



3DMASC: Accessible, explainable 3D point clouds classification. Application to Bi-Spectral Topo-Bathymetric lidar data

Mathilde Letard, Dimitri Lague, Arthur Le Guennec, Sébastien Lefèvre, Baptiste Feldmann, Paul Leroy, Daniel Girardeau-Montaut, Thomas Corpetti

► To cite this version:

Mathilde Letard, Dimitri Lague, Arthur Le Guennec, Sébastien Lefèvre, Baptiste Feldmann, et al.. 3DMASC: Accessible, explainable 3D point clouds classification. Application to Bi-Spectral Topo-Bathymetric lidar data. ISPRS Journal of Photogrammetry and Remote Sensing, 2024, 207, pp.175-197. 10.1016/j.isprsjprs.2023.11.022 . hal-04353070v1

HAL Id: hal-04353070

<https://hal.science/hal-04353070v1>

Submitted on 17 Apr 2023 (v1), last revised 15 Jan 2024 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

3DMASC: ACCESSIBLE, EXPLAINABLE 3D POINT CLOUDS CLASSIFICATION. APPLICATION TO BI-SPECTRAL TOPO-BATHYMETRIC LIDAR DATA.

Mathilde Letard^{1*}, Dimitri Lague^{1,2*}, Arthur Le Guennec^{1,3}, Sébastien Lefevre⁴, Baptiste Feldmann^{1,2}, Paul Leroy^{1,2}, Daniel Girardeau-Montaut⁵ and Thomas Corpetti³

¹ Univ Rennes, Geosciences Rennes, UMR 6118 CNRS, France.

² Univ Rennes, Plateforme LiDAR Topo-Bathymétrie Nantes-Rennes, OSUR, UAR 3343 CNRS, France.

³ LETG UMR 6554, CNRS, F-35000 Rennes, France

⁴ IRISA UMR 6074, Université Bretagne Sud, F-56000 Vannes, France

⁵ Johnson and Johnson

* Correspondence: dimitri.lague@univ-rennes1.fr, mathilde.letard@univ-rennes1.fr

KEYWORDS: bispectral LiDAR, multi-scale classification, multi-cloud classification, feature selection, 3D data, machine learning.

ABSTRACT:

Three-dimensional data have become increasingly present in earth observation over the last decades and, more recently, with the development of accessible 3D sensing technologies. However, many 3D surveys are still underexploited due to the lack of accessible and explainable automatic classification methods. In this work, we introduce explainable machine learning for 3D data classification using Multiple Attributes, Scales, and Clouds under 3DMASC, a new workflow. It handles multiple clouds at once, including or not spectral and multiple returns attributes. Through 3DMASC, we use classical 3D data multi-scale descriptors and new ones based on the spatial variations of geometrical, spectral and height-based features of the local point cloud. We also introduce dual-cloud features, encrypting local spectral and geometrical ratios and differences, which improve the interpretation of multi-cloud surveys. 3DMASC thus offers new possibilities for point cloud classification, namely for the interpretation of bi-spectral lidar data. Here, we experiment on topo-bathymetric lidar data, which are acquired using two lasers at infrared and green wavelengths, and feature two irregular point clouds characterized by different samplings of vegetated and flooded areas, that 3DMASC can harvest. By exploring the contributions of 88 features and 30 scales – including two types of neighborhoods – we identify a core set of features and scales particularly relevant for coastal and riverine scenes description, and give indications on how to build an optimal predictor vector to train 3D data classifiers. Our findings highlight the predominance of lidar return-based attributes over classical features based on dimensionality or eigenvalues, and the significant contribution of spectral information to the detection of more than a dozen of land and sea covers – artificial/vegetated/rocky/bare ground, rocky/sandy seabed, intermediate/high vegetation, buildings, vehicles, power lines. The experimental results show that 3DMASC competes with state-of-the-art methods in terms of classification performances while demanding lower complexity and thus remaining accessible to non-specialist users. Relying on a random forest algorithm, it generalizes and applies quickly to large datasets, and offers the possibility to filter out misclassified points depending on their prediction confidence. Classification accuracies between 91% for complex scene classifications and 98% for lower-level processing are observed, with average prediction confidences above 90% and models relying on less than 2000 samples per class and at most 30 descriptors – including both features and scales. Though dual-cloud features systematically outperform their single cloud equivalents, 3DMASC also performs on single cloud lidar data, or structure from motion point clouds. Our contributions are made available through a self-contained plugin in CloudCompare allowing non-specialist users to create a classifier and apply it, and an opensource labelled dataset of topo-bathymetric data.

1. INTRODUCTION

The introduction of Topo-bathymetric (TB) airborne LiDAR sensors (also called hydrographic LiDAR) to document the land-shallow water continuum in coastal and inland waters goes back less than a decade ago. These sensors aim at bridging the gap between high-resolution narrow aperture NIR airborne topographic LiDAR – generating shot densities above 10 pts/m² – that cannot penetrate water and large aperture bathymetric, airborne LiDAR. Such a bathymetric LiDAR reaches depths over 20 m in clear waters. Still, it has reduced point density (~ 1 pt/m²) and spatial resolution (> 1 m) and implies high mobilization costs that make it unsuitable for topographic surveys. TB LiDAR sensors generally combine a NIR laser ($\lambda=1064$ nm) nearly fully absorbed by water and a green laser ($\lambda=532$ nm) with water penetration that depends on

sensor and flight characteristics, water clarity, and submerged bottom reflectance (Guenther et al., 2000; Philpot, 2019). Associated TB LiDAR datasets are bi-spectral, consisting of one point cloud (PC) per wavelength. Their typical shot densities above 10 pts/m² result in submerged topographies being as detailed as emerging parts (see Figure 1). Shallow submerged parts usually represent a significant source of uncertainty when using only topographic LiDAR data to study land-water interface dynamics (Lague and Feldmann, 2020). Combining high-resolution data about the submerged and emerged surfaces offers new opportunities to map habitats in fluvial (Fernandez-Diaz et al., 2014; Mandlbürger et al., 2015; McKean et al., 2009; Pan et al., 2015) or coastal (Chust et al., 2010; Hansen et al., 2021; Launeau et al., 2018; Parrish et al., 2016; Smeeckaert et al., 2013; Wilson et al., 2019) environments, improve high-resolution modeling of flood

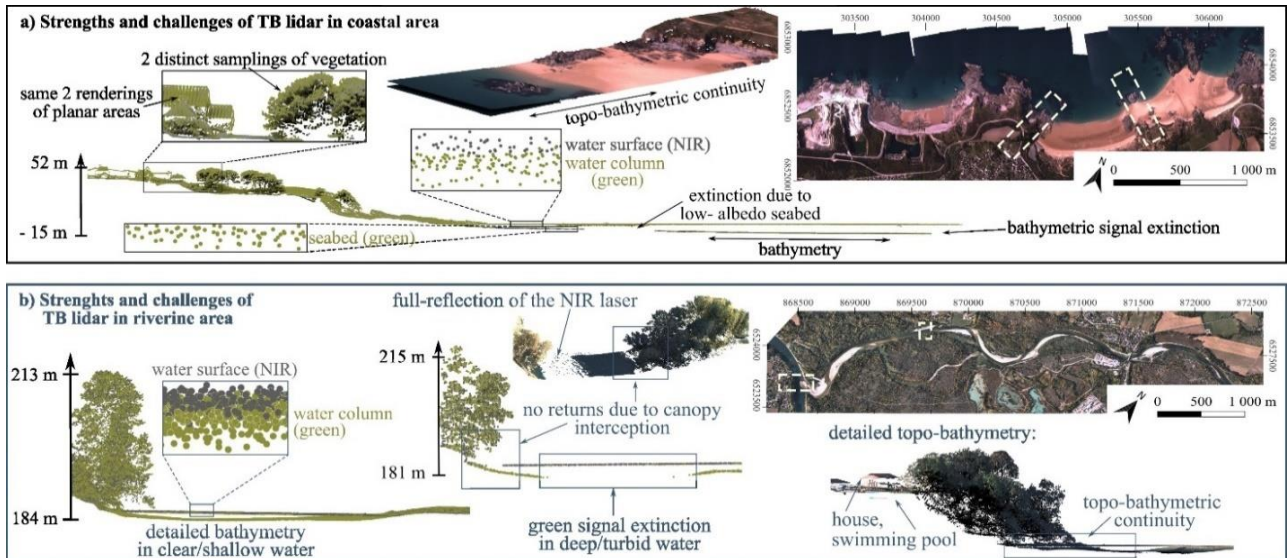


Figure 1: Strengths and challenges of topo-bathymetric lidar data. Examples of (a) the coastal setting of the surroundings of Fréhel (France) and (b) along the Ain river (France). Datasets are presented in the RGF93 coordinates system.

inundation (Lague and Feldmann, 2020; Mandlbürger et al., 2015) or track sediment transport at the land-water interface. Nevertheless, to fully use these datasets and leverage the scientific potential of extensive datasets made of billions of points, automatic classification of green LiDAR data directly at the 3D PC level is essential.

Several methodological challenges complexify the development of adapted classification workflows. First, the refraction of the green laser in water makes it critical to detect all green points below water during data production to subsequently perform accurate refraction correction. This correction requires accurately knowing the spatial extent of water in the scene and the local water elevation, which the NIR channel gives when data are available in the area. While relatively straightforward in coastal environments or large lakes as water will be at a constant elevation, it is far more challenging in fluvial environments for four reasons: (i) water elevation decreases downstream, sometimes abruptly at the vicinity of dams; (ii) rivers can have several active braids or complex hydrological connection with abandoned channels or lakes in adjacent floodplains; (iii) full mirror-like NIR reflection may occur on flat water such that the NIR PC may lack water surface echoes over large areas (fig. 1); (iv) vegetation frequently grows on the floodplain such that river banks and small lakes may be completely below vegetation making things even more complex as canopy interception reduces the backscattered intensity and the likelihood of having a water surface NIR echo and bottom green echo (fig. 1).

