Classification of Compressed Full-Waveform Airborne Lidar Data

Sadiq Olayiwola Macaulay[a], Eleonora Maset[a], Andrea Fusiello[a]

[a]Polytechnic Department of Engineering and Architecture, University of Udine, Via Delle Scienze, 206 - 33100 Udine, Italy

ARTICLE HISTORY
Compiled 10th July 2024

Abstract
Airborne laser scanners produce 3D data that can be used for a range of applications, such as urban planning, facility monitoring, flood mapping, and forest management. Additional information on the surveyed area can be obtained from the backscattered waveforms recorded by modern light detection and ranging (lidar) sensors. However, the high-dimensional representation of full-waveform data has hindered progress in its use due to difficulties in processing and storage. This paper develops a quantized convolutional autoencoder network to compress lidar waveform data into a condensed feature representation, resulting in a compression rate of up to 20:1. This, together with height information, is fed into a U-net convolutional neural network that achieves an accuracy of 93.7% on six classes.

KEYWORDS
Lidar; Full-waveform; Data compression; Classification

1. Introduction

Airborne laser scanners (ALS) provide detailed 3D point clouds of the terrain and surface objects. After data classification, essential for accurate interpretation, ALS point clouds can be used for a variety of applications, such as urban planning, facility monitoring, flood mapping, and forest management. This surveying technique can also be employed to study changes in the environment over time and identify areas at risk from natural disasters.

The development of light detection and ranging (lidar) sensors acquiring point clouds and full-waveform (FW) data has led to improved classification results, because the waveform of the backscattered signal is related to the physical qualities of the reflecting surfaces, such as the material, angle of incidence, and geometry (Molnar, Laky, and Toth 2013). As a matter of fact, several methods of FW classification have been proposed in recent years, including decision trees (Wagner et al. 2008), Support Vector Machine (Mallet et al. 2011), a Random Forest algorithm (Guo et al. 2011), Self Organizing Maps (Maset, Carniel, and Crosilla 2015), artificial neural network (Höfle, Hollaus, and Hagenauer 2012; Zorzi et al. 2019b) and more. Many of these studies, including (Reitberger, Krzystek, and Stilla 2009) and (Molnar, Laky, and Toth 2013), proved that the additional information represented in the FW data could potentially improve classification metrics of aerial scans of urban regions. Pashaei et al.

CONTACT Eleonora Maset. Email:eleonora.maset@uniud.it

(2021, 2023) show that direct classification of raw FW outperforms the use of waveform attributes (e.g., amplitude and pulse deviation) assigned to each echo after FW decomposition and parametric modeling. Also Shinohara, Xiu, and Matsuoka (2020) demonstrate the potential of the FW signals which, coupled with 3D coordinates, are provided as input to a 3D deep learning model for ALS data classification. These recent works further confirm the rich information content of FW data, relevant for point cloud classification tasks.

However, the large amount of data produced by FW lidar sensors has created a challenging situation in terms of processing and storage, potentially slowing the pace of interest in exploiting the functionality of the FW lidar, particularly with data-driven classification methods. As a result, the potential of these data, already demonstrated in the literature, is not fully exploited in practice.

To address this issue, we present a novel approach to compress the full-waveform using a quantized convolutional autoencoder network. Our proposed method can compress FW data into a compact feature representation (we achieve a compression rate of up to 20:1) while still preserving relevant classification information. In contrast to the approach implemented by Shinohara, Xiu, and Matsuoka (2020), that relies on both the complete raw waveforms and the 3D point cloud to extract local and global features for classification, our method performs waveform compression regardless of the point coordinates and nearby object information. Therefore, our algorithm allows the FW data to be stored in compressed form as soon as it is recorded by the scanner and before the georeferencing step.

Subsequently, the compact feature vector can be directly fed to the classifier, with no need to reconstruct the original signal. To this end, we redesigned the network implemented by Zorzi et al. (2019b) to work with compressed FW data, with the aim of distinguishing among six classes: soil, vegetation, building, power line, transmission tower and street.

