classification named Receptive Field Fusion and Stratification Network (RFFS-Net) that is based on the PointConv network suggested by Wu et al. [112]. It consists of two steps: hierarchical graph generation and encoder–decoders feature extraction and aggregation. The input is provided by a hierarchical graph generation model and point features after which the point features are aggregated. Ibrahim et al. [113] used CNN architectures to semantically classify the terrestrial LiDAR data. They divided the point cloud into angle-wise slices that are transformed in the next step into enhanced pseudo images using the intensity and reflectivity values. Then, these images are employed to feed an encoder–decoder CNN model.

Finally, despite the promising results obtained by deep learning as well as encoder–decoder structure, more focus is needed on unsupervised learning techniques which may cancel the request for training data.

Having presented the main ML algorithms used to process LiDAR data, the next section will discuss current applications of ML technique on LiDAR point cloud.

5. Applications of ML on LiDAR Data

The use of laser scanning technology is widespread. It has been applied in urban, rural, and forested areas to target natural as well as artificial objects such as buildings (inside and outside), roads, railways, bridges, tunnels, and pipelines. Almost inevitably, a point cloud of a scanned area will consist of several object classes such as terrain, vegetation, buildings, standing water, noise, and artificial objects. As each class has a different modelling concept, it is essential to classify the point cloud into its main classes before starting the modelling step [5]. Once the point cloud of the scanned area is classified, the obtained classes can be analyzed and modelled according to the project goal. In this context, a large list of class modelling operations could be described. From the creation of laser scanning technology, most of the suggested approaches in the literature have been rule-based. Within the last five years, ML techniques have become an important approach for LiDAR data processing [2,8]. Unfortunately, ML techniques have hitherto only been used to a limited number of procedures, e.g., according to Hamedianfar et al. [114], the main applications of deep learning algorithms in forest areas are biomass estimation and tree species classification. In the next subsections, the main applications of ML techniques on LiDAR data are detailed.

5.1. Building Detection

The ML classifiers are sometimes focused on the building class in urban areas, with the aim of classifying the scanned scene into two classes: buildings and non-buildings. Nahhas et al. [46] suggested a deep learning approach based on the feature-level. CNN was used to transform compressed features into high-level features, which were used in building detection. Zhang et al. [47] used the U-NET model [115] to detect building polygons from orthophotos. Hence, to increase the point cloud density, the LiDAR, and photogrammetric point clouds are merged and employed for each polygon for feature extraction goals. Ojogbane et al. [116] improved a deep learning network suggested by Seydi et al. [117] to detect the building class. The suggested framework fuses the features obtained from interpolated airborne LiDAR data into DSM, in addition to a very high-resolution aerial imagery. Shin et al. [60] applied PointNet++ [61] for building extraction using multiple returns data.

While ML algorithms are employed by several authors for building recognition, in fact, the urban scene cannot just be simplified into building and non-building classes. Hence, the next section will go further through applying ML to achieve full classification.

5.2. Scene Segmentation

The classification question is widely discussed in this research area. One scanned scene consists of several classes, and the question that arises is: can the classification algorithm be used to extract the desired class list? Or can one algorithm only recognize certain classes? For this reason, we have chosen to identify the classification algorithms according to detected classes. With respect to airborne data, not all authors agree about the ideal number of classes. Wen et al. [41] developed a deep learning network that classified the airborne LiDAR data into nine classes: powerlines, low vegetation, cars, fences, roofs, facades, shrubs, and trees. Despite Wang and Gu [118] using the same number of classes, their class list is different: earth bar, grass, roads, buildings, trees, water, powerlines, cars, and ships. Li et al. [78] suggested a deep learning pixel-based analysis network to distinguish four classes in airborne data: trees, grass, roads, and buildings. Another class list is suggested by Ekhtari et al. [119] classified their scene into six classes: buildings, soil, grass, trees, asphalt, and concrete. An example of the final data set is shown in Figure 4. Zhao et al. [52] made small modifications to these classes as follows: roads, buildings, grass, trees, powerlines, and soil. Another modification to these classes is suggested by Shinohara et al. [59]: roads, buildings, transmission towers, trees, powerlines, and ground. Liao et al. [97] classified the airborne point cloud into three main classes: terrain, buildings, and vegetation using the RF algorithm. Zhao et al. [120] suggested a Point Expanded Multi-Scale Convolutional Network (PEMCNet) to classify the airborne LiDAR data containing point cloud, intensity, and return number, into five classes: ground, high vegetation, building, water, and raised road. To calculate the point features, it created point expanded grouping units that combined the extracted features at diverse scales. It is fair to say that the classes chosen in each study are a product of the study area and study aim rather than a desire to develop a universal class set.

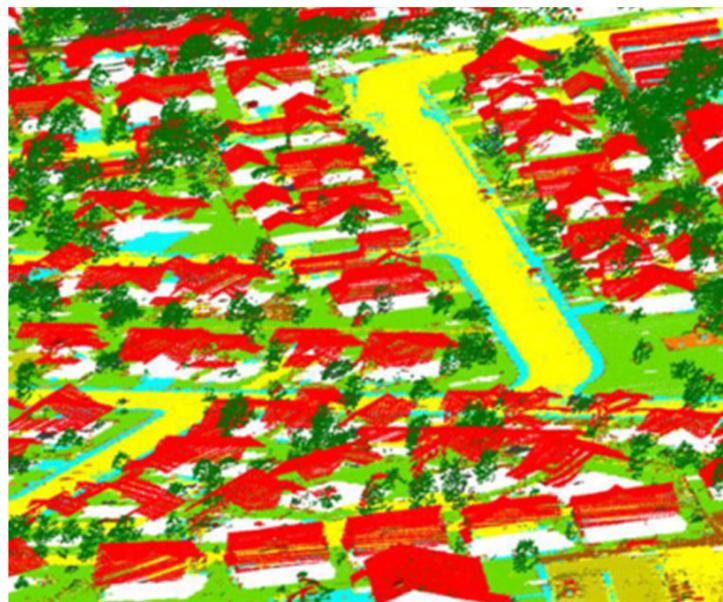


Figure 4. An example of a 3D point cloud classified into six classes (buildings, soil, grass, trees, asphalt, and concrete) by Ekhtari et al. [119].

In the case of terrestrial data, a huge diversity of suggested class lists reflects the diversity of scanned scenes. Wang et al. [28], Qi et al. [27], Wang et al. [29], Hu et al. [30], Wei et al. [17], Zhang et al. [31], Xiu et al. [23], and Jing et al. [80] classified the terrestrial LiDAR data into several classes according to the scanned objects. To classify the terrestrial LiDAR data, Wen et al. [121] converted the LiDAR point cloud into a pseudo image and applied a semantic segmentation algorithm named Hybrid CNN-LSTM that has a neural network framework. Hence, the pseudo image is considered within Long Short-Term

Memory (LSTM) network that combines the different channel features generated by a convolutional neural network. Shuang et al. [122] proposed for terrestrial LiDAR point cloud classification, a Multi-Spatial Information and Dual Adaptive (MSIDA) network, which consists of encoding and dual adaptive sub-networks. To encode the point coordinates, each point and its neighborhood are transferred into a cylindrical and spherical coordinate system. The DA sub-network comprises a Coordinate System Attention Pooling Fusion (CSAPF) block in addition to a Local Aggregated Feature Attention (LAFA) one.

5.3. Vegetation Detection

Some classification algorithms are developed especially for forest areas, that focus on the vegetation class. In this case, they classify the scanned scene into two classes: vegetation and non-vegetation. Luo et al. [24] developed a semantic segmentation deep network to extract vegetation points from the LiDAR point cloud, where the tree points are grouped into a set of tree clusters using Euclidean distance clustering. A Pointwise Direction Embedding deep network (PDE-net) is employed to calculate the direction vectors of tree centers. Chen et al. [123] compared four ML algorithms: RF, Cubist, XGBoost, and CatBoost with rule-based algorithms to improve the estimation performance of forest biomass. The ML algorithms outperformed parametric stepwise regression, with the CatBoost network being superior, followed by XGBoost, RF, Cubist, and stepwise regression.