Second, the backscattered green laser energy generates two prominent echoes in an ideal clear water column. The first is a volume echo just below the water surface whose position can extend from the water surface to 1 m deep depending on water surface characteristics and clarity (Guenther et al., 2000; Lague and Feldmann, 2020; Philpot, 2019). The second echo corresponds to the bathymetry. The volume echo is of no use but is systematic for any shot. For a given sensor and flight elevation, the maximum measurable water depth highly depends on water clarity and bottom reflectance (Guenther et al., 2000; Lague and Feldmann, 2020; Philpot, 2019). For

instance, in clear coastal waters, the Teledyne Optech Titan sensor can reach depth down to 10-15 m over bright sand but can be limited to 0.5 m over dark rocks and will typically only reach depths of 1-4 m in rivers owing to the reduced water clarity (Lague and Feldmann, 2020). Thus, it is commonplace in inland water surveys that deeper parts of rivers or lakes are locally not detected due to green laser extinction.

Consequently, as for ground detection below a dense vegetation canopy, one cannot assume that a simple operation such as picking the lowest point over a specific area or extracting the last LiDAR recorded echo will systematically isolate the bathymetry. Similarly, because volume echoes can occur up to 1 m below the water surface, removing all green points below this depth is impossible as a large part of very shallow bathymetry will be discarded. There is currently no proposed method to automatically separate bathymetric echoes from volume echoes over large PC datasets in complex inland water environments.

Finally, beyond the detection and separation of bathymetric and volume echoes of the green laser, classifying the nature of the land-water continuum – seabed or riverbed covers and above ground features – on 3D PCs is a significant challenge. Most of the existing approaches rely on 2D rasters such as digital terrain models or digital depth models classified with traditional algorithms like maximum likelihood, support vector machine, or decision trees (Letard et al., 2021; Sun and Shyue, 2017; Tulldahl and Wikström, 2012; Wedding et al., 2008; Zavalas et al., 2014). Although these methods exploit the terrain's geometrical features, they analyze the topography's gridded and average features due to the rasterization step. Rasterization may lead to mixed pixels (Hsieh et al., 2001) and a smoothed-out description of the active sensors' 3D structure of the scene samples. As a result, vertically highly dense data is condensed into equally spaced punctual values, losing the spatial point pattern information. Few studies provide 3D classifications of underwater environments using bathymetric lidar (Hansen et al., 2021; Letard et al., 2022b, 2022a). Additionally, most of them require full-waveform information (Letard et al., 2022b, 2022a), which is complex to process and often unavailable or

unpublished. On the land covers, TB lidar datasets offer potentially new classification opportunities using the bi-spectral backscattered intensity. Research using multispectral lidar to study urban environments already leverages this opportunity by emphasizing the exploitation of the backscattered signal on top of geometrical features evaluated in 2D or 3D (Morsy et al., 2022, 2017a; Teo and Wu, 2017). However, there have been, to date, no applications attempting to classify topo-bathymetric 3D PCs directly.

This work presents an original framework called 3DMASC for 3D point classification with Multiple Attributes, Multiple Scales, and Multiple Clouds and its application to coastal and fluvial TB airborne lidar datasets. 3DMASC combines proven classical elements of single PC semantic classification, such as geometric feature extraction from multi-scale spherical neighborhoods or k-nearest neighbors (Thomas et al., 2018) and a random forest machine learner (Breiman, 2001). In addition, it adds new features specifically engineered to leverage the NIR and green PCs. Our contributions consist in:

- Designing new joint-cloud features calculated on two PCs using their local geometry and backscattered intensity. 3DMASC uses a flexible method to compute features from two PCs, potentially resulting in more than 80 different features;
- Screening systematically over 80 features, both classical and new, to select the essential features and scales contributing to 3D semantic point classification success to develop optimal classifiers in terms of computational efficiency, generalization capability, and interpretability;
- Demonstrating that with limited training data (< 2000 points per class) and less than ten features and five scales, the classification accuracy of TB lidar datasets can be excellent (>0.95);
- Providing a standalone plugin in the open source software Cloudcompare (CC) implementing 3DMASC that non-specialists can use for generic training and classification of any 3D PC (Airborne lidar, Terrestrial lidar, Photogrammetric PCs) and by experts for fast 3D feature computations on PCs;
- Sharing two manually labeled state-of-the-art lidar datasets with two different levels of detail (up to 13 classes).

The paper is organized as follows: the next section introduces the related works on the processing of 3D point clouds; our methodology is then introduced in section 3, and experimental results are shown in section 4, associated with a discussion in section 5.

In this paper, we wish to provide a detailed description of 3DMASC and insist on the explainability of the classifications obtained. Consequently, after presenting the implementation of the workflow, we present typical use cases of 3DMASC in TB environments, give examples of relevant settings, and explain the predictions, their origins, and their reliability.

2. RELATED WORK

In this section, we review methods producing 3D supervised classifications. Clustering methods and approaches relying on

rasterized lidar data are not reviewed. Classification of 3D data is a challenge, as 3D data are unstructured, irregular, and unordered. Their characterization is made harder by the local density variations, and the complex objects they contain. Existing supervised 3D point cloud classification methods can be organized into two categories: handcrafted features with conventional classifiers and learned features with deep neural networks. To the best of our knowledge, no research on the application of 3D neural networks to TB lidar data has yet been published. Existing approaches rely exclusively on handcrafted features extracted on full-waveforms (Launeau et al., 2018; Letard et al., 2022b), rasters (Wedding et al., 2008), or PCs (Hansen et al., 2021).

2.1 Learned features with neural networks

Deep neural networks implement learning mechanisms inspired by neurobiology (Lecun et al., 2015). They rely on successive layers of mathematical operators involving weighting and use error backpropagation to optimize the weights of each operation. Through this process, features of the data progressively stand out. Neural networks thus learn relevant features directly from the data, and eliminate the need to define features and scales upstream contrary to shallow learning approaches.

Within the 3D deep neural networks family, **convolutional** ones currently are among the most performant, namely because they natively consider spatial context between points in their predictions. Their development is recent, as the complexity of 3D convolutions computation hindered their application to PCs for a long time. The first deep learning method on unordered 3D data was PointNet, an architecture relying on multi-layer perceptrons published in 2017 (CharlesR Qi et al., 2017). KPConv, introduced in 2019, was the first 3D CNN developed, through the implementation of a 3D convolution operator (Thomas et al., 2019). Classifications exploiting this operator produce highly accurate results, supported by abstract, multiscale features learned by the networks. Other performant alternatives are developing, relying on graph convolutions and point convolutions, introduced in Pointwise CNN (Hua et al., 2018), SpiderCNN (Xu et al., 2018), or PCNN (Atzmon et al., 2018). They are however designed for mobile lidar data classification in urban environments and much less explored in natural environments, where geometries are less strictly defined.

Since 3D deep neural networks are recent developments, their higher optimization and application complexity remains their main limitation. Very large training datasets in excess of thousands of samples per class are needed, and dedicated GPUs and CPUs are necessary to perform learning and inference.

2.2 Handcrafted features and conventional classifiers

2.2.1 Features definition

Supervised machine learning classifiers require the definition of an input vector to feed to the classifier. They often consist in handcrafted attributes that encode features of the points and its context. The spatial repartition of the points in a PC and, for multiple return lidar, the number, ordering and characteristics of the echoes, depend on a combination of sensor physics and

surface geometry. The reflected intensity is also linked to the albedo of the surveyed object and to the sensor. They thus act as proxies of the actual surface characteristics. PC classifications consequently exploit the **geometry of the PCs** (Hackel et al., 2016), and their spectral dynamics (Chehata et al., 2009) or their **local dimensionality** (Brodu and Lague, 2012; Vandapel et al., 2004). For example, the eigenvectors of each point's neighborhood covariance matrix are popular attributes to identify isolated points, lines, planes, volumes, contours and edges in PCs (Gross and Thoennessen, 2006). Using ratios of these eigenvalues allows to assess the linearity, planarity, sphericity, anisotropy, eigenentropy, omnivariance, scattering or change of curvature of the 3D shape (Chehata et al., 2009; Gross and Thoennessen, 2006; Pauly, 2003; West et al., 2004). The Principal Component Analysis of the points' positions are also used to describe the shape of the local PC, through the analysis of their eigenvalues and the proportion of variance they respectively account for (Brodu and Lague, 2012). Estimates of the local point density (Weinmann et al., 2013) or the verticality (Demantké et al., 2012) of the PC are other helpful parameters to classify points. **Multiples return characteristics** associated to airborne lidar data also constitute information on the objects surveyed: the number of return, return number or ratio of both are useful to identify ground, buildings or vegetation (Chehata et al., 2009). **Height-derived features** such as elevation variations between points of a neighborhood or point distribution kurtosis, skewness are also used for classification purposes (Antonarakis et al., 2008; Chehata et al., 2009; Guan et al., 2012; Yan et al., 2015). Though some studies solely exploit PC geometry to identify 3D objects (West et al., 2004), the **radiometric information** contained in lidar data can further improve PC interpretation where objects have similar geometries (Yan et al., 2015). Radiometric information is rarely used on its own (Song et al., 2002) and often integrated as a complement to previously mentioned geometrical features. It is often among the most contributive features to improve segmentation (Dai et al., 2018) and classification results (Im et al., 2013). The most popular attribute is the mean value of the backscattered intensity over a neighborhood or between first and last returns (Antonarakis et al., 2008). Combining **multispectral radiometric measurements** provides even more reliable information than single wavelength data (Morsy et al., 2017). Using multispectral LiDAR systems allows to incorporate intensity ratios – for example, vegetation indexes – to classification predictors (Chen et al., 2017; Morsy et al., 2017b; Wichmann et al., 2015), or to compare surface reflectances in different optical domains (Chen et al., 2017; Gong et al., 2015) and even create color composites with different channel combinations (Wichmann et al., 2015), thus refining point identification.