2. Proposed framework

We modify the model proposed by Zorzi et al. (2019b) in two ways. First, we introduce a convolutional autoencoder at the front-end, which is used to compress the FW data into a lower dimensional feature space for easier manipulation. In (Zorzi et al. 2019b), each waveform is fed into a 1D convolutional neural network that produces a $1 \times n$ vector, whose elements are the probabilities for the considered n classes. The output vector is treated as a compact representation of the FW that, however, depends on the target classes and requires a supervised training stage. The proposed autoencoder replaces therefore the n-class classifier used to represent the FW as a vector of probabilities (in the present work, we assume $n = 6$). The advantage of the autoencoder is that it is independent of the final classes, so the compressed representation can substitute the raw data in any subsequent use. A further reduction in memory consumption is achieved by applying a quantization scheme to the entire network model, resulting in less bits being used to represent the compressed waveform.

Second, we redesigned the output layer of the back-end U-net of Zorzi et al. (2019b) to support hierarchical classification, which gives the user more flexibility in deciding which labels to return.

These models are discussed in detail in the following subsections.

2.1. *Compression model*

The architecture of the waveform compression network follows the typical structure of a 1D autoencoder, with an encoding block that compresses the 1×160 raw input signal into a 1×8 code (latent-space representation) and a decoder that reconstructs the input from this code. The latter is used only during training and is then removed.

The encoding block is composed of three one-dimensional convolutional layers with kernel size 3, each followed by a rectified linear unit (ReLU) activation function and batch normalization. The output size of these layers decreases step-by-step, from 64 to 16 units. A max-pool layer followed by a dense layer is then employed to compact the feature map from the previous convolutional layer into a 8-dimensional compressed latent feature space. The decoding block includes dense layers that are the same size as the encoding block, which creates a reconstruction of the original waveform from the latent-space representation.

To further reduce the amount of data stored on the device (and the amount of computing power used in the classification stage as well), the quantized learning approach proposed by Jacob et al. (2018) is applied to this network model. In this way, the network weights and activations (including the code) are approximated by 8-bit integers. The compressed data is down to 5% of the original FW size while retaining important classification information. It occupies less storage space on local devices, with negligible performance degradation, and can be stored and loaded as a standard Waveform Data Packets (WDP) file. Figure 1 shows some sample waveforms with the corresponding decompressed signal for visual comparison.

2.2. *Classification model*

Following the approach by Zorzi et al. (2019b), the compressed waveform (1×8 8-bit integers) and height from the lidar point cloud data are combined and projected into a multi-channel image. In this way, exploiting the (x, y) point coordinates the 3D point cloud is converted into a 2D orthographic representation, that, however, maintains the (vertical) z information through the height channel and is enriched with the compressed waveform. The pixel size of the multi-channel image should be chosen according to the point cloud density (a pixel size of 0.05 m was used in our test). Since an image of arbitrary size can be created from the point cloud projection, a tiling with overlapping windows is then applied to obtain multi-channel patches of fixed size (in our case, $256 \times 256 \times 9$, corresponding to an area of 12.8 m $\times$ 12.8 m) that are fed into the second network. Its architecture follows a U-net model, i.e., a fully convolutional network widely employed in the literature to perform image segmentation tasks (Ronneberger, Fischer, and Brox 2015). Thanks to the U-net, a label is predicted for each pixel of the input image and, consequently, for the point falling in the pixel.

Unlike the previous architecture (Zorzi et al. 2019b), the network proposed in this paper is structured to support hierarchical classification, which gives the user more flexibility in deciding which labels to return.

Hierarchical categorisation is where the classes are arranged in a hierarchy (Silla and Freitas 2011). This means that each class has one or more child classes and a parent class. For example, the class ground is a parent class of the child classes soil and street (see, e.g., Fig. 2).