In the context of individual tree detection, Schmohl et al. [65] exploited the 3D LiDAR point cloud by using a 3D NN to detect individual trees. A sparse convolutional network was applied for feature calculation and feeding of the semantic segmentation output. Furthermore, they defined five semantic classes obtained from the dataset: terrain, buildings, low points, bridges, and vegetation. Luo et al. [124] proposed a tree detection algorithm through a deep learning framework based on a multi-channel information complementarity illustration. An adapted graph convolution network with local topological information was developed to extract the ground class thus avoiding parameters selection that did not consider different ground topographies. Then, a multichannel representation in addition to Multi-Branch Network (MBNet) was used through fusing multi-channel features. Corte et al. [125] used uncrewed aerial vehicle LiDAR point cloud to test four different ML approaches to detect individual trees and estimate their metrics such as diameter at breast height, total height, and timber volume. The tested methods were SVM, RF, NN, and Extreme Gradient Boosting. Windrim and Bryson [126] isolated individual trees, determine stem points, and further built a segmented model of the main tree stem that encompasses tree height, and diameter. This approach used deep learning models passing through multiple stages starting by ground characterization and removal, delineation of individual trees, and segmentation of tree points into stem and foliage. An example of output of their algorithm is shown in Figure 5. For extracting grasses and individual trees, Man et al. [90] extracted the two-dimensional distribution map of urban vegetation using the object-based RF classification method. Chen et al. [127] employed a PointNet network [27] for segmenting the individual tree crowns using the voxelization strategy.

Finally, Vayghan et al. [3] extracted high-elevation objects from the LiDAR data using the developed scan labelling method, and then the classification methods of a NN. Adaptive Neuro-Fuzzy Inference System (ANFIS), and Genetic Based K-Means algorithm (GBKMs) were used to separate buildings and trees with the purpose of evaluating their performance.

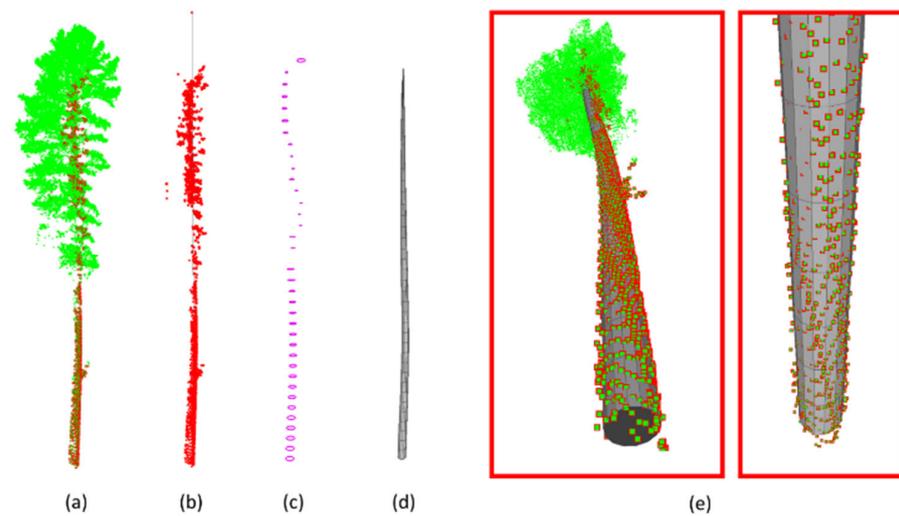


Figure 5. An example of the output of Windrim and Bryson's [126] (Figure 8) deep learning model where (a) is the segmented point cloud, (b) isolated stem points, (c) RANdom SAmple Consensus (RANSAC) algorithm circles attach stem section and (d) refined stem sections estimate based on robust least-squares fitting process. Panel (e) shows examples of the final fitted stem model.

5.4. Classification of Tree Species

Zou et al. [26] suggested a voxel-based deep learning method to classify terrestrial LiDAR point clouds of a forested area into species. They used three consecutive steps. After the extraction of individual trees using the density of the point clouds, a low-level feature voxel-based representation was constructed and then the classification of tree species was achieved by using a deep learning model.

Marrs and Ni-Meister [50] compared NNs, k-nearest neighbors, and RF approaches for recognizing tree species. The used variable reduction techniques and showed mixed results depending on the exact set of inputs to each machine learner. Dimensionality reduction based on classification tree nodes is a technique worth trying on multisource datasets. Mizoguchi et al. [128] classified individual tree species using terrestrial LiDAR based on CNN. The key component was the initial step of a depth image creation which well described the characteristics of each species from a point cloud.

Ba et al. [93] employed SVM and RF algorithms to test the discrimination level between tree genera. In this context, tree crowns were isolated and global morphology and internal structure features were computed. Yu et al. [51], Budei et al. [129], and Blomley et al. [57] estimated tree species based on an RF using tree features as predictors and tree species as a response for correctly extracted trees. Figure 6 shows an example of a successful detection phase. Yang et al. [58], and Nguyen et al. [130] both identified the tree species from LiDAR data in addition to other airborne measurements such as hyperspectral images using an SVM classifier. Hell et al. [131] tested the capacity of two deep learning networks PointCNN [132], and 3DmFV-Net [133] for the classification of four different tree species, both living and dead, using LiDAR data. It was shown that 3DmFV-Net is adequate for the geometry of the single trees, whereas PointCNN permits the incorporation of other features.

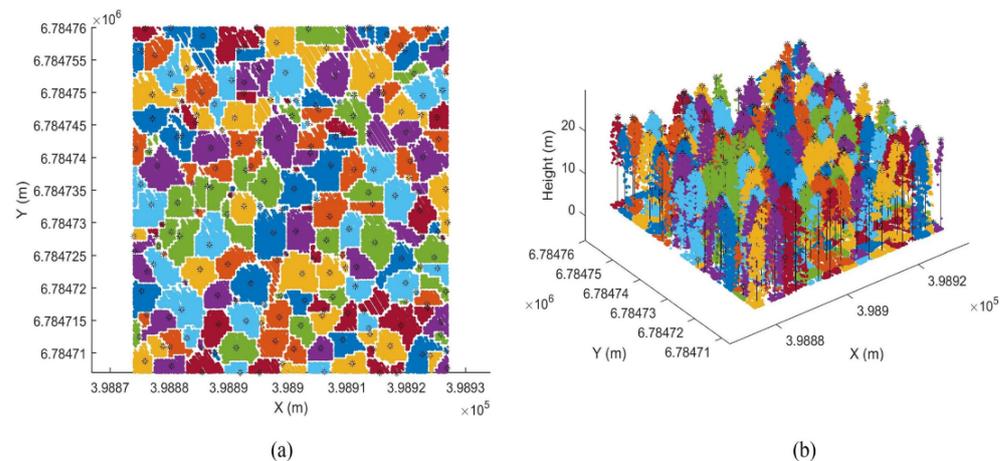


Figure 6. An example of the results of Yu et al. [51] tree detection stage from (a) a plan view and (b) a 3D view.