2.2.2 Features extraction

Considering that 3D point clouds are unordered and have varying densities, descriptors are computed on the **neighborhood** of each point. The spherical neighborhood is the most common for PC processing, defined by its radius or diameter. Cylindrical neighborhoods are also exploited in Niemeyer et al. (2012), and cubic or cuboid ones are explored in Dong et al., (2017). Overall, spherical neighborhoods are considered the most helpful, based on the observations of Thomas et al. (2018) and Hermosilla et al., (2018), which

compared the use of nearest neighbors (NN) and spherical searches to describe PCs. They are considered more stable than NN to the variations of density (Hermosilla et al., 2018), surface slope or orientation and point pattern that occur in PCs, and more efficient for handcrafted feature extraction (Thomas et al., 2018). Thomas et al. (2019) additionally state that a consistent spherical domain helps classifiers learn more meaningful representations of the local aspect of the PCs during training.

Independently from the type of neighborhood implemented, descriptive features of 3D data can be computed at a **single constant scale** (Chehata et al., 2009) or **multiple scales** (Brodu and Lague, 2012; Hackel et al., 2017, 2016). Multiple scales successively applied to each point have proven to have greater descriptive power than a single constant scale since they can better capture scene elements of different sizes (e.g., vegetation) and the variations of object geometry with scale (Brodu and Lague, 2012; Hackel et al., 2017, 2016; Thomas et al., 2018). Considering the diversity of objects in PCs, the neighborhood type, the number of scales used and their values impact the classification of the data, and thus necessitate careful parameterization. Automatic **optimal scale identification** has been investigated to avoid empiric selection. It mainly relies on minimizing information redundancy – through correlation or entropy estimates – and maximizing relevance in terms of classification accuracy. For single scale classification, Niemeyer et al. (2011) advised an optimal scale of 7 NN in terms of classification accuracy when classifying urban scenes with lidar data. Rather than defining a fixed set of multiple scales, Demantke et al. (2011) try to identify automatically the most relevant scale to describe each point's neighborhood by using its dimensionality. Similarly, Weinmann et al. (2015) select each point's individual optimal scale before extracting and selecting descriptive features. These approaches combine the use of multiple scales across the PC and the computation of features at a single scale for each point. Dong et al. (2017) propose to select an optimal neighborhood type and its scale for each feature rather than optimizing the scale for each point, thus combining the advantages of different types of neighborhoods, multiple scales and uncorrelated features.

2.2.3 Features selection

Similarly to the scales exploited, the optimal feature set should incorporate the most information possible, while also limiting redundancy between attributes. Considering the variety of information derivable from 3D data, empirically selecting the attributes to integrate to a classification is not only time consuming, but also hazardous, as it might impair classification performances. **Feature selection** methods allow to automatize a great part of the feature vector construction. They are mainly based on the estimation of an attribute's relevance relatively to the predicted variables, and on the minimization of correlation between relevant parameters. As explained in Dash and Liu, (1997), feature selection methods can be split into three categories. **Filter-based or univariate methods** aim at maximizing the relevance of the predictors used. They use relevance score functions and rankings of the scores to only keep a subset of the most informative features for classification. Popular score functions include Fisher's index or Information Gain index, but adapted metrics that take

multiple aspects of feature relevance into account also exist (Weinmann et al., 2013). **Multivariate methods** try to minimize feature redundancy among the relevant attributes, often by combining score functions with correlation assessments (Dong et al., 2017; Martin Weinmann et al., 2015). Both univariate and multivariate approaches are independent from the classifier used, and its settings, which is sometimes seen as a generalization advantage (Martin Weinmann et al., 2015), but also do not account for inter feature synergies, and may evict highly correlated but still informative features (Guyon and Elisseeff, 2003). **Wrapper methods and embedded feature selection** consist in exploiting classifier outputs to select features. They either use classification accuracy obtained using each feature separately as a score to prune the input vector (Dong et al., 2017) through backwards or forward selection, or rely on feature importance information provided by algorithms, to evict least important predictors and improve accuracy (Guan et al., 2012). Random forest-based metrics are among the most common embedded selection strategies.

2.2.4 Features classification

Many classification algorithms apply successfully to 3D PCs. The most common ones **classify each point individually**, without considering the relationships between the point's label and its neighbor's assigned labels. They include instance-based techniques such as NN classification, rule-based predictions as applied by decision trees, probabilistic learners like Maximum Likelihood, max-margin learners as Support Vector Machines, and ensemble learning. Ensemble learning is the most popular among individual point classification strategies. It relies on bagging, which consists in assembling several independent weak learners and combine them into a single strong learner, using a voting mechanism. Random Forest (RF) models implement ensemble learning. Their ease of use, efficiency, robustness to overfitting, generalization abilities and production of a feature importance metric (Breiman, 2001; Pal, 2007) explain their frequent use for 3D data classification. They have been used successfully in (Chehata et al., 2009; Hansen et al., 2021; Letard et al., 2022b, 2022a) for point-based classifications of both topographic and TB lidar. In RF, since the decision trees are independent, one cannot compensate the potential weaknesses of another to improve the global performance of the forest. Algorithms like AdaBoost (Hastie et al., 2009) and XGBoost (Chen and Guestrin, 2016) overcome this limitation by incorporating boosting, which consists in training each weak learner to correct its predecessor's errors, however they require more parameters compared to RF.

Individual point classifiers can only consider the spatial context of each point by encrypting it into the feature vector. However, they ignore the fact that neighbor points' labels tend to be linked. Some algorithms thus implement **contextual classification** which involves an estimation of the relationships between 3D points from a neighborhood – often different from the one used for feature extraction – in the training data. They aim at producing spatially consistent classifications of 3D PCs, avoiding the noisy output that individual point classifiers can produce, and thus tend to reach higher accuracies. Examples of such approaches are applications of Associative and Non-associative Markov

Networks, Conditional Random Fields, and Markov Random Fields to 3D data. However, modelling 3D spatial relationships is computationally intensive and thus challenging to apply to large 3D datasets. These approaches also depend on the relationships observable in the training data, which makes exact inference of correlations between labels unattainable.

3. FRAMEWORK/METHODOLOGY

In this section, we describe the 3DMASC method, included in a CloudCompare plugin (Girardeau-Montaut, 2022). Appendix A provides details of the implementation and operation.

3.1 3D features extraction

3DMASC operates directly on unordered sets of points, producing a 3D classification without requiring an intermediate rasterization step. A PC is a set of n 3D points $\{P_k \mid k=1, \dots, n\}$ in which each element P_k is a vector of coordinates (x, y, z) with associated **point based features** : *intensity*, *multi-echo characteristics*, *RGB color* (see Appendix B for a detailed list of features).

On top of point-based features, 3DMASC uses **neighbourhood-based** features defined using a spherical neighbourhood or a k -nearest neighbour search (KNN). A maximum of four different 3D entities are involved in the process of neighbourhood feature extraction:

- **1-2)** two point clouds. The originality of 3DMASC lies in using up to two PCs to characterize the scene of interest. For topo-bathymetric applications, they originate from different wavelengths, typically 532 nm and 1064 nm. We refer to them as PC1 and PC2, respectively.
- **3)** A set of *core points* (Brodu and Lague, 2012), denoted PCX, that 3DMASC classifies at the end of the process. They may be a subset of points from PC1 or PC2 with a regular subsampling or other positions spread within the extent of PC1 and PC2.
- **4)** An optional *context* PC, denoted CTX, containing any relevant context information in its *Classification* attribute at a potentially much lower resolution than PC1 or PC2. A typical CTX would be previously classified ground points at 2 m spatial resolution.

3.1.1 Neighbourhood selection and scales

3DMASC mainly uses a spherical neighborhood search in the relevant PC – PC1 or PC2, depending on the feature to compute – to capture the surroundings of each core point (fig. 2a). The neighbourhood scale is defined as the sphere diameter. 3DMASC use a multi-scale classifier computing multiple neighborhoods for each core point (fig. 2c). The user typically provides minimum and maximum scales and a step (e.g., 1 m) between successive scales. The minimum scale must be consistent with the PC's density to compute features for most core points. The largest scale is typically set by the size of the objects of interests. Defining the optimal set of scales for various types of TB airborne LiDAR (e.g., coastal, fluvial...) is a challenge not yet resolved that we address in this work. Beyond ensuring classification success, it is also crucial for operational efficiency, as the feature computation

time increases strongly with the scale and number of different scales.

3DMASC also supports KNN to measure the vertical or horizontal distance between PC1 and PC2, or CTX (fig. 2b). This supplements relative position measurements between PC's where diameter-based features are impossible to compute due to a lack of neighbors.

3.1.2 Single cloud neighbourhood based features

Single cloud features describe PC1 or PC2 once at a time. Since many criteria characterize a 3D object and can help identify its nature, the plugin natively encompasses 15 different features (see Appendix B for the complete list of features). The broad set of features available is presented in the following paragraphs.