Hierarchical classification has several advantages over flat classification. Firstly, it provides a natural way to organize the results, giving the user the freedom to choose the granularity of the classification, usually trading accuracy with specificity. For example,

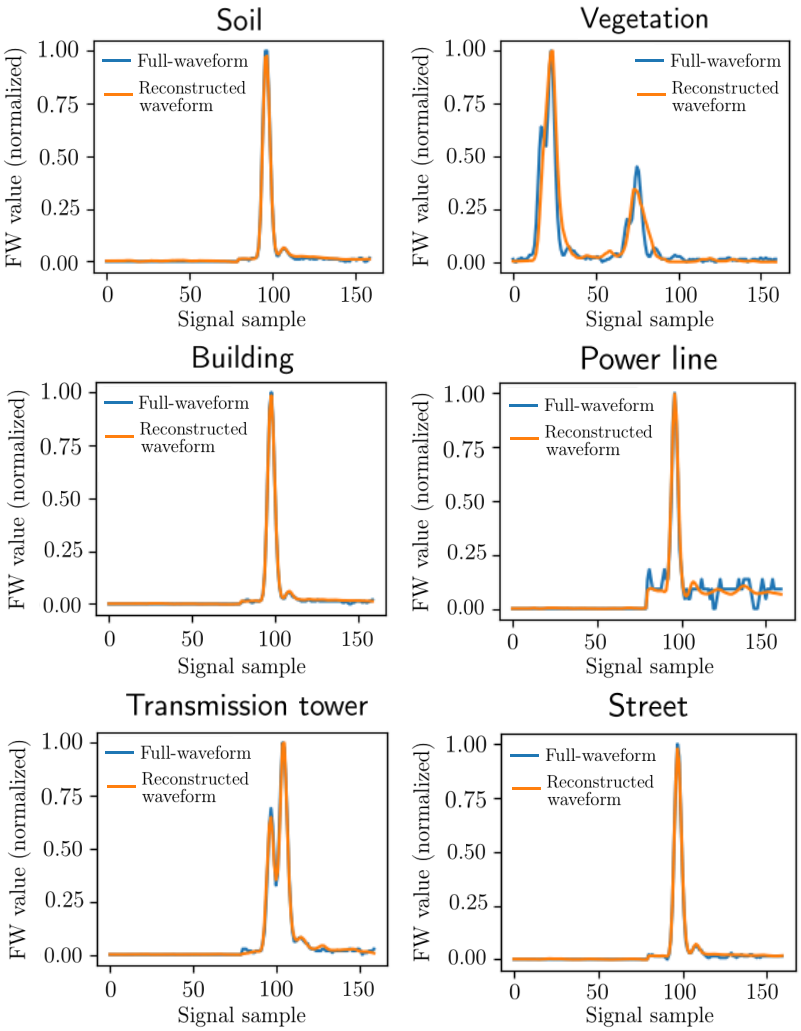

Figure 1. Examples of full-waveform (FW) signals corresponding to different materials (6 classes). The original and decompressed signals are shown for comparison

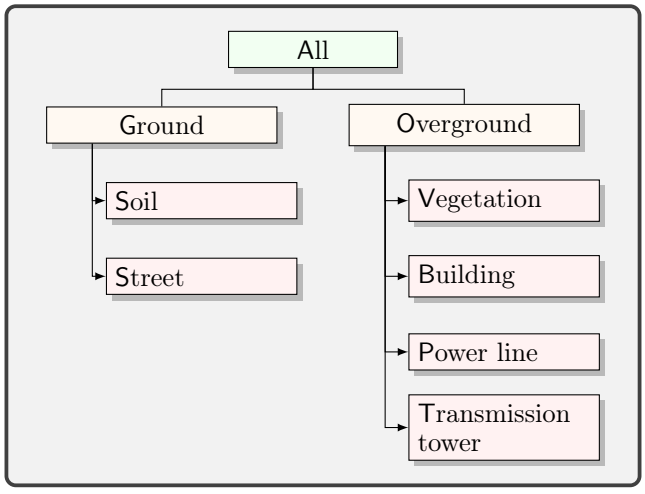

Figure 2. The classes hierarchy

in the context of the creation of digital terrain models, one might be interested in a ground vs overground classification instead of the finer categorization into 6 classes.