5.5. Road Marking Classification

The high retro-reflective materials of road markings cause a high laser intensity with respect to the surrounding areas. On one hand, this feature allows easy identification of the road markings but unfortunately, the road markings are not only incomplete but also contain discontinues. That is why the road marking classification represents a challenging task [42]. In this context, Wen et al. [134] used a modified U-net model to segment road marking pixels to overcome the intensity variation, low contrast, and other obstacles. (Ma et al. [135] developed a capsule-based deep learning framework for road marking extraction and classification that consists of three modules. This approach starts with the segmentation of road surfaces. Thereafter, an Inverse Distance Weighting (IDW) interpolation is applied. Based on the convolutional and deconvolutional capsule operations, a U-shaped capsule-based network is created, and a hybrid network is developed using a revised dynamic routing algorithm and Softmax loss function. Fang et al. [42] proposed a graph attention network named GAT_SCNet to simultaneously group the road markings into 11 categories from LiDAR point clouds. The GAT_SCNet model builds serial computable subgraphs and uses a multi-head attention mechanism to encode the geometric and topological links between the node and neighbors to calculate the different descriptors of road marking.

5.6. Other Applications

In addition to the main applications presented previously, several important attempts to employ the ML for achieving other automatic operations on LiDAR data are documented in the literature. Ma et al. [136] proposed a workflow for the automatic extraction of road footprints from urban airborne LiDAR point clouds using deep learning PointNet++ [61]. In addition to the point cloud and laser intensity, the co-registered images and generated geometric features are employed to describe a strip-like road. In this context, graph-cut and constrained Triangulation Irregular Networks (TIN) are considered. Shajahan et al. [137] suggested a view-based method called a MultiView Convolutional Neural Network with Self-Attention (MVCNN-SA), which recognizes the roof geometric forms by considering multiple roof point cloud views.

In self-driving cars, several applications such as object recognition, automatic classification, and feature extraction are carried out using ML techniques [34,36,138–142]. The importance of data filtering before starting the modelling operation has been established and Gao et al. [143] proposed a filtering algorithm that uses deep learning and a multi-position sensor comparison approach to eliminate reflection noise. Nurunnabi et al. [144] introduced a local feature-based non-end-to-end deep learning approach that generated a binary classifier for terrain class filtering from which the feature relevance in addition to

the models of different feature combinations were analyzed. Cao and Scaioni [145] applied a deep learning algorithm for semantic segmentation of terrestrial building point clouds. To reduce the number of labels, they suggested a label-efficient deep learning network (3DLEB-Net) that obtained per-point semantic labels of building point clouds with limited supervision. Shokri et al. [100] proposed an SVM approach for automated detection of powerlines from a LiDAR point cloud. In roadside laser scanning system applications, Zhang et al. [146] addressed the goal of a joint detection and tracking scheme by applying PV-RCNN [147] to automatic vehicle and pedestrian detection from the measured moving point cloud. Yin et al. [148] established a squeeze-excite mechanism in local aggregation procedures and employed deep residual learning through a suggested deep learning network that classified complicated piping elements. Amakhchan et al. [149] applied an MLP to filter the LiDAR building point cloud by eliminating the non-roof points. Mammoliti et al. [150] applied the semi-supervised clustering which combined semi-supervised learning and cluster analysis, to evaluate the rock mass discontinuities, orientation and spacing.

6. Conclusions

This paper has summarized and reviewed the state-of-the-art ML approaches applied to topographical LiDAR data. Four aspects were considered to analyze the studied methods. First, although all suggested approaches use an airborne or terrestrial LiDAR point cloud of the scanned scene as input data, some of them use, sometimes simultaneously, additional data such as real images, multispectral images, and waveforms to improve their efficiency. Of course, *prima facie*, using supplementary data may improve the conditions for obtaining the target result, but it is worth considering the contribution of the additional data to the final result. How critical the additional data are to the success of the target task needs to be verified.

Second, in literature, a long list of supervised and unsupervised ML techniques is available. As the unsupervised methods do not need labelled data, the use of these methods can solve the training data labelling problem. Unfortunately, most of the suggested approaches focus only on three supervised ML techniques: NN, RF, and SVM. More research is necessary to investigate the possible application, on LiDAR data, of other ML techniques, especially the unsupervised variety. These may provide opportunities for more efficient and lower cost solutions.

The third aspect is the concept of the LiDAR point cloud structure used within ML algorithms. Many of the proposed algorithms try to transform the question of 3D LiDAR data processing into 2D imagery processing so as to exploit the availability of the image processing informatics tools. These transformations lead to loss of information partly because of dimension reduction. Furthermore, the data reduction through downsampling techniques is similar to the pooling operation employed in image processing algorithms. This procedure is undesirable because it leads to the loss of information which may be beneficial to classify the data successfully. In this context, more research is needed to design a new methodology that simultaneously conserves the LiDAR data and saves the processing time.

Fourth, in regard to the new tools or trends for large-scale mapping and 3D modelling, ML techniques can mainly be employed to achieve five operations on topographical LiDAR data which are: buildings class detection, data classification, point cloud segmentation into vegetation and non-vegetation classes, separation of different tree species, and road marking classification. Some other applications of ML appear rarely in the literature. In fact, most feature-detection operations from topographical LiDAR data can be carried out with the help of classification procedures such as the detection of lines, planes, vertices, surfaces, breaklines, and borders. Filtering operations and modelling also represent an investigation area to apply ML techniques. Clearly, more effort and investigation are needed to improve and to apply ML algorithms on topographical LiDAR data.

Author Contributions: Conceptualization: F.T.K., Z.G. and G.C.; Methodology: Z.G. and F.T.K.; Investigation: F.T.K.; Resources: F.T.K. and Z.G.; Writing—original draft preparation: F.T.K., Z.G. and G.C.; Writing—review and editing: F.T.K., Z.G. and G.C.; Visualization: F.T.K. and Z.G.; Supervision: F.T.K., Z.G. and G.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Figure 4 was adopted from Ekhtari et al. [119], Figure 5 was adopted from Windrim and Bryson’s [126], and Figure 6 was adopted from Yu et al. [51].