Six dimensionality-based features aim to describe the local PC's general aspect and identify if the object has a linear, planar or spherical outlook (e.g., Brodu and Lague, 2012; Gross and Thoennessen, 2006; Vandapel et al., 2004). They rely on the eigenvalues of principal component analysis (PCA) of the sub-cloud within spherical neighborhoods. 3DMASC can directly use the 3 normalized eigenvalues or classical combinations resulting in sphericity, linearity and planarity metrics.

Six geometry-based features inform on the shape of the PC. 3DMASC computes and use the slope angle, the detrended roughness, the curvature, the anisotropy, the number of points at a given scale and the first-order moment, introduced for contour detection in Hackel et al. (2017).

Three height-based features characterize the vertical structure of the local neighbourhood with respect to the minimum elevation z_{min} and maximum elevation z_{max} . For a core point with elevation z , 3DMASC computes $z_{max}-z$, $z-z_{min}$ and the local thickness of the point cloud $z_{max}-z_{min}$ as explained in (Chehata et al., 2009).

Optional contextual features are used to place each core point in its spatial context and get its position relative to the ground, the water surface, or any specific pre-existing class, labelled in the CTX point cloud. They are computed with a KNN neighbourhood. They generalize the *distance to ground* feature used in Chehata et al. (2009) and Niemeyer et al. (2012).

On top of these classical features, we propose novel 3D descriptors based on the application of **statistical operators on point-based features within spherical neighbourhoods**. Six statistical descriptors can be used: *mean*, *mode*, *median*, *standard deviation*, *range* and *skewness*. They are designed to inform on the multi-scale variations of backscattered **intensity** and **multi-echo LiDAR features** : *return number*, *number of returns* and their ratio, called *echo ratio*. These 4 point-based features combined with the 6 statistical descriptors, results in 24 neighborhood based features at a given scale. To our knowledge, these types of rich multi-scale statistics were never used before for raw 3D PC classification.

Similar features can be built from the 3 components of the **RGB** color information, and we evaluate at a later stage the benefits of this information for classification.

3.1.3 Dual cloud features

Dual cloud features describe the geometrical, spectral, height statistics or multi-echo characteristics differences between the neighborhood of the core point in PC1 and PC2. Spectral ratios have been introduced in the context of multi-spectral LiDAR classification (Chen et al., 2017; Morsy et al., 2017b; Wichmann et al., 2015), but geometrical, height statistics and multi-echo characteristics are new contributions. We designed them to leverage the bi-spectral information and improve the descriptions of scenes characterized by a different 3D aspect in PC1 and PC2. In TB LiDAR datasets, the NIR and green PCs are most significantly distinct above water and vegetation (fig. 1), but they can also be slightly different over other surfaces. This is due to the different surface optical characteristics and the NIR and green laser emitters that can have different angles of incidence or aperture. These may cause differences in the returned signal intensity, and the 3D position of the points. The definition of these features assumes that both PCs are correctly registered and that the alignment error is as low as possible for geometric differences to be related to objects characteristics and not registration errors.

Dual-cloud features consist of **mathematical operations** between single cloud features of the same core point's neighborhood in PC1 and PC2. They can be feature *differences*, *additions*, *multiplications*, or *divisions*. Here we have used *differences* to measure dissimilarity in particular for elevation, geometry and multi-echo features, and *divisions* to normalize one feature by another, typically for intensity. Figure 2a illustrates two examples of dual cloud features: the *mode difference of elevation* that is expected to be close to zero on ground, but different over water; the *median intensity ratio* between the green and NIR channel that is expected to be distinct over different ground. A selection of dominant features are presented, illustrated and explained in the Results section.

Dual cloud features also encompass a distance computation (vertical or horizontal) between the core points PC and another PC (PC1, PC2, or CTX), using KNN.

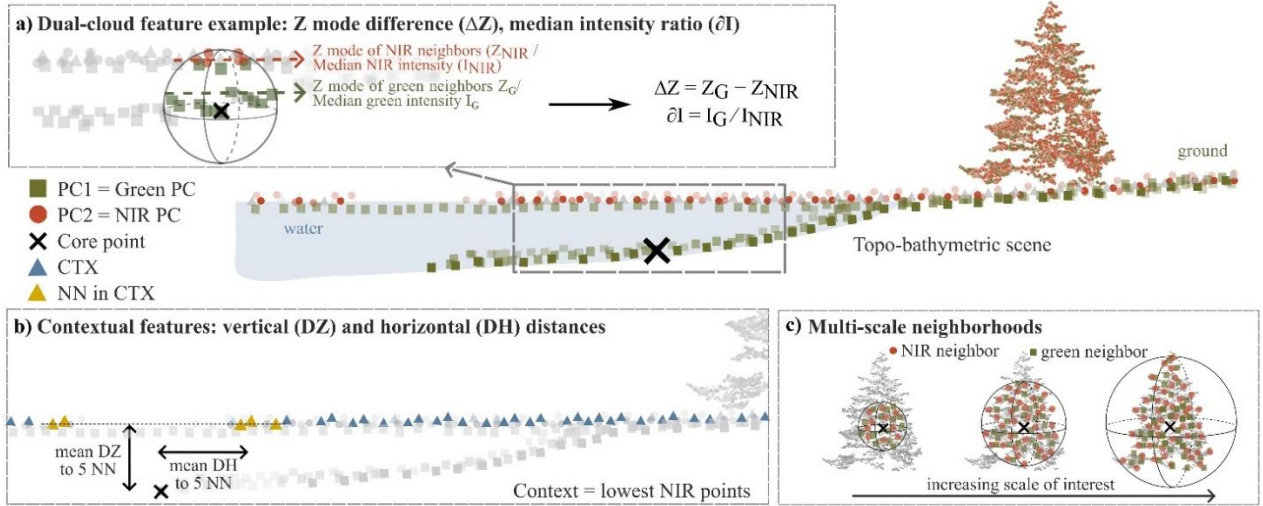


Figure 2: Illustration of the main characteristics of 3DMASC: a) the new dual-cloud features providing a better description of the differences between clouds, b) the generalized contextual attributes placing each point in its spatial setting and c) the multi-scale neighborhoods used to describe the many aspects of 3D objects

3.2 Random Forest Classification

3DMASC uses a Random Forest (RF) algorithm (Breiman, 2001) to perform PC classification, i.e., predict a label $y \in \{1, 2, \dots, c\}$ for each point P_k of the input PC, using the predictor vector $F \{F_{kij} \mid k=1, \dots, n; i=1, \dots, f; j=1, \dots, s\}$, where f =number of features, s =number of scales and n =number of core points and each $F_{kij} \in \mathbb{R}$. For instance, the label can represent the type of object sampled by P .

Here, the feature importance is the product between the probability of reaching a node (i.e., the proportion of samples that get that node) and the Gini impurity decrease of that node. Feature importance is normalized, to sum up to 1. A higher value symbolizes a more significant influence of the feature on the prediction.

RF does not handle NaN values which may be present with our features (depending on the scale, NaN value can occur). This requires specific pre-processing of the predictor vector. Indeed, replacing NaNs with a fixed value may imply irrelevant representations of the local sub-clouds and thus incorporate bias in the classifier training. To tackle this issue, 3DMASC relies on the RF implementation of the cross-platform library OpenCV (Bradski, 2000), which incorporates surrogate splits to handle missing measurements. We use base settings and forests populated with 150 decision trees, having a maximum node depth of 25. We also compared it with the RF implementation in the python library Scikit-learn (Pedregosa et al., 2011), and found similar results.

To further improve the robustness of the classifiers, we also exploit the prediction probability output by RF and use it as a classification confidence indicator, as seen in (Brodu and Lague, 2012) and (Letard et al., 2022b). The prediction probability corresponds to the proportion of forest trees that voted for the class assigned to the point. It ranges between 0 and 1.

3.3 Features and scale pre-selection to control the size of the predictor vector

We propose a feature selection routine (Dash and Liu, 1997) to improve the explainability and the efficiency – through the number of predictors – of the trained algorithm, as there can be almost 90 features per scale in TB environments.

Although information redundancy supposedly does not impact RF performances, it disrupts the explainability of the feature importance values, since if two features bring similar information, their relative importance will be underrepresented. Thus, we keep only a set of uncorrelated features, by using a bivariate feature selection (Dash and Liu, 1997; Guyon and Elisseeff, 2003), incorporating an assessment of the features' Information Gain (IG) (Dash and Liu, 1997) and the Pearson linear correlation coefficient of attribute pairs. The correlation threshold and the scale at which each feature is evaluated are user-defined, and determined after an empirical investigation.

The same bivariate procedure allows the selection of scales. However, we also decided to promote small scales to limit the computation cost of the classifier. The selection process relies on a majority voting procedure. Since it is impossible to consider a scale independently from its application to a feature, we retain the scales that are the most often selected when they are evaluated for each feature independently.

Considering the variety of features included in 3DMASC, removing correlated features and scales does not provide a significantly smaller set of features. Typically, around 40 features per scale of interest remain after correlation filtering. The classifier obtained may thus not be easier to explain, and the training and application steps may be unnecessarily computationally heavy.

To further reduce the dimension of the predictor vector, we considered a feature ranking depending on the IG. However, defining a fixed number of features and scales is highly task- and site-dependent, and filter-based selection would not consider internal synergies between features. Consequently, we use an embedded backward feature selection, relying on the RF feature importance, as detailed in (Aggarwal, 2014; Dash and Liu, 1997). This selection is performed on the uncorrelated set previously obtained. The optimal predictor vector is then

identified through automatic OA monitoring, using a sliding window and keeping the last best iteration before OAs start to drop. In the rest of the paper, we will refer to this step as classifier optimization.