Second, it can improve the classification accuracy. This is because it takes into account the relationships between classes. For example, a ground sample can be properly recognised even if it does not fit neatly into the soil or street categories.

Consider also that the features of overground objects can vary widely in shape, size, surface material, etc., while the features of the ground class are mostly similar in shape but have greater differences in surface properties. The hierarchical classification approach allows to focus on discriminant during training, while effectively minimising the classification error in each parent node.

In particular, to achieve classification with different numbers of classes, we implemented on top of the U-net the global approach described in (Kiritchenko et al. 2006), which casts hierarchical classification as a multi-label classification problem. To do this, the training phase is modified to take into account all classes in the hierarchy by augmenting the non-leaf nodes with their ancestors' labels. During the test phase, the algorithm does not consider the hierarchy. Since it was trained on consistent data, most of the test instances will also be labelled consistently. However, there is no guarantee, and inconsistent labels may appear on some test instances, which occurs when the algorithm predicts a class that is not a child of the predicted parent class. For example, if the algorithm predicts that an object is a ground, but it also predicts that it is a vegetation, then this is an inconsistency. For this reason, Kiritchenko et al. (2006) propose a post-processing step to ensure that the predicted classes are consistent with the hierarchical structure.

3. Experimental results

In this section, we present the results obtained from the implementation of the model described in the previous section. All experiments were carried out on an Nvidia GeForce RTX 3070, Graphics Processing Unit (GPU)-capable device running Tensorflow (version 2.2.4) with Keras framework.

3.1. *Dataset*

The FW compression network and the U-net have been trained and evaluated with the dataset used in Zorzi et al. (2019b). The dataset, available on the web (Zorzi et al. (2019a)), contains point cloud data with corresponding FW data, intensity and labels. The FW data is represented by a vector consisting of 160 values of signal samples per point (the shortest waveforms are padded with zeros). The label data was manually annotated with numbers from one to six for soil, vegetation, building, power line, transmission tower and street, respectively. To encourage the usability of our implementation, the networks input and output are compliant to the standard ASPRS specification of LAS format that includes a Waveform Data Packets (WDP) file.

The dataset contains 9.8 million points and is characterized by a mean surface density of 45 points/m^2. For training and testing we adopted the original split of the dataset provided by Zorzi et al. (2019a). The test set contains approximately 10% of the total number of points, while the remainder is used for training the neural network model. The training set is further divided into 80% training/20% validation sets.

The street and soil classes together form the ground class. Overground contains vegetation, building, power line, and transmission tower (see Fig. 2).

The use of a public domain dataset allows reproducibility and comparison by other authors.

3.2. *Training Strategy*

To train the autoencoder, we used the Adam optimizer with an initial learning rate set to 0.001. As the loss function we used the correlation between the original waveform and the decompressed signal, the latter obtained as the output of the decoding block (Sect. 2.1). Training lasted for 50 epochs. The code length was determined by practical reasons: the loss function decreases monotonically with increasing code length, but we stopped at a size of 8 because the results were already comparable to (Zorzi et al. 2019b). In fact, the original model represented FW signals with a 1×6 probability vector, so this is quite consistent with previous results.

The classification model was trained on randomly cut out windows of size $256 \times 256 \times 9$ from the training set, ensuring that all classes are represented, as done in (Zorzi et al. 2019b). The network was trained through 50 epochs with categorical cross-entropy loss function and Adam optimizer with initial learning rate set to 0.001. Please note that although the prediction of the network can be be specified in terms of different number of classes, only one training is performed.

3.3. *Evaluation*

The simplest way to deal with hierarchical classification problems is the so-called *flat* classification, which ignores the class hierarchy altogether and only predicts classes at the leaf nodes. During training and testing, this method behaves like a standard classification algorithm. When a leaf class is assigned to an example, all of its parent classes are implicitly assigned to that instance.