Acknowledgments: Thanks to Paul Reed Managing Director of East Coast Surveys (Aust) Pty Ltd. and CloudXPlus company to provide the dataset of Figure 1 which was measured in Queensland, Australia, <http://www.eastcoastsurveys.com.au> (accessed on 9 August 2022).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tarsha Kurdi, F.; Gharineiat, Z.; Campbell, G.; Dey, E.K.; Awrangjeb, M. Full series algorithm of automatic building extraction and modelling from LiDAR data. In Proceedings of the 2021 Digital Image Computing: Techniques and Applications (DICTA), Gold Coast, Australia, 29 November–1 December 2021; pp. 1–8. [CrossRef]
2. Shan, J.; Toth, C.K. *Topographic Laser Ranging and Scanning Principles and Processing*, 2nd ed.; Taylor & Francis Group, LLC: Boca Raton, FL, USA, 2018; 630p; ISBN 13: 978-1-4987-7227-3.
3. Vayghan, S.S.; Salmani, M.; Ghasemkhani, N.; Pradhan, B.; Alamri, A. Artificial intelligence techniques in extracting building and tree footprints using aerial imagery and LiDAR data. *Geocarto Int.* **2022**, *37*, 2967–2995. [CrossRef]
4. Ayazi, S.M.; SaadatSeresht, M. Comparison of traditional and machine learning base methods for ground point cloud labeling. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2019**, *42*, 141–145. [CrossRef]
5. Tarsha Kurdi, F.; Gharineiat, Z.; Campbell, G.; Awrangjeb, M.; Dey, E.K. Automatic filtering of LiDAR building point cloud in case of trees associated to building roof. *Remote Sens.* **2022**, *14*, 430. [CrossRef]
6. De Geyter, S.; Bassier, M.; Vergauwen, M. Automated training data creation for semantic segmentation of 3D point clouds. *Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2022**, *46*, 59–67. [CrossRef]
7. Michałowska, M.; Rapiński, J. A Review of tree species classification based on airborne LiDAR data and applied classifiers. *Remote Sens.* **2021**, *13*, 353. [CrossRef]
8. Shan, J.; Toth, C.K. *Topographic Laser Ranging and Scanning Principles and Processing*; Taylor & Francis Group, LLC: Boca Raton, FL, USA, 2008; 593p, ISBN 13: 978-1-4200-5142-1.
9. Dey, E.K.; Tarsha Kurdi, F.; Awrangjeb, M.; Stantic, B. Effective selection of variable point neighbourhood for feature point extraction from aerial building point cloud data. *Remote Sens.* **2021**, *13*, 1520. [CrossRef]
10. Ben-Shabat, Y.; Lindenbaum, M.; Fischer, A. Nesti-net: Normal estimation for unstructured 3D point clouds using convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16–20 June 2019; pp. 10112–10120. [CrossRef]
11. Weinmann, M.; Jutzi, B.; Hinz, S.; Mallet, C. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS J. Photogramm. Remote Sens.* **2015**, *105*, 286–304. [CrossRef]
12. Sanchez, J.; Denis, F.; Coeurjolly, D.; Dupont, F.; Trassoudaine, L.; Checchin, P. Robust normal vector estimation in 3D point clouds through iterative principal component analysis. *ISPRS J. Photogramm. Remote Sens.* **2020**, *163*, 18–35. [CrossRef]
13. Thomas, H.; Goulette, F.; Deschaut, J.; Marcotegui, B.; LeGall, Y. Semantic classification of 3D point clouds with multiscale spherical neighborhoods. In Proceedings of the International Conference on 3D Vision (3DV), Verona, Italy, 5–8 September 2018; pp. 390–398. [CrossRef]
14. Nurunnabi, A.; Teferle, F.N.; Laefer, D.F.; Lindenbergh, R.C.; Hunegnaw, A. A two-step feature extraction algorithm: Application to deep learning for point cloud classification. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **2022**, *46*, 401–408. [CrossRef]
15. Tarsha Kurdi, F.; Landes, T.; Grussenmeyer, P. Joint combination of point cloud and DSM for 3D building reconstruction using airborne laser scanner data. In Proceedings of the 4th IEEE GRSS/WG III/2+5, VIII/1, VII/4 Joint Workshop on Remote Sensing & Data Fusion over Urban Areas and 6th International Symposium on Remote Sensing of Urban Areas, Télécom Paris, Paris, France, 11–13 April 2007; p. 7.
16. Tarsha Kurdi, F.; Landes, T.; Grussenmeyer, P.; Koehl, M. Model-driven and data-driven approaches using Lidar data: Analysis and comparison. In *ISPRS Workshop, Photogrammetric Image Analysis (PIA07)*; Institut für Photogrammetrie und Fernerkundung (IPF): Munich, Germany, 2007; Part 3 W49A; Volume XXXVI, pp. 87–92; ISSN 1682-1750.
17. Wei, H.; Xu, E.; Zhang, J.; Meng, Y.; Wei, J.; Dong, Z.; Li, Z. BushNet: Effective semantic segmentation of bush in large-scale point clouds. *Comput. Electron. Agric.* **2022**, *193*, 106653. [CrossRef]
18. Winiwarter, L.; Esmorís Pena, A.M.; Weiser, H.; Anders, K.; Martínez Sánchez, J.; Searle, M.; Höfle, B. Virtual laser scanning with HELIOS++: A novel take on ray tracing-based simulation of topographic full-waveform 3D laser scanning. *Remote Sens. Environ.* **2021**, *269*, 112772. [CrossRef]

19. Lin, Y.; Vosselman, G.; Yang, M.Y. Weakly supervised semantic segmentation of airborne laser scanning point clouds. *SPRS J. Photogramm. Remote Sens.* **2022**, *187*, 79–100. [[CrossRef](#)]
20. Mao, Y.; Chen, K.; Diao, W.; Sun, X.; Lu, X.; Fu, K.; Weinmann, M. Beyond single receptive field: A receptive field fusion-and-stratification network for airborne laser scanning point cloud classification. *ISPRS J. Photogramm. Remote Sens.* **2022**, *188*, 45–61. [[CrossRef](#)]
21. Ao, Z.; Su, Y.; Li, W.; Guo, Q.; Zhang, J. One-Class Classification of Airborne LiDAR Data in Urban Areas Using a Presence and Background Learning Algorithm. *Remote Sens.* **2017**, *9*, 1001. [[CrossRef](#)]
22. Du, J.; Cai, G.; Wang, Z.; Huang, S.; Su, J.; Marcato Junior, J.; Smit, J.; Li, J. ResDLPS-Net: Joint residual-dense optimization for large-scale point cloud semantic segmentation. *ISPRS J. Photogramm. Remote Sens.* **2021**, *182*, 37–51. [[CrossRef](#)]
23. Xiu, H.; Liu, X.; Wang, W.; Kim, K.S.; Shinohara, T.; Chang, Q.; Matsuoka, M. Enhancing local feature learning for 3D point cloud processing using unary-pairwise attention. In Proceedings of the 32nd British Machine Vision Conference, Online, 22–25 November 2021.
24. Luo, H.; Khoshelham, K.; Chen, C.; He, H. Individual tree extraction from urban mobile laser scanning point clouds using deep pointwise direction embedding. *ISPRS J. Photogramm. Remote Sens.* **2021**, *175*, 326–339. [[CrossRef](#)]
25. He, D.; Abid, F.; Kim, Y.-M.; Kim, J.-H. SectorGSnet: Sector learning for efficient ground segmentation of outdoor LiDAR point clouds. *IEEE Access* **2022**, *10*, 11938–11946. [[CrossRef](#)]
26. Zou, X.; Cheng, M.; Wang, C.; Xia, Y.; Li, J. Tree classification in complex forest point clouds based on deep learning. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 2360–2364. [[CrossRef](#)]
27. Qi, C.R.; Su, H.; Mo, K.C.; Guibas, L.J. PointNet: Deep learning on point sets for 3D classification and segmentation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 77–85. [[CrossRef](#)]
28. Wang, Y.; Sun, Y.; Liu, Z.; Sarma, S.E.; Bronstein, M.M.; Solomon, J.M. Dynamic graph CNN for learning on point clouds. *ACM Trans. Graph.* **2019**, *38*, 1–12. [[CrossRef](#)]
29. Wang, L.; Huang, Y.C.; Hou, Y.L.; Zhang, S.M.; Shan, J. Graph attention convolution for point cloud semantic segmentation. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 10288–10297. [[CrossRef](#)]
30. Hu, Q.; Yang, B.; Xie, L.; Rosa, S.; Guo, Y.; Wang, Z.; Trigoni, N.; Markham, A. Randla-Net: Efficient semantic segmentation of large-scale point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 11108–11117.
31. Zhang, W.; Su, S.; Wang, B.; Hong, Q.; Sun, L. Local K-NNs pattern in omni-direction graph convolution neural network for 3D point clouds. *Neurocomputing* **2020**, *413*, 487–498. [[CrossRef](#)]
32. Sheikh, M.; Asghar, M.A.; Bibi, R.; Malik, M.N.; Shorfuzzaman, M.; Mehmood, R.M.; Kim, S.-H. DFT-Net: Deep feature transformation based network for object categorization and part segmentation in 3-dimensional point clouds. *Sensors* **2022**, *22*, 2512. [[CrossRef](#)]
33. Li, L.; Kong, X.; Zhao, X.; Huang, T.; Li, W.; Wen, F.; Zhang, H.; Liu, Y. RINet: Efficient 3D Lidar-based place recognition using rotation invariant neural network. *IEEE Robot. Autom. Lett.* **2022**, *7*, 4321–4328. [[CrossRef](#)]
34. Silva, J.; Pereira, P.; Machado, R.; Névoa, R.; Melo-Pinto, P.; Fernandes, D. Customizable FPGA-based hardware accelerator for standard convolution processes empowered with quantization applied to LiDAR data. *Sensors* **2022**, *22*, 2184. [[CrossRef](#)]
35. Xu, Y.; Lin, J.; Shi, J.; Zhang, G.; Wang, X.; Li, H. Robust self-supervised LiDAR odometry via representative structure discovery and 3D inherent error modeling. *IEEE Robot. Autom. Lett.* **2022**, *7*, 1651–1658. [[CrossRef](#)]
36. Nunes, L.; Marcuzzi, R.; Chen, X.; Behley, J.; Stachniss, C. SegContrast: 3D point cloud feature representation learning through self-supervised segment discrimination. *IEEE Robot. Autom. Lett.* **2022**, *7*, 2116–2123. [[CrossRef](#)]
37. He, K.; Fan, H.; Wu, Y.; Xie, S.; Girshick, R. Momentum contrast for unsupervised visual representation learning. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 14–19 June 2020; pp. 9726–9735. [[CrossRef](#)]
38. Wang, B.; Ma, G.; Zhu, M. Fast momentum contrast learning for unsupervised person re-identification. *IEEE Signal Process. Lett.* **2021**, *28*, 2073–2077. [[CrossRef](#)]
39. Huang, J.; Yuan, J.; Qiao, C. Generation for unsupervised domain adaptation: A Gan-based approach for object classification with 3D point cloud data. In Proceedings of the ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Singapore, 22–27 May 2022; pp. 3753–3757. [[CrossRef](#)]
40. Toth, C.K. The strip adjustment and registration. In *Topographic Laser Ranging and Scanning Principles and Processing*; Shan, J., Toth, C.K., Eds.; CRC Press: Boca Raton, FL, USA, 2008; pp. 254–2287.
41. Wen, C.; Yang, L.; Li, X.; Peng, L.; Chi, T. Directionally constrained fully convolutional neural network for airborne LiDAR point cloud classification. *ISPRS J. Photogramm. Remote Sens.* **2020**, *162*, 50–62. [[CrossRef](#)]
42. Fang, L.; Sun, T.; Wang, S.; Fan, H.; Li, J. A graph attention network for road marking classification from mobile LiDAR point clouds. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *108*, 102735. [[CrossRef](#)]
43. Murray, X.; Apan, A.; Deo, R.; Maraseni, T. Rapid assessment of mine rehabilitation areas with airborne LiDAR and deep learning: Bauxite strip mining in Queensland, Australia. *Geocarto Int.* **2022**, 1–24. [[CrossRef](#)]