3.4 Framework implementation

Figure 3 sums up the global framework introduced in this work and illustrates how the different steps explained follow each other when processing a PC. As detailed in Appendix A, the Cloudcompare 3DMASC plugin can be used at two level of complexity : for beginners, a complete GUI exists from feature computation to classifier training and class inference; for expert users, 3DMASC can be called through command line solely for fast feature computation with its parallelized C++ implementation, and the results subsequently used in any other environment such as python. Feature and scales preselection, as well as classifier optimization described in section 3.3 follows this latter approach and operate through a complementary python script. To avoid feature preselection and classifier optimization for non-specialist users, a key objective of this work is to identify a minimal set of features and scales that can systematically be used for TB LiDAR classification.

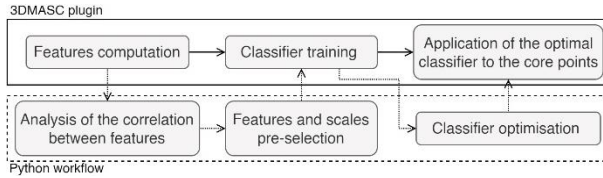


Figure 3: Illustration of the 3D Multi-Attributes, Multi-Scale, Multi-Cloud (3DMASC) classification workflow.

4. DATASETS AND EXPERIMENT PROTOCOL

4.1 Experimental datasets and classes

To illustrate the use of 3DMASC for bispectral lidar data classification, we selected two topo-bathymetric lidar datasets, representing one coastal and one fluvial environment, respectively. These two datasets only differ in the type of environment they model. They were both surveyed with a Teledyne-Optech Titan airborne LiDAR with two wavelengths, 532 nm and 1064 nm (Lague and Feldmann, 2020). The green laser points with a forward pitch of 7°, necessary to avoid strong surface reflection on water and has a beam divergence of 0.7 mrad. The NIR laser has no forward pitch and a beam divergence of 0.3 mrad. Consequently the incidence angle, surface sampling and laser spot size is never the same at a given location of the scene for the two lasers. The sensor produces high-density PCs, typically 36 pts/m² on land – when combining both wavelengths – and 18 pts/m² under water in a single pass (Lague and Feldmann, 2020). More details about the sensor and the acquisition conditions – typical aircraft altitude, speed, overlap between flight lines and preprocessings – are available in (Lague and Feldmann, 2020). The mean vertical offset between the two channels measured on flat horizontal surfaces is typically less than 1 cm. The precision evaluated as the standard deviation of point elevation measured on flat horizontal surfaces is around 5 cm on topography, and 10 cm on submerged surfaces. The first site lies on the French coast of the Channel, in Brittany, near the town of Fréhel; the second is a portion of the Ain river in

South-Eastern France near its confluence with the Rhône river. The surveys were conducted in May 2021 and September 2016, respectively. Figure 1 features the two scenes. They both contain natural and anthropic land covers and include a part of the bathymetric environment: in the first case, shallow sea water with laser extinction at 10.5 m; in the second case, a river with laser extinction at 3.5 m. The flights combined lidar surveys and simultaneous RGB imagery acquisitions with the control camera, which produced orthoimages with ground sampling distances of 25 cm and a registration error of about 20 cm. As RGB imagery acquisition was not the main objective of the surveys, pronounced shadows exist in particular on the Ain survey as it happened late in the afternoon.

4.2 Classes definition and 3D annotation

We evaluate the performances of 3DMASC on two levels of detail: a *primary* classification of 5 land covers – strictly identical for both areas – and *advanced* labeling of 11 and 13 types of objects on the Ain and Fréhel datasets, respectively. We chose the classes, depending on the diversity of land and sea covers we could observe in each area. Table 1 contains all the categories that we use for the primary and advanced classifications. *Artificial ground* includes roads and surfaces covered with concrete or tar (parking lots, dykes). *Vegetated ground* is grass or other low vegetation, such as low-growing heather in moors. In the Ain survey, *intermediate vegetation* is defined as bushes or shrubs with a different aspect than high trees and a smaller growing height. We did not use a classical classification based on a strict height threshold, as usually made in vegetation mapping applications. Our objective was to avoid the traditional misclassification of low branches attached to high trees as shrubs while they are points belonging to high vegetation. The definition of *intermediate vegetation* and *high vegetation* therefore balances 3D aspect and height above ground. Compared to other classes that can be objectively defined, our separation between intermediate and high vegetation is rather subjective. The lack of various types of vegetation in the Fréhel datasets prevented us to refine the vegetation class.

| Primary | Advanced | |
|---------------------|-------------------|-------------------|
| Both | Ain (river) | Fréhel (coast) |
| Ground | Bare ground | Sand |
| | | Pebble/cobble |
| | | Rock |
| | Artificial ground | Artificial ground |
| Vegetation | Vegetated ground | Vegetated ground |
| | Intermediate | Vegetation |
| Artificial elements | High | |
| | Buildings | Buildings |
| | Power lines | Power lines |
| Bathymetry | Vehicles | Vehicles |
| | Water bottom | Underwater sand |
| | | Underwater rock |
| Water | Swimming Pools | / |
| | | Water column |
| | | Surf zone |

Table 1: List of classes defined for the experiments.

We annotated portions of data manually using visual interpretation of the PCs and the RGB imagery acquired simultaneously using Cloudcompare (Girardeau-Montaut, 2022) including new specific developments for quick labelling of 3D point clouds. Four training and validation datasets – one for each level of detail of each scene – were created, all labeled and balanced, for the classification experiments. They all contain 2000 points of each label. To eliminate potential spatial bias due to the use of multi-scale spherical neighborhoods, we forced each training and validation point of the same label to be at least 20 m away, considering we used spheres with diameters up to 15 m. Figure 4 illustrates the resulting sets of points labeled for training and validation. The annotated datasets are available along with the source codes of the plugin and of the scripts used to perform further analysis at the following link https://github.com/p-leroy/lidar_platform..

4.3 Evaluation metrics

We use the **Overall Accuracy** (OA) to quantify the correct proportion of global predictions. The **precision** estimates, for each label, the actual correct proportion of positive predictions. The **recall** value evaluates the part of true positives identified correctly. Precision thus tends to outline over-estimation of some classes, while recall highlights under-estimation. The smaller the difference between both metrics, the better the result. The **F-score** combines the information provided by both precision and recall.

The class-wise performances are explainable with the approach of Lundberg et al. (2017) by computing the **Shapley values** (Shapley, 1952). These range between 0 and 1 and quantify, for each point, the influence of each feature on the label prediction, based on game theory concepts. We performed this analysis using the SHAP Python library (Lundberg et al., 2017). Using these values as a complement to the variable importance measurement and a low number of predictors in the optimized models allows us to have a more robust explanation, less dependent on the randomness of descriptors and samples selection at each node of the decision trees.

5. RESULTS

This section first presents the overall classification results obtained in the fluvial and coastal environments and the impact of feature preselection and optimization. We then present the class-wise results and the dominant scales and features that emerge from the experiment. Finally, we explore the benefits of using RGB information, contextual data and the classification capabilities when using only green LiDAR data. All results presented are obtained on a validation dataset strictly different than the training dataset.

5.1 Overall classification results depending on the number of predictors

We use three different terminologies: Systematic Multi-Scale Classification (SMS) implies the computation of all features at all selected scales; Optimal Multi-Scale Classification (OMS) does not consider all features at all scales; Single Scale Classification uses an identical single scale for all feature calculations. The starting set of features contains 88 features,

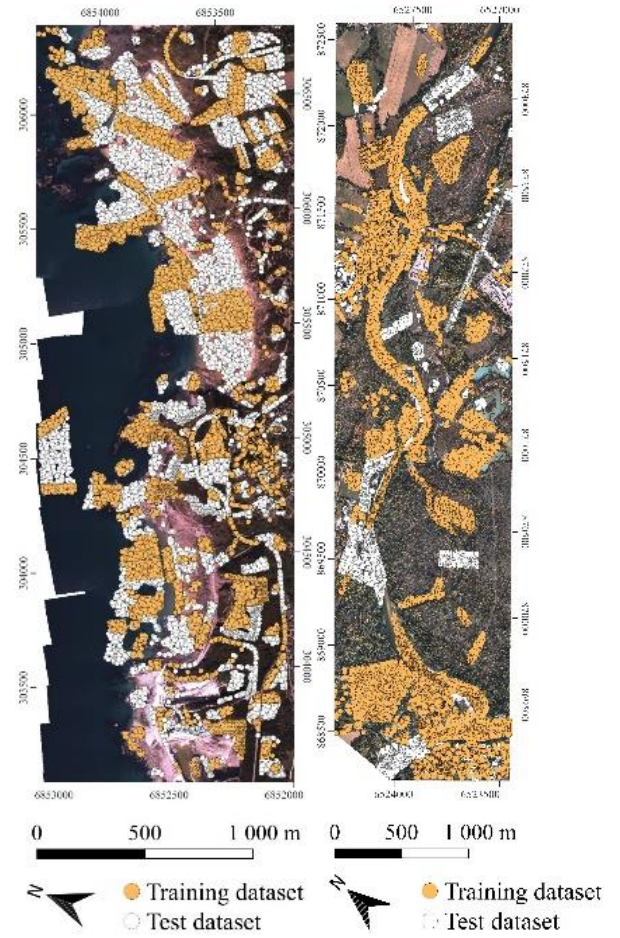


Figure 4: location of the training and validation datasets in both experimental areas; the coast around Fréhel (left) and the surroundings of the Ain river (right) (RGF93).

which include all possible features of 3DMASC computed on PC1, PC2 and their difference or ratio between both PCs (see Appendix B). They are calculated at 29 different scales from 1 m to 15 m with a 0.5 m increment and for KNNs with K in {1;2;3;4;5;10}. The complete predictor has 2011 columns (4 point-based features, 23 features computed at 29 scales for 3 spherical neighborhoods – green, NIR, both – and 6 KNN-based features).