Table 1 compares the overall accuracy of the new (hierarchical) model (right column) with the original figures from Zorzi et al. (2019b) (left column) and flat classification (central column) when predicting the six original classes, five classes (where street is grouped with soil to form ground) and two classes (ground/overground). The hierarchical classification method produced a slightly better prediction, with a 20-fold reduction in the size of the input data.

Table 1. Overall classification accuracy for different models and different classes

No. of classes	Zorzi et al. (2019b) (%)	Flat (%)	Hierarchical (%)
2	N.A.	97.2	97.4
5	96.1	96.5	96.7
6	92.6	93.2	93.7

As noted in (Zorzi et al. 2019b), points in the street class are sometimes confused with points in the soil class, and performance is higher when these two are grouped together. This observation is supported by Fig. 3, which shows the confusion matrices for the 6 and 5 classes. As one could expect, maximum accuracy is achieved in the two-class situation.

Figure 4 reports a sample output of the classification, where colors correspond to classes.

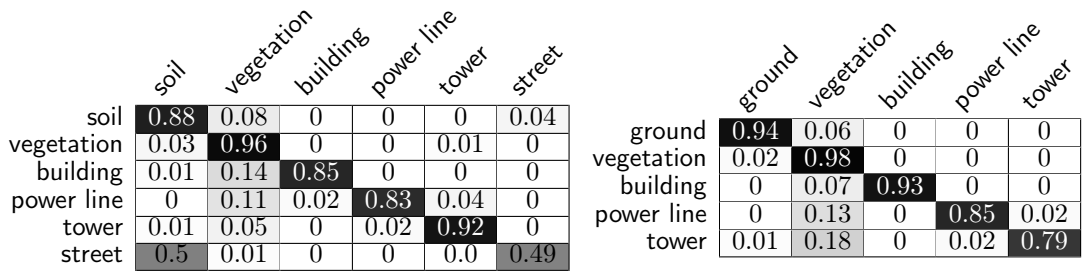

Figure 3. Confusion matrices: each row of the matrix represents the instances in an actual class while each column represents the instances in a predicted class. Values are normalized so that the sum of every row is equal to 1. Left: 6 classes. Right: 5 classes (street ∪ soil = ground).

4. Conclusions

Although the lidar waveform provides valuable insight into the physical properties of a surface, the combination of FW and point cloud data can be challenging to store and process. This paper builds on (Zorzi et al. 2019b) and presents an innovative method to compress and effectively store and classify lidar point clouds with FW data. The approach uses a quantized convolutional autoencoder network to learn a compact code that allows the subsequent classification of the FW data. Our method can dramatically compress waveform data into a low-dimensional feature space while effectively preserving the valuable information. The U-net classifier reads the compressed data and organises it into a hierarchy of classes that the user can slice at the desired level. The classification accuracy is slightly better than (Zorzi et al. 2019b), but the main advantage of our approach is that the FW data can be compressed at source and then stored on disk without the need to uncompress it to perform the classification, thereby solving a practical problem that has hindered the exploitation of FW data.

Disclosure statement

The authors report there are no competing interests to declare.

Funding

S.O.M. has been supported by the Italian "MISE PoC 2020" project for the exploitation of IPs, with reference to the EU Patent N. 3.794.376 "Apparatus and method for classification of backscattered full-waveform signals".

Data Availability Statement

The data that support the findings of this study are openly available at the link `https://uniudamce-my.sharepoint.com/:f:/g/personal/compvis_uniud_it/EshMZziamr5MjSIcqxhiesUB3R0EnYDnQbFdp_ae-K6xvQ?e=oEnxzN`.