44. Cao, D.; Xing, H.; Wong, M.S.; Kwan, M.-P.; Xing, H.; Meng, Y.A. Stacking ensemble deep learning model for building extraction from remote sensing images. *Remote Sens.* **2021**, *13*, 3898. [[CrossRef](#)]
45. Shirmard, H.; Farahbakhsh, E.; Müller, R.D.; Chandra, R. A review of machine learning in processing remote sensing data for mineral exploration. *Remote Sens. Environ.* **2022**, *268*, 112750. [[CrossRef](#)]
46. Nahhas, F.H.; Shafri, H.Z.M.; Sameen, M.I.; Pradhan, B.; Mansor, S. Deep learning approach for building detection using LiDAR–orthophoto fusion. *Hindawi J. Sens.* **2018**, *2018*, 7212307. [[CrossRef](#)]
47. Zhang, P.; He, H.; Wang, Y.; Liu, Y.; Lin, H.; Guo, L.; Yang, W. 3D urban buildings extraction based on airborne lidar and photogrammetric point cloud fusion according to U-Net deep learning model segmentation. *IEEE Access* **2022**, *10*, 20889–20897. [[CrossRef](#)]
48. Shi, C.; Li, J.; Gong, J.; Yang, B.; Zhang, G. An improved lightweight deep neural network with knowledge distillation for local feature extraction and visual localization using images and LiDAR point clouds. *ISPRS J. Photogramm. Remote Sens.* **2022**, *184*, 177–188. [[CrossRef](#)]
49. Pu, R. *Hyperspectral Remote Sensing: Fundamentals and Practices*; Taylor & Francis Group: Boca Raton, FL, USA, 2017; p. 466; ISBN 9781498731591.
50. Marrs, J.; Ni-Meister, W. Machine learning techniques for tree species classification using co-registered LiDAR and hyperspectral data. *Remote Sens.* **2019**, *11*, 819. [[CrossRef](#)]
51. Yu, X.; Hyyppä, J.; Litkey, P.; Kaartinen, H.; Vastaranta, M.; Holopainen, M. Single-sensor solution to tree species classification using multispectral airborne laser scanning. *Remote Sens.* **2017**, *9*, 108. [[CrossRef](#)]
52. Zhao, P.; Guan, H.; Li, D.; Yu, Y.; Wang, H.; Gao, K.; Junior, J.M.; Li, J. Airborne multispectral LiDAR point cloud classification with a feature Reasoning-based graph convolution network. *Int. J. Appl. Earth Obs. Geoinf.* **2021**, *105*, 102634. [[CrossRef](#)]
53. Zhou, Q.; Yu, L.; Zhang, X.; Liu, Y.; Zhan, Z.; Ren, L.; Luo, Y. Fusion of UAV Hyperspectral Imaging and LiDAR for the Early Detection of EAB Stress in Ash and a New EAB Detection Index—NDVI_(776,678). *Remote Sens.* **2022**, *14*, 2428. [[CrossRef](#)]
54. Peng, Y.; Zhang, Y.; Tu, B.; Zhou, C.; Li, Q. Multiview Hierarchical Network for Hyperspectral and LiDAR Data Classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2022**, *15*, 1454–1469. [[CrossRef](#)]
55. Stilla, U.; Jutzi, B. Waveform analysis for small-footprint pulsed Laser systems. In *Topographic Laser Ranging and Scanning Principles and Processing*; Shan, J., Toth, C.K., Eds.; CRC Press: Boca Raton, FL, USA, 2008; pp. 234–253.
56. Guan, H.; Yu, Y.; Ji, Z.; Li, J.; Zhang, Q. Deep learning-based tree classification using mobile Lidar data. *Remote Sens. Lett.* **2015**, *6*, 864–873. [[CrossRef](#)]
57. Blomley, R.; Hovi, A.; Weinmann, M.; Hinz, S.; Korpela, I.; Jutzi, B. Tree species classification using within crown localization of waveform LiDAR attributes. *ISPRS J. Photogramm. Remote Sens.* **2017**, *133*, 142–156. [[CrossRef](#)]
58. Yang, G.; Zhao, Y.; Li, B.; Ma, Y.; Li, R.; Jing, J.; Dian, Y. Tree species classification by employing multiple features acquired from integrated sensors. *J. Sens.* **2019**, *2019*, 1–12. [[CrossRef](#)]
59. Shinohara, T.; Xiu, H.; Matsuoka, M. FWNNet: Semantic segmentation for full-waveform LiDAR data using deep learning. *Sensors* **2020**, *20*, 3568. [[CrossRef](#)]
60. Shin, Y.H.; Son, K.W.; Lee, D.C. Semantic segmentation and building extraction from airborne LiDAR data with multiple return using PointNet++. *Appl. Sci.* **2022**, *12*, 1975. [[CrossRef](#)]
61. Qi, C.R.; Yi, L.; Su, H.; Guibas, L.J. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Proc. Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5099–5108.
62. Zhang, Q.; Ge, L.; Hensley, S.; Metternicht, G.I.; Liu, C.; Zhang, R. PolGAN: A deep-learning-based unsupervised forest height estimation based on the synergy of PolInSAR and LiDAR data. *ISPRS J. Photogramm. Remote Sens.* **2022**, *186*, 123–139. [[CrossRef](#)]
63. Park, Y.; Guldmann, J.M. Creating 3D city models with building footprints and LiDAR point cloud classification: A machine learning approach. *Comput. Environ. Urban Syst.* **2019**, *75*, 76–89. [[CrossRef](#)]
64. Feng, C.C.; Guo, Z. Automating parameter learning for classifying terrestrial LiDAR point cloud using 2D land cover maps. *Remote Sens.* **2018**, *10*, 1192. [[CrossRef](#)]
65. Schmohl, S.; Narváez Vallejo, A.; Soergel, U. Individual tree detection in urban ALS point clouds with 3D convolutional networks. *Remote Sens.* **2022**, *14*, 1317. [[CrossRef](#)]
66. Kogut, T.; Tomczak, A.; Słowik, A.; Oberski, T. Seabed modelling by means of airborne laser bathymetry data and imbalanced learning for offshore mapping. *Sensors* **2022**, *22*, 3121. [[CrossRef](#)] [[PubMed](#)]
67. Barbarella, M.; Di Benedetto, A.; Fiani, M. Application of Supervised Machine Learning Technique on LiDAR Data for Monitoring Coastal Land Evolution. *Remote Sens.* **2021**, *13*, 4782. [[CrossRef](#)]
68. Duran, Z.; Ozcan, K.; Atik, M.E. Classification of photogrammetric and airborne LiDAR point clouds using machine learning algorithms. *Drones* **2021**, *5*, 104. [[CrossRef](#)]
69. Mohammed, M.; Badruddin Khan, M.; Bashier, E.B.M. *Machine Learning Algorithms and Applications*, 1st ed.; CRC Press: Boca Raton, FL, USA, 2016; p. 226. [[CrossRef](#)]
70. Kim, P. *MATLAB Deep Learning with Machine Learning, Neural Networks and Artificial Intelligence*; Apress: Berkeley, CA, USA, 2017; p. 151. [[CrossRef](#)]
71. Mirzaei, K.; Arashpour, M.; Asadi, E.; Masoumi, H.; Bai, Y.; Behnood, A. 3D point cloud data processing with machine learning for construction and infrastructure applications: A comprehensive review. *Adv. Eng. Inform.* **2022**, *51*, 101501; ISSN 1474-0346. [[CrossRef](#)]