To determine the scale to use for feature evaluation – i.e. IG assessment – we analyzed the OA obtained when selecting features based on their IG at scales varying from 1 m to 12 m. This first analysis shows that features computed at 2 m allow the best selection for OA (see Supplementary Material). Similarly, testing for the optimal correlation threshold results in a value of 0.85 to obtain the highest OA (see Supplementary Material).

5.1.1 Impact of correlation and feature pre-selection

It is expected that the same feature computed at two scales separated with a small gap will produce redundant information. Here we explore if all types of feature exhibit similar level of correlation with scale. Figure 5 presents the mean Pearson coefficient between features computed at scales separated by 1 m or 3 m for different type of single cloud or dual cloud attributes: geometrical, echo-based, and intensity-based. The general tendency is for correlation to increase with

scale and to saturate or increase only slightly above a threshold scale of about 4 to 6 m. The maximum correlation level depends on the type of feature and environment. Dual-cloud geometric features are less correlated than single cloud features. Intensity-based features exhibit high correlation levels suggesting a potentially strong redundancy across scales. The comparison between the dual cloud geometric features at Fréhel for steps of 1 m and 3 m shows that the larger is the step between scales, the lower is the correlation. As expected intuitively, the step between scales should thus tend to increase with scale, in particular above 6 m, to limit information redundancy.

These results indicate that given the high correlation of certain features, especially above 4-6 m, there is hardly a need for a large number of individual scales above this scale, in particular for single cloud intensity and geometric features. Consequently we enforce the maximum number of scales kept in the preselection phase to be 10, compared to the initial 29. Finally, figure 5 demonstrates that intra-feature scale correlation is site-dependant and that there are no clear principles ruling correlation dynamics. Consequently, selecting scales for features based on their correlation is impossible without first computing them.

After feature preselection accounting for correlation, different features are eliminated depending on the site. Overall, there were fewer correlated features on the Ain site and more correlation when using a higher number of classes (and therefore feature samples). The number of features passing the selection step ranges between 36 (Fréhel, primary classification) and 44 (Ain, primary classifier). Height-derived and PCA based attributes were the most pruned types of features during correlation filtering. NIR and green roughness, linearity, planarity, sphericity and return numbers are strongly correlated in both areas. Measures of echo ratio were too much correlated in the Ain, whereas they were not in Fréhel, which reflects the differences between riverine and coastal TB surveys.

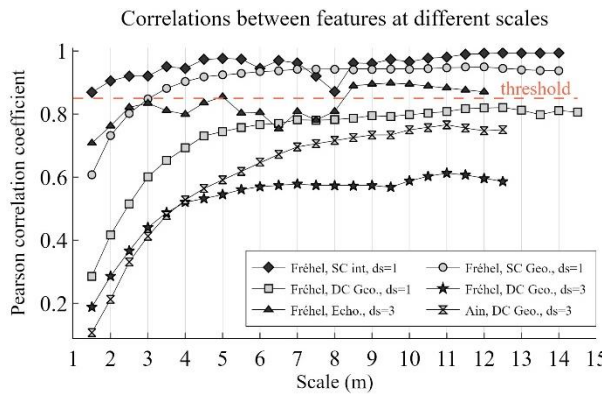


Figure 5: Linear correlation between features computed at scales separated by $dx=1$ m or $dx=3$ m for different examples of features. SC = Single Cloud ; DC = Dual Cloud. The threshold at 0.85 emerge from an empirical analysis.

5.1.2 Impact of predictors number and optimization: from systematic multi-scale to optimal multi-scale classifiers

We explore the influence of the number of features and the number of scales used on the OA and present the results in Figure 6. The results confirm the conclusions of (Brodu and

Lague, 2012; Thomas et al., 2018) on the superiority of multi-scale algorithms compared to single-scale classifiers. This analysis also illustrates the decreasing benefit of increasing the number of features and scales past 20 features and 6 scales, even using uncorrelated entities only. Figure 6 highlights that adding features increases OA more than adding scales. For instance, adding a second scale to a single feature classifier systematically results in an OA surge, while harvesting 10 features and two scales produces more accurate results than relying on two features and 10 scales. Due to the majority voting used for scale selection, the scale used for single-scale classification varies, explaining the variations of accuracy (see Figure 6, single-scale curve), and showing the dependence between the features' relevance and their computation scales.

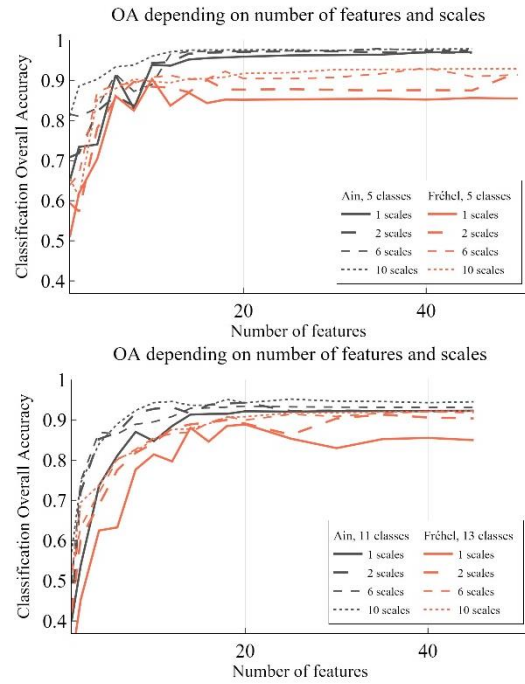


Figure 6: Classification performances depending on the number of descriptive features used for different numbers of scales of computation used.

Since, the accuracies presented are the results of applying the trained algorithms to data unseen during training, these results also showcase the stability of RF relative to overfitting and generalization. Even when training the model with hundreds of predictors, OAs remain stable (between 90% and 97% depending on the use case) when classifying the distinct validation points. Furthermore, 3DMASC's features succeed at characterizing the nature of the objects lying behind the points, as accuracies converge towards values ranging between 90% (Fréhel, advanced) and 97% (Ain, primary). It is, however, delicate to determine the ideal number of features and scales to retain. The optimization procedure provides more information on the required number of predictors to achieve high-accuracy identification of the different classes. Figure 7 presents the OA dynamics when reducing the predictor set iteratively. Using automatic monitoring of the OA's significant variations, we solve the optimization problem for our four classifiers. Table 2 gathers the main characteristics of the optimal multi-scale classifications.

| Classifier | Ain (5 cl) | Ain (11 cl) | Fréhel (5 cl) | Fréhel (13 cl) |
|------------------------|---------------|----------------|------------------|-------------------|
| SMS OA | 97.9% | 94.6% | 92.8% | 91.9% |
| SMS nb of pred. | 371 | 352 | 315 | 330 |
| OMS OA | 97.7% | 93.1% | 91.4% | 90.7% |
| OMS nb of pred. | 14 | 17 | 23 | 30 |
| Features | 11 | 13 | 12 | 18 |
| Scales | 5 | 6 | 6 | 6 |
| Confidence | 0.935 | 0.893 | 0.885 | 0.83 |
| F-score | 0.977 | 0.93 | 0.913 | 0.907 |
| Precision | 0.977 | 0.933 | 0.915 | 0.915 |
| Recall | 0.977 | 0.932 | 0.914 | 0.907 |

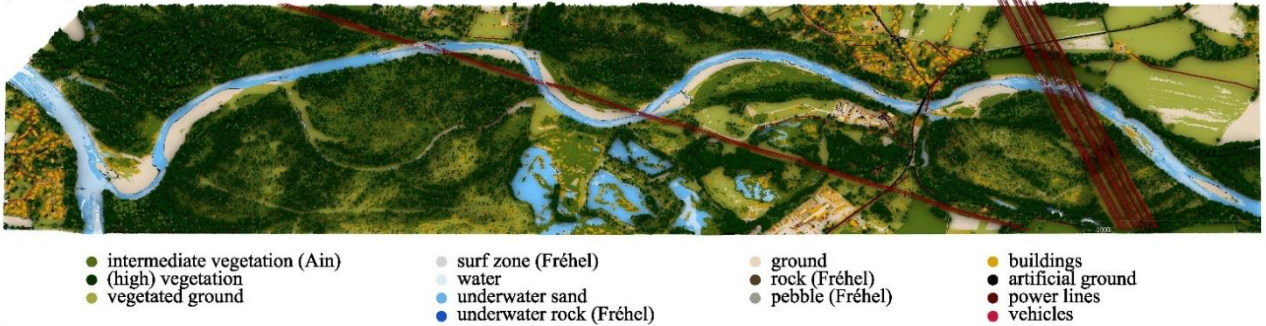
Table 2: Classification metrics for the four models. SMS = systematic multi-scale, OMS = optimized multi-scale. Confidence, F-score, Precision, and Recall are average values for all labels. Nb of pred. refers to the number of predictors

The results in Table 2 and Figure 7 confirm what we observed in Figure 6: a small number of features and scales produces highly accurate classifications. The most complex classifier obtained incorporates 30 predictors, including 18 features and six different scales. Table 2 also outlines that more predictors are needed to correctly identify a larger number of labels: advanced Fréhel classification requires seven more predictors than primary. The optimized models obtain accuracies ranging between 90.7% and 97.7% and harvest more features than scales, confirming the superior efficiency of feature diversity over scale abundance. Overall, the maximal difference in OA between systematic multi-scale and optimal multi-scale classifiers is 1.5%. Models are highly simplified: on average, the optimization reduces the predictor vector's dimension by 94%. However, the fully iterative procedure is necessary: keeping only the first features in importance at the first iteration does not provide good results. For example, when using the 14 highest-ranked predictors at the first RF classification of the Ain, the OA is only 22.9%.