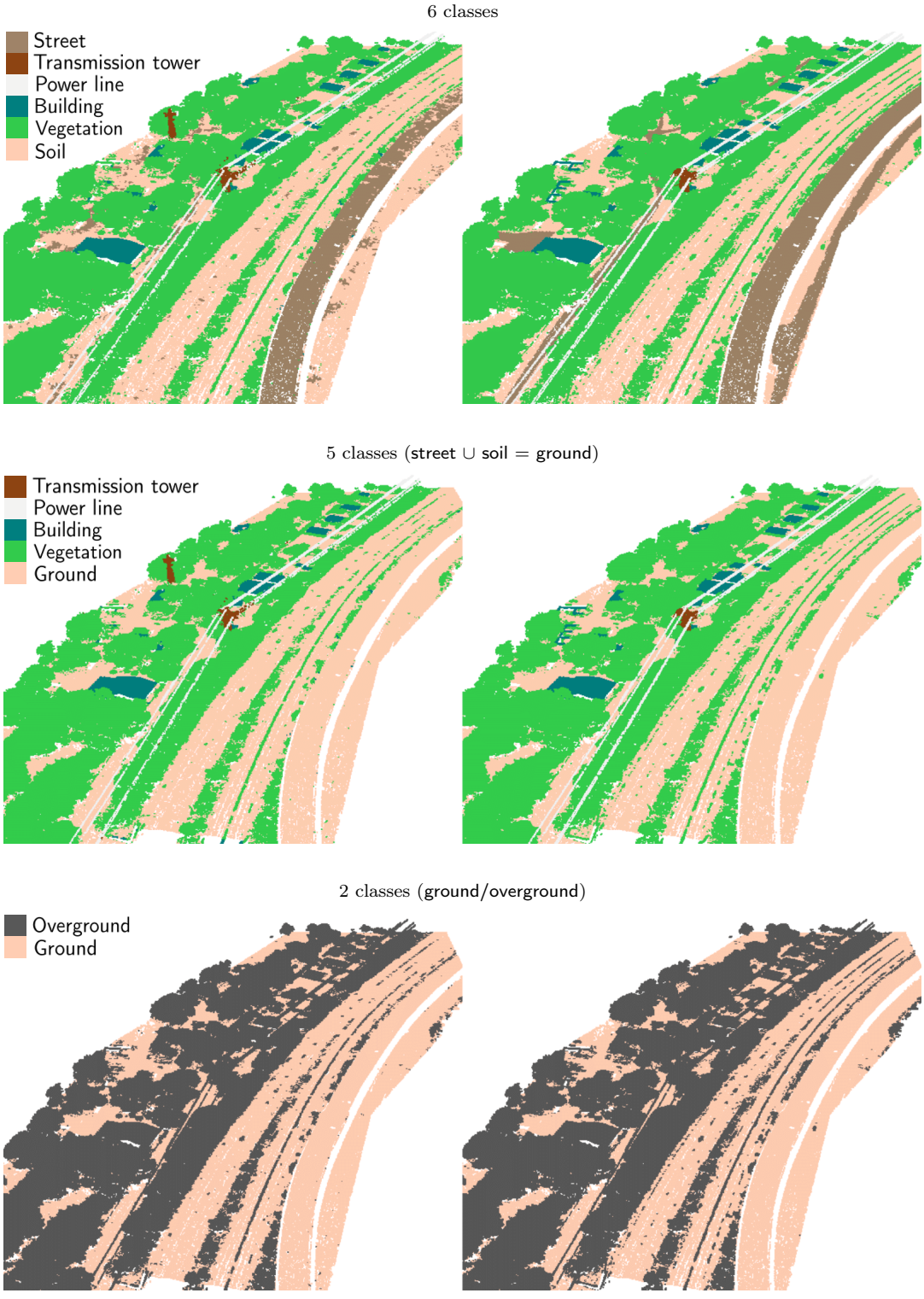

Figure 4. Result obtained from the U-net. Each class is assigned a different color, the predicted point labels are shown on the left and the ground-truth point labels on the right.