72. Maturana, D.; Scherer, S. Voxnet: A 3D convolutional neural network for real-time object recognition. In Proceedings of the 2015 IEEE/RISJ International Conference on Intelligent Robots and Systems (IROS), Hamburg, Germany, 28 September–3 October 2015; pp. 922–928. [\[CrossRef\]](#)
73. Gargoum, S.A.; Koch, J.C.; El-Basyouny, K. A voxel-based method for automated detection and mapping of light poles on rural highways using lidar data. *Transp. Res. Rec.* **2018**, *2672*, 274–283. [\[CrossRef\]](#)
74. Shuang, F.; Huang, H.; Li, Y.; Qu, R.; Li, P. AFE-RCNN: Adaptive feature enhancement RCNN for 3D object detection. *Remote Sens.* **2022**, *14*, 1176. [\[CrossRef\]](#)
75. Wijaya, K.T.; Paek, D.; Kong, S.H. Multiview attention for 3D object detection in Lidar point cloud. In Proceedings of the 2022 International Conference on Artificial Intelligence in Information and Communication (ICAIIIC), Jeju Island, Korea, 21–24 February 2022; pp. 210–215. [\[CrossRef\]](#)
76. Simonovsky, M.; Komodakis, N. Dynamic edge-conditioned filters in convolutional neural networks on graphs. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 29–38. [\[CrossRef\]](#)
77. Wang, C.; Samari, B.; Siddiqi, K. Local spectral graph convolution for point set feature learning. In *Computer Vision—ECCV 2018. Lecture Notes in Computer Science*; Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y., Eds.; Springer: Cham, Switzerland, 2018; p. 11208. [\[CrossRef\]](#)
78. Li, D.; Shen, X.; Guan, H.; Yu, Y.; Wang, H.; Zhang, G.; Li, J.; Li, D. AGFP-Net: Attentive geometric feature pyramid network for land cover classification using airborne multispectral LiDAR data. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *108*, 102723; ISSN 0303-2434. [\[CrossRef\]](#)
79. Wen, C.; Li, X.; Yao, X.; Peng, L.; Chi, T. Airborne LiDAR point cloud classification with global-local graph attention convolution neural network. *ISPRS J. Photogramm. Remote Sens.* **2021**, *173*, 181–194. [\[CrossRef\]](#)
80. Jing, W.; Zhang, W.; Li, L.; Di, D.; Chen, G.; Wang, J. AGNet: An attention-based graph network for point cloud classification and segmentation. *Remote Sens.* **2022**, *14*, 1036. [\[CrossRef\]](#)
81. Chen, Y.; Luo, Z.; Li, W.; Lin, H.; Nurunnabi, A.; Lin, Y.; Wang, C.; Zhang, X.-P.; Li, J. WGNNet: Wider graph convolution networks for 3D point cloud classification with local dilated connecting and context-aware. *Int. J. Appl. Earth Obs. Geoinf.* **2022**, *110*, 102786. [\[CrossRef\]](#)
82. Defferrard, M.; Bresson, X.; Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 3844–3852. [\[CrossRef\]](#)
83. Wan, J.; Xie, Z.; Xu, Y.; Zeng, Z.; Yuan, D.; Qiu, Q. DGANet: A dilated graph attention-based network for local feature extraction on 3D point clouds. *Remote Sens.* **2021**, *13*, 3484. [\[CrossRef\]](#)
84. Hua, B.; Tran, M.; Yeung, S. Pointwise convolutional neural networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 984–993. [\[CrossRef\]](#)
85. Klokov, R.; Lempitsky, V. Escape from cells: Deep KD-networks for the recognition of 3D point cloud models. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 863–872. [\[CrossRef\]](#)
86. Song, Y.; He, F.; Duan, Y.; Liang, Y.; Yan, X. A kernel correlation-based approach to adaptively acquire local features for learning 3D point clouds. *Comput.-Aided Des.* **2022**, *146*, 103196, ISSN 0010-4485. [\[CrossRef\]](#)
87. Tarsha Kurdi, F.; Amakhchan, W.; Gharineiat, Z. Random Forest machine learning technique for automatic vegetation detection and modelling in LiDAR data. *Int. J. Environ. Sci. Nat. Resour.* **2021**, *28*, 556234. [\[CrossRef\]](#)
88. Chen, W.; Li, X.; Wang, Y.; Chen, G.; Liu, S. Forested landslide detection using LiDAR data and the random forest algorithm: A case study of the three Gorges, China. *Remote Sens. Environ.* **2014**, *152*, 291–301. [\[CrossRef\]](#)
89. Huang, R.; Zhu, J. Using Random Forest to integrate LiDAR data and hyperspectral imagery for land cover classification. In Proceedings of the 2013 IEEE International Geoscience and Remote Sensing Symposium—IGARSS, Melbourne, Australia, 21–26 July 2013; pp. 3978–3981. [\[CrossRef\]](#)
90. Man, Q.; Dong, P.; Yang, X.; Wu, Q.; Han, R. Automatic extraction of grasses and individual trees in urban areas based on airborne hyperspectral and LiDAR data. *Remote Sens.* **2020**, *12*, 2725. [\[CrossRef\]](#)
91. Yu, X.; Hyyppä, J.; Vastaranta, M.; Holopainen, M.; Viitala, R. Predicting individual tree attributes from airborne laser point clouds based on the random forests technique. *ISPRS J. Photogramm. Remote Sens.* **2011**, *66*, 28–37. [\[CrossRef\]](#)
92. Levick, S.R.; Hessenmöller, D.; Schulze, E.D. Scaling wood volume estimates from inventory plots to landscapes with airborne LiDAR in temperate deciduous forest. *Carbon Balance Manag.* **2016**, *11*, 7. [\[CrossRef\]](#)
93. Guan, H.; Yu, J.; Li, J.; Luo, L. Random forests-based feature selection for land-use classification using LiDAR data and orthoimagery. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2012**, *XXXIX-B7*, 203–208. [\[CrossRef\]](#)
94. Ba, A.; Laslier, M.; Dufour, S.; Hubert-Moy, L. Riparian trees genera identification based on leaf-on/leaf-off airborne laser scanner data and machine learning classifiers in northern France. *Int. J. Remote Sens.* **2020**, *41*, 1645–1667. [\[CrossRef\]](#)
95. Arumäe, T.; Lang, M.; Sims, A.; Laarmann, D. Planning of commercial thinnings using machine learning and airborne lidar data. *Forests* **2022**, *13*, 206. [\[CrossRef\]](#)
96. Dong, Y.; Li, Y.; Hou, M. The point cloud semantic segmentation method for the Ming and Qing dynasties’ official-style architecture roof considering the construction regulations. *ISPRS Int. J. Geo-Inf.* **2022**, *11*, 214. [\[CrossRef\]](#)