5.2 Class-wise results with optimal multi-scale classifier

5.2.1 Class-wise metrics

a) Ain (11 classes)



b) Fréhel (13 classes)

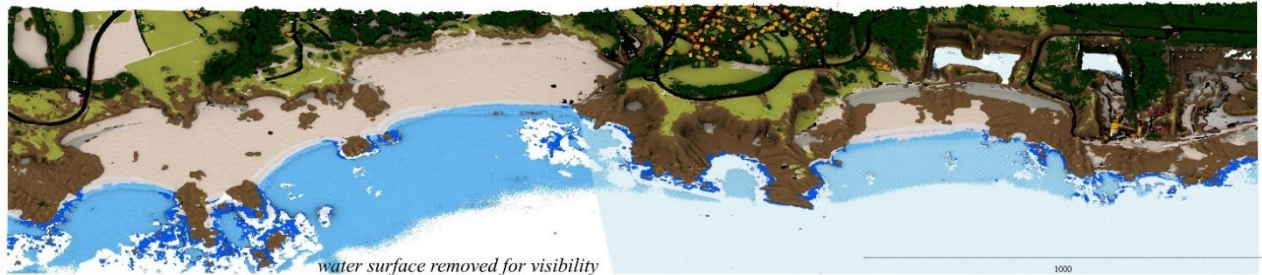


Figure 8: Classified point clouds of both areas, with Fréhel on top and the Ain under.

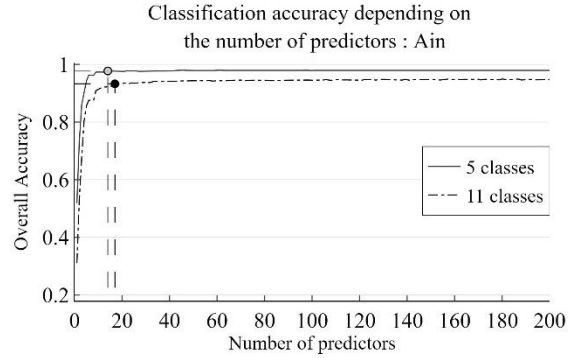
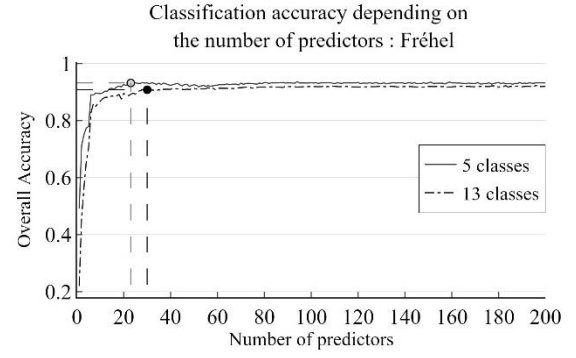


Figure 7: Overall accuracy depending on the number of predictors used during classifier optimization.

Figure 8 illustrates the application of the optimized classifiers for the *advanced* classification. The land-water transition is well identified and the main elements such as ground and above ground features are separated. Figure 9 sums up the class-wise results obtained for each experiment. The main classes of the Ain site obtain F-scores higher than 90%. On the coastal area, they are distinguished with F-scores over 85%.

The difficulty imposed by the distinction of objects with similar geometries does not impact the performances severely. All F1-scores are higher than 80%, and average confidences are over 83% and 88% in primary and advanced cases,

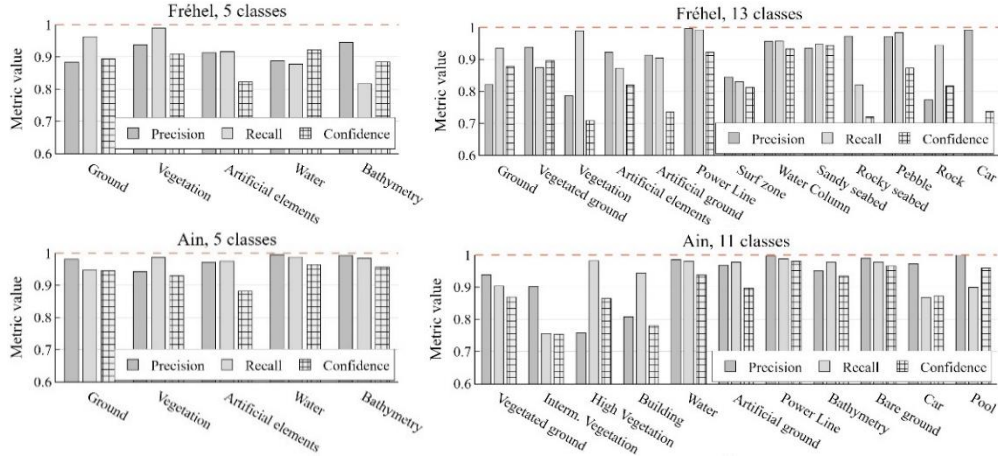


Figure 9: Precision, recall, and prediction confidence per class for the four classifiers after optimization.

respectively. These observations suggest efficient construction of the classifiers, as correct predictions obtain the vote of most of the decisions trees.

The identification of *water* is highly accurate (99%) in the Ain, but there is confusion in the more diverse bathymetric environment of Fréhel, namely between *bathymetry* and *water*. The *surf zone* is also challenging to distinguish from *ground* or *rocky seabed* in some areas.

Some classes show gaps between precision and recall, reflecting the over-detection of *buildings* in rocky areas or of *intermediate vegetation* in the Ain (see Figure 8).

5.2.2 Dominant scales analysis

The optimized predictor vectors indicate that some features are particularly informative at specific scales, and conversely, some scales are essential for given features only. The optimization phase alters the systematic multi-scale character of the classification since the number of predictors in the optimized models is smaller than the product between the number of scales and the number of features. For example, the advanced classification of the Ain site has an optimal multi-scale predictor vector exploiting 13 features at six scales, yet, its total size is 17. In contrast, if the optimal classification were a systematic multi-scale model, it would be 78.

Table 3 sums up the specific scales retained for each experiment. It shows that finer scales are necessary to describe the Ain site: the minimal scale selected is 1.5 m, whereas it is double for Fréhel. All classifiers follow a similar pattern: they exploit small to medium scales up to about 6 m, and a much larger scale of about 14 m without transitioning via a medium value. The advanced models both reuse similar scales to their primary equivalents but incorporate new ones in between, reducing the typical sampling step of object sizes. However, the 11 labels classifier of the Ain is the only one discarding the 14 m scale, thus exploiting only small to medium diameters.

| Classifier | Optimal scales |
|------------------|----------------------------------|
| Ain, primary | 1.5 – 4 – 5.5 – 14 – 10NN |
| Ain, advanced | 2.5 – 3.5 – 4 – 5.5 – 7.5 – 5NN |
| Fréhel, primary | 3 – 3.5 – 5.5 – 6 – 14.5 – 1NN |
| Fréhel, advanced | 3.5 – 4.5 – 6.5 – 7 – 14.5 – 1NN |

Table 3: Remaining scales in the four optimized multi-scales classifiers. kNN indicates k nearest neighbors.

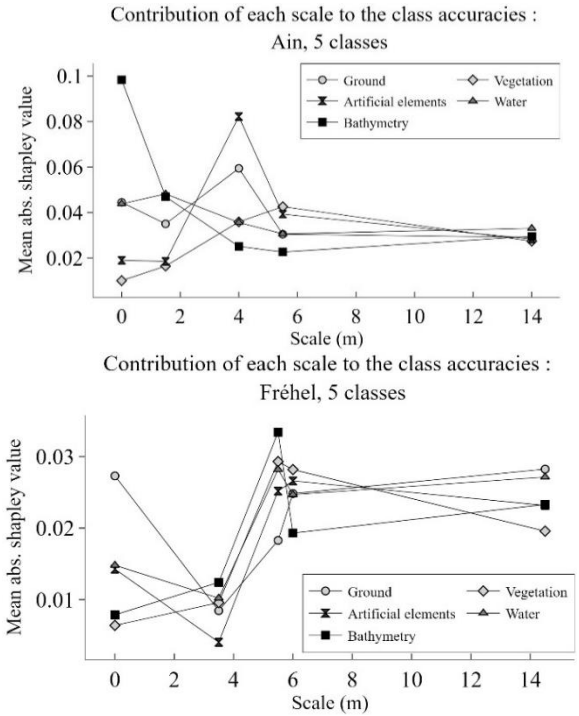


Figure 10: Mean absolute Shapley value obtained by each scale of the optimized predictor vector depending on the class considered (a scale of 0 m represents features computed with a kNN search).

To better identify the contribution of specific scale to various classes in the two environments, Figure 10 shows the Shapley analyses for the standard classification. Dominant scales are drastically different between the Ain area and Fréhel. Water and bathymetry are dominated by features computed with around 6 m and 14.5 m diameter in Fréhel, whereas 1.5 m and 10NN features are more useful in the Ain. Similarly, artificial elements and trees do not exploit the same sphere sizes over the two sites. The scales also adapt to each label. For example, artificial elements – containing buildings, vehicles, and power lines – rely less on KNN features than bathymetry in the Ain or ground in Fréhel.