References

Guo, L., N. Chehata, C. Mallet, and S. Boukir. 2011. "Relevance of airborne lidar and multispectral image data for urban scene classification using Random Forests." *ISPRS Journal of Photogrammetry and Remote Sensing* 66 (1): 56–66.

Höfle, B., M. Hollaus, and J. Hagenauer. 2012. "Urban vegetation detection using radiometrically calibrated small-footprint full-waveform airborne LiDAR data." *ISPRS Journal of Photogrammetry and Remote Sensing* 67: 134–147.

Jacob, B., S. Kligys, B. Chen, M. Zhu, M. Tang, A. Howard, H. Adam, and D. Kalenichenko. 2018. "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference." In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2704–2713.

Kiritchenko, S., S. Matwin, R. Nock, and A. F. Famili. 2006. "Learning and Evaluation in the Presence of Class Hierarchies: Application to Text Categorization." In *Advances in Artificial Intelligence: 19th Conference of the Canadian Society for Computational Studies of Intelligence, Canadian AI 2006,Québec, Canada, June 7-9, 2006.*, 395–406. Springer.

Mallet, C., F. Bretar, M. Roux, U. Soergel, and C. Heipke. 2011. "Relevance assessment of full-waveform lidar data for urban area classification." *ISPRS Journal of Photogrammetry and Remote sensing* 66 (6): S71–S84.

Maset, E., R. Carniel, and F. Crosilla. 2015. "Unsupervised classification of raw full-waveform airborne lidar data by self organizing maps." In *Image Analysis and Processing—ICIAP 2015: 18th International Conference, Genoa, Italy, September 7-11, 2015, Proceedings, Part I 18*, 62–72. Springer.

Molnar, B., S. Laky, and C. Toth. 2013. "Using Full Waveform Data in Urban Areas." *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 38: 203–208.

Pashaei, M., M. J. Starek, C. L. Glennie, and J. Berryhill. 2021. "Terrestrial lidar data classification based on raw waveform samples versus online waveform attributes." *IEEE Transactions on Geoscience and Remote Sensing* 60: 1–19.

Pashaei, M., M. J. Starek, C. L. Glennie, and J. Berryhill. 2023. "Classification of Terrestrial Lidar Data Directly From Digitized Echo Waveforms." *IEEE Transactions on Geoscience and Remote Sensing* 61: 1–12.

Reitberger, J., P. Krzystek, and U. Stilla. 2009. "Benefit of airborne full waveform lidar for 3D segmentation and classification of single trees." In *ASPRS 2009 Annual Conference*, Vol. 2, 9–13. Citeseer.

Ronneberger, O., P. Fischer, and T. Brox. 2015. "U-Net: Convolutional Networks for Biomedical Image Segmentation." In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015*, 234–241. Springer.

Shinohara, T., H. Xiu, and M. Matsuoka. 2020. "FWNet: semantic segmentation for full-waveform LiDAR data using deep learning." *Sensors* 20 (12): 3568.

Silla, C., and A. Freitas. 2011. "A survey of hierarchical classification across different application domains." *Data Mining and Knowledge Discovery* 22: 31–72.

Wagner, W., M. Hollaus, C. Briese, and V. Ducic. 2008. "3D vegetation mapping using small-footprint full-waveform airborne laser scanners." *International Journal of Remote Sensing* 29 (5): 1433–1452.

Zorzi, S., E. Maset, A. Fusiello, and F. Crosilla. 2019a. "Full-Waveform Airborne LiDAR Data Classification Dataset." `https://uniudamce-my.sharepoint.com/:f:/g/personal/compvis_uniud_it/EshMZziamr5MjSIcqxhiesUB3R0EnYDnQbFdp_ae-K6xvQ?e=oEnxzN`. Accessed: 2021-07-11.

Zorzi, S., E. Maset, A. Fusiello, and F. Crosilla. 2019b. "Full-Waveform Airborne LiDAR Data Classification Using Convolutional Neural Networks." *IEEE Transactions on Geoscience and Remote Sensing* 57 (10): 8255–8261.