97. Liao, L.; Tang, S.; Liao, J.; Li, X.; Wang, W.; Li, Y.; Guo, R. A Supervoxel-based random forest method for robust and effective airborne LiDAR point cloud classification. *Remote Sens.* **2022**, *14*, 1516. [[CrossRef](#)]
98. Hoang, L.; Lee, S.H.; Kwon, K.R. A 3D shape recognition method using hybrid deep learning network CNN-SVM. *Electronics* **2020**, *9*, 649. [[CrossRef](#)]
99. Zhang, J.; Lin, X.; Ning, X. SVM-based classification of segmented airborne LiDAR point clouds in urban areas. *Remote Sens.* **2013**, *5*, 3749–3775. [[CrossRef](#)]
100. Shokri, D.; Rastiveis, H.; Sheikholeslami, S.M.; Shah-hosseini, R.; Li, J. Fast extraction of power lines from mobile LiDAR point clouds based on SVM classification in non-urban area. *Earth Obs. Geomat. Eng.* **2022**, *5*, 63–73. [[CrossRef](#)]
101. Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial nets. *Advances in neural information processing systems*. In Proceedings of the International Conference on Neural Information Processing Systems (NIPS 2014), Montreal, QC, Canada, 8–13 December 2014; Volume 27, pp. 2672–2680.
102. Wei, J.; Lin, G.; Yap, K.H.; Hung, T.Y.; Xie, L. Multi-path region mining for weakly supervised 3D semantic segmentation on point clouds. In Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 4383–4392. [[CrossRef](#)]
103. Li, N.; Kähler, O.; Pfeifer, N. A comparison of deep learning methods for airborne Lidar point clouds classification. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2021**, *14*, 6467–6486. [[CrossRef](#)]
104. Graham, B.; Engelcke, M.; Maaten, L.V.D. 3D semantic segmentation with submanifold sparse convolutional networks. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 9224–9232. [[CrossRef](#)]
105. Thomas, H.; Qi, C.R.; Deschaud, J.E.; Marcotegui, B.; Goulette, F.; Guibas, L.J. KPConv: Flexible and deformable convolution for point clouds. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27 October–2 November 2019; pp. 6411–6420. [[CrossRef](#)]
106. Théodose, R.; Denis, D.; Chateau, T.; Frémont, V.; Checchin, P. A deep learning approach for LiDAR resolution-agnostic object detection. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 14582–14593. [[CrossRef](#)]
107. Hoang, L.; Lee, S.H.; Lee, E.J.; Kwon, K.R. GSV-NET: A Multi-modal deep learning network for 3D point cloud classification. *Appl. Sci.* **2022**, *12*, 483. [[CrossRef](#)]
108. Chen, J.; Kakillioglu, B.; Velipasalar, S. Background-aware 3-D point cloud segmentation with dynamic point feature aggregation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–12. [[CrossRef](#)]
109. Song, W.; Li, D.; Sun, S.; Zhang, L.; Xin, Y.; Sung, Y.; Choi, R. 2D&3DNet for 3D object classification in LiDAR point cloud. *Remote Sens.* **2022**, *14*, 3146.
110. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [[CrossRef](#)] [[PubMed](#)]
111. Medina, F.P.; Paffenroth, R. Machine learning in LiDAR 3D point clouds. In *Advances in Data Science*; Association for Women in Mathematics Series, 26; Demir, I., Lou, Y., Wang, X., Welker, K., Eds.; Springer: Cham, Switzerland, 2021. [[CrossRef](#)]
112. Wu, W.; Qi, Z.; Fuxin, L. Pointconv: Deep convolutional networks on 3D point clouds. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 9613–9622. [[CrossRef](#)]
113. Ibrahim, M.; Akhtar, N.; Ullah, K.; Mian, A. Exploiting Structured CNNs for Semantic Segmentation of Unstructured Point Clouds from LiDAR Sensor. *Remote Sens.* **2021**, *13*, 3621. [[CrossRef](#)]
114. Hamedianfar, A.; Mohamedou, C.; Kangas, A.; Vauhkonen, J. Deep learning for forest inventory and planning: A critical review on the remote sensing approaches so far and prospects for further applications. *Forestry* **2022**, *95*, 451–465. [[CrossRef](#)]
115. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional networks for biomedical image segmentation. In Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241, LNCS. 9351. [[CrossRef](#)]
116. Ojogbane, S.S.; Mansor, S.; Kalantar, B.; Khuzaimah, Z.B.; Shafri, H.Z.M.; Ueda, N. Automated building detection from airborne LiDAR and very high-resolution aerial imagery with deep neural network. *Remote Sens.* **2021**, *13*, 4803. [[CrossRef](#)]
117. Seydi, S.T.; Hasanlou, M.; Amani, M. A new end-to-end multi-dimensional CNN framework for land cover/land use change detection in multi-source remote sensing datasets. *Remote Sens.* **2020**, *12*, 2010. [[CrossRef](#)]
118. Wang, Q.; Gu, Y. A Discriminative tensor representation model for feature extraction and classification of multispectral LiDAR data. *IEEE Trans. Geosci. Remote Sens.* **2020**, *58*, 1568–1586. [[CrossRef](#)]
119. Ekhtari, N.; Glennie, C.; Fernandez-Diaz, J.C. Classification of airborne multispectral lidar point clouds for land cover mapping. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2018**, *11*, 2068–2078. [[CrossRef](#)]
120. Zhao, G.; Zhang, W.; Peng, Y.; Wu, H.; Wang, Z.; Cheng, L. PEMCNet: An efficient multi-scale point feature fusion network for 3D LiDAR point cloud classification. *Remote Sens.* **2021**, *13*, 4312. [[CrossRef](#)]
121. Wen, S.; Wang, T.; Tao, S. Hybrid CNN-LSTM architecture for LiDAR point clouds semantic segmentation. *IEEE Robot. Autom. Lett.* **2022**, *7*, 5811–5818. [[CrossRef](#)]
122. Shuang, F.; Li, P.; Li, Y.; Zhang, Z.; Li, X. MSIDA-Net: Point Cloud Semantic Segmentation via Multi-Spatial Information and Dual Adaptive Blocks. *Remote Sens.* **2022**, *14*, 2187. [[CrossRef](#)]

123. Chen, M.; Qiu, X.; Zeng, W.; Peng, D. Combining sample plot stratification and machine learning algorithms to improve forest aboveground carbon density estimation in northeast China using airborne LiDAR data. *Remote Sens.* **2022**, *14*, 1477. [[CrossRef](#)]
124. Luo, Z.; Zhang, Z.; Li, W.; Chen, Y.; Wang, C.; Nurunnabi, A.; Li, J. Detection of individual trees in UAV LiDAR point clouds using a deep learning framework based on multichannel representation. *IEEE Trans. Geosci. Remote Sens.* **2022**, *60*, 1–15. [[CrossRef](#)]
125. Corte, A.P.D.; Souza, D.V.; Rex, F.E.; Sanquetta, C.R.; Mohan, M.; Silva, C.A.; Zambrano, A.M.A.; Prata, G.; Alves de Almeida, D.R.; Trautenmüller, J.W.; et al. Forest inventory with high-density UAV-Lidar: Machine learning approaches for predicting individual tree attributes. *Comput. Electron. Agric.* **2020**, *179*, 105815. [[CrossRef](#)]
126. Windrim, L.; Bryson, M. Detection, segmentation, and model fitting of individual tree stems from airborne laser scanning of forests using deep learning. *Remote Sens.* **2020**, *12*, 1469. [[CrossRef](#)]
127. Chen, X.; Jiang, K.; Zhu, Y.; Wang, X.; Yun, T. Individual tree crown segmentation directly from UAV-borne LiDAR data using the PointNet of deep learning. *Forests* **2021**, *12*, 131. [[CrossRef](#)]
128. Mizoguchi, T.; Ishii, A.; Nakamura, H.; Inoue, T.; Takamatsu, H. *Lidar-Based Individual Tree Species Classification Using Convolutional Neural Network*; Proc. SPIE 10332, Videometrics, Range Imaging, and Applications XIV; SPIE Optical Metrology: Munich, Germany, 2017; p. 103320O. [[CrossRef](#)]
129. Budei, B.; St-Onge, B.; Hopkinson, C.; Audet, F.A. Identifying the genus or species of individual trees using a three-wavelength airborne Lidar system. *Remote Sens. Environ.* **2018**, *204*, 632–647. [[CrossRef](#)]
130. Nguyen, H.; Demir, B.; Dalponte, M. Weighted support vector machines for tree species classification using Lidar data. In Proceedings of the IGARSS 2019–2019 IEEE International Geoscience and Remote Sensing Symposium, Yokohama, Japan, 28 July–2 August 2019; pp. 6740–6743. [[CrossRef](#)]
131. Hell, M.; Brandmeier, M.; Briechle, S.; Krzystek, P. Classification of tree species and standing dead trees with Lidar point clouds using two deep neural networks: PointCNN and 3DmFV-Net. *PFG* **2022**, *90*, 103–121. [[CrossRef](#)]
132. Li, Y.; Bu, R.; Sun, M.; Wu, W.; Di, X.; Chen, B. PointCNN: Convolution on x-transformed points. In Proceedings of the Advances in Neural Information processing systems 31 (NIPS 2018), Montreal, QC, Canada, 3–8 December 2018; pp. 820–830. [[CrossRef](#)]
133. Ben-Shabat, Y.; Lindenbaum, M.; Fischer, A. 3DmFV: Three-dimensional point cloud classification in real-time using convolutional neural networks. *IEEE Robot Autom. Lett.* **2018**, *3*, 3145–3152. [[CrossRef](#)]
134. Wen, C.; Sun, X.; Li, J.; Wang, C.; Guo, Y.; Habib, A. A deep learning framework for road marking extraction, classification and completion from mobile laser scanning point clouds. *ISPRS J. Photogramm. Remote Sens.* **2019**, *147*, 178–192. [[CrossRef](#)]
135. Ma, L.; Li, Y.; Li, J.; Yu, Y.; Junior, J.M.; Goncalves, W.N.; Chapman, M.A. Capsule-based networks for road marking extraction and classification from mobile LiDAR point clouds. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 1981–1995. [[CrossRef](#)]
136. Ma, H.; Ma, H.; Zhang, L.; Liu, K.; Luo, W. Extracting urban road footprints from airborne LiDAR point clouds with PointNet++ and two-step post-processing. *Remote Sens.* **2022**, *14*, 789. [[CrossRef](#)]
137. Shajahan, D.A.; Nayel, V.; Muthuganapathy, R. Roof classification from 3-D LiDAR point clouds using multiview CNN with self-attention. *IEEE Geosci. Remote Sens. Lett.* **2020**, *17*, 1465–1469. [[CrossRef](#)]
138. Silva, A.; Fernandes, D.; Névoa, R.; Monteiro, J.; Novais, P.; Girão, P.; Afonso, T.; Melo-Pinto, P. Resource-constrained onboard inference of 3D object detection and localisation in point clouds targeting self-driving applications. *Sensors* **2021**, *21*, 7933. [[CrossRef](#)]
139. Lee, Y.; Park, S. A Deep Learning-Based Perception Algorithm Using 3D LiDAR for Autonomous Driving: Simultaneous Segmentation and Detection Network (SSADNet). *Appl. Sci.* **2020**, *10*, 4486. [[CrossRef](#)]
140. Kang, D.; Wong, A.; Lee, B.; Kim, J. Real-time semantic segmentation of 3D point cloud for autonomous driving. *Electronics* **2021**, *10*, 1960. [[CrossRef](#)]
141. Sun, Y.; Zuo, W.; Huang, H.; Cai, P.; Liu, M. PointMoSeg: Sparse Tensor-Based End-to-End Moving-Obstacle Segmentation in 3-D Lidar Point Clouds for Autonomous Driving. *IEEE Robot. Autom. Lett.* **2022**, *6*, 510–517. [[CrossRef](#)]
142. Peng, K.; Fei, J.; Yang, K.; Roitberg, A.; Zhang, J.; Bieder, F.; Heidenreich, P.; Stiller, C.; Stiefelwagen, R. MASS: Multi-Attentional Semantic Segmentation of LiDAR Data for Dense Top-View Understanding. *IEEE Trans. Intell. Transp. Syst.* **2022**, *23*, 15824–15840. [[CrossRef](#)]
143. Gao, R.; Li, M.; Yang, S.J.; Cho, K. Reflective noise filtering of large-scale point cloud using transformer. *Remote Sens.* **2022**, *14*, 577. [[CrossRef](#)]
144. Nurunnabi, A.; Teferle, F.N.; Li, J.; Lindenbergh, R.C.; Hunegnaw, A. An efficient deep learning approach for ground point filtering in aerial laser scanning point clouds. *Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci.* **2021**, *XLIII-B1-2021*, 31–38. [[CrossRef](#)]
145. Cao, Y.; Scaioni, M. 3DLEB-Net: Label-efficient deep learning-based semantic segmentation of building point clouds at LoD3 level. *Appl. Sci.* **2021**, *11*, 8996. [[CrossRef](#)]
146. Zhang, J.; Xiao, W.; Mills, J.P. Optimizing moving object trajectories from roadside Lidar data by joint detection and tracking. *Remote Sens.* **2022**, *14*, 2124. [[CrossRef](#)]
147. Shi, S.; Guo, C.; Jiang, L.; Wang, Z.; Shi, J.; Wang, X.; Li, H. PV-RCNN: Point-voxel feature set abstraction for 3D object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 13–19 June 2020; pp. 10526–10535. [[CrossRef](#)]
148. Yin, C.; Cheng, J.C.P.; Wang, B.; Gan, V.J.L. Automated classification of piping components from 3D LiDAR point clouds using SE-PseudoGrid. *Autom. Constr.* **2022**, *139*, 104300. [[CrossRef](#)]

-
149. Amakhchan, W.; Tarsha Kurdi, F.; Gharineiat, Z.; Boulaassal, H.; El Kharki, O. Automatic filtering of LiDAR building point cloud using multilayer perceptron Neuron Network. In Proceedings of the 3rd International Conference on Big Data and Machine Learning (BML'22), Istanbul, Turkey, 22–23 December 2022. Available online: <https://bml.maasi.org/> (accessed on 9 August 2022).
 150. Mammoliti, E.; Di Stefano, F.; Fronzi, D.; Mancini, A.; Malinverni, E.S.; Tazioli, A.A. Machine learning approach to extract rock mass discontinuity orientation and spacing, from laser scanner point clouds. *Remote Sens.* **2022**, *14*, 2365. [[CrossRef](#)]