We can also identify two groups of classes having similar scale contribution patterns. The first includes *water* and *bathymetry*, and the second includes *ground* and *artificial elements*.



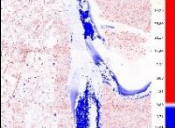
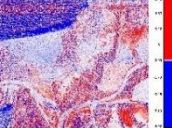
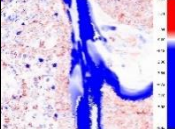
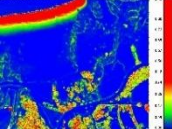
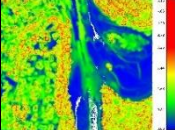
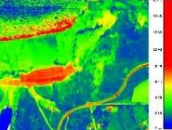
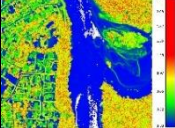
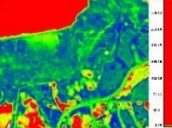
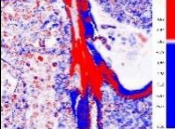
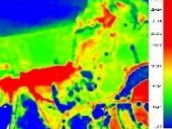
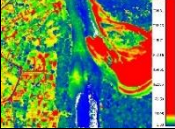
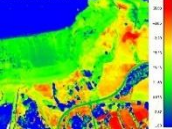
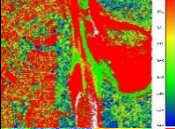
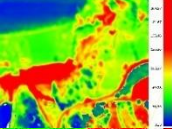
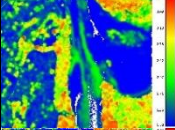
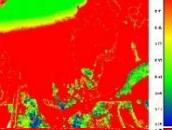
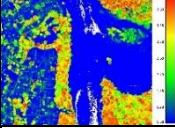
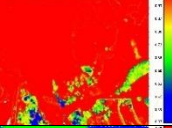
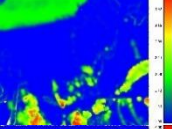
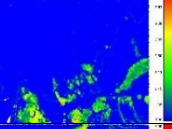
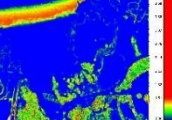
| AIN | | FREHEL | |
|---------------------------------|---|-----------------------------|---|
| RGB Image |  | RGB Image |  |
| Z difference with NIR kNN |  | Z difference with NIR kNN |  |
| Difference of Z modes |  | Sphericity (green) |  |
| Standard deviation of Z (green) |  | Ratio of median intensities |  |
| Standard deviation of Z (NIR) |  | Skewness of intensity (NIR) |  |
| Difference of roughness |  | Mode of intensity (green) |  |
| Mean intensity (green) |  | Mode of intensity (NIR) |  |
| Mean EchoRatio (green) |  | Mean intensity (green) |  |
| Mean Nb of Returns (green) |  | Mean EchoRatio (green) |  |
| Mean Nb of Returns (NIR) |  | Mean EchoRatio (NIR) |  |
| | | Mean Nb of Returns (green) |  |
| | | Mean Nb of Returns (NIR) |  |
| | | Mean Return Nb (green) |  |

Table 4: Optimal features for both sites.

Vegetation of the Ain study site does not follow a similar trend to any other classes, but in Fréhel, its dynamics are comparable to those of *bathymetry* and *water*.

5.2.3 Dominant features analysis

To simplify, we only review dominant features of the primary classifications in this section. Several features stood out from the rest and passed both selection and optimization phases. They theoretically contain the essential information to distinguish the defined classes. Table 4 introduces and illustrates each of them.

The sets of optimal predictors obtained, presented in Figure 11 and Table 4, seem to be tailored to each site. The Shapley analysis, in Figure 11, corroborates this observation. **Only five features common to both sites are identifiable:** *vertical distance of green points to their NIR neighbors (KNN), mean green intensity, mean echo ratio in green neighborhood, mean number of returns in the green neighborhood, and mean number of returns in NIR neighborhood.*

Two **groups of labels have similar feature contribution patterns.** *Ground, bathymetry, and water* on one side and *vegetation and artificial elements* on the other. The first group is mainly identified by multi-echo features and NIR intensity. The second relies primarily on dual cloud features – median intensity differences and distances between points of the two wavelengths – and NIR multi-echo attributes.

In both cases, **the TB aspect of the datasets is fully exploited:** for the Ain, there are as many green PC features as NIR PC features, and for Fréhel, the optimized set includes 4 and 6 features of the NIR and green PCs respectively. NIR PC-derived features are more contributive to topographic objects, while both PCs are equally crucial for ground/seabed/water distinction. The experiments on Fréhel also draw more on NIR intensity-derived parameters than the models to process the Ain, in which only one green spectral parameter is involved with low relative importance (Figure 11). The class-wise feature importance analysis also shows that **features do not have the same descriptive power in both NIR and green domains.** The number of returns of the NIR echoes is more informative on the nature of the surface than its green equivalent.

Both results show a **predominance of newly introduced 3DMASC features over classical features** used in other studies (Chehata et al., 2009; Hackel et al., 2016; Thomas et al., 2018; M. Weinmann et al., 2015). 8 out of 11 for the Ain site and 10 out of 12 for Fréhel are attributes we propose with 3DMASC: means, modes, or skewness values of PC characteristics. Geometrical and dimensionality-based features are scarce: only NIR PC roughness, NIR PC dip, and green PC sphericity pass the optimization phase. The mean green intensity is the only other example of classical feature observable (see Table 4). Intensity-based features constitute nearly half of the predictors of the Fréhel optimized classification, but are few in the Ain model. The other half of the Fréhel predictor set is dominated by multi-echo features of both wavelengths, that evict height-based features and geometrical features. In the Ain, they appear through the differences in elevation modes. Other dual cloud features stand out: *vertical distances between green and NIR points*, *elevation mode differences*, and *median intensity differences between the two PCs* (Table 4).

Figure 11 also reveals that the **new 3DMASC features outperform usually dominant characteristics** like intensity. The difference in elevation modes between NIR and green PCs, is more relevant to identify vegetation than intensity. Similarly, the roughness difference between PCs systematically dominates single cloud NIR roughness, even for ground and over-ground objects separation. The ratio of median NIR and green intensities is particularly useful to outline vegetation and artificial elements. Dual cloud features are present in both OMS classifiers, illustrating how they complement separate single cloud attributes. Multi-echo features also contribute significantly to the predictions. The

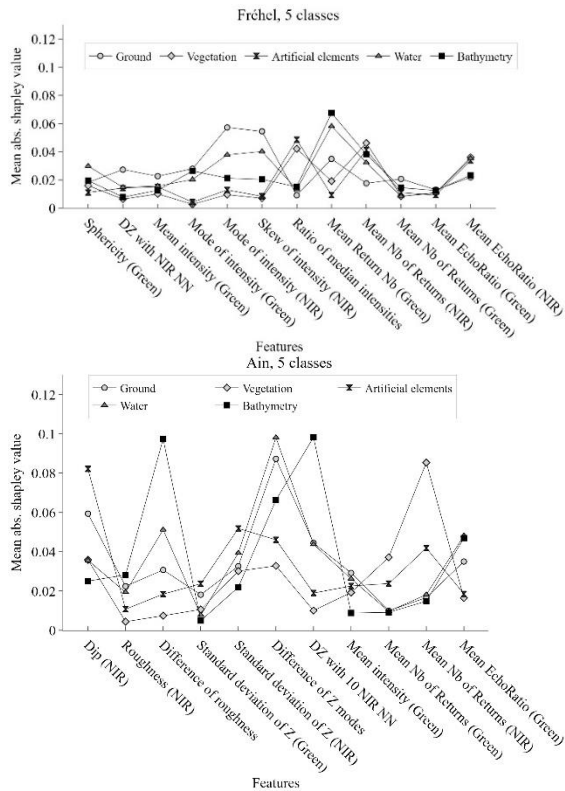


Figure 11: Mean absolute Shapley value obtained by each feature of the optimized predictor vector depending on the class considered

mean echo ratio helps identify *ground*, *water*, and *bathymetry*. In contrast, the mean number of returns characterizes *vegetation* and *artificial elements*.

5.3 Results using other predictors

In this section, we test 3DMASC in different settings: using a context PC, RGB information, and simulating the unavailability of the NIR wavelength. All results are summed up in Figure 12. They are obtained by running the complete framework on initial predictor vectors including contextual features, RGB-derived features, or green features only. The presence or absence in the optimal predictor set of each tested attribute is thus already an indication of their informative character.

The contextual features used were vertical distances to a PC containing only ground or water surface points, for different scales (1, 3, 5, and 10 NN). These predictors allowed to use smaller scales (see Supplementary Materials) and improved the prediction confidence and quality of almost each class of the Ain, except *bathymetry* and *artificial elements*. In Fréhel, they mostly improved the accuracy of *artificial elements*, but tend to penalize *water* and *vegetation*.

The reflectance in the blue domain is the only RGB derived attribute that passed optimization. Its mode is used in two models: Fréhel primary, and Fréhel advanced. This shows that RGB features are not crucial to detect the classes of the Ain but may serve to differentiate coastal land and sea covers. They also seem to penalize our classifier optimization framework when they are used but do not appear in the best models, as the losses in F-scores on the Ain reveal. This show that RGB parameters may evict somme more useful features.

When using green laser data only, OAs range between 85% (Fréhel, advanced) and 94% (Ain, primary). Predictors vectors are dominated by multi-echo features and intensity-derived attributes. In Fréhel, dip and standard deviation of intensity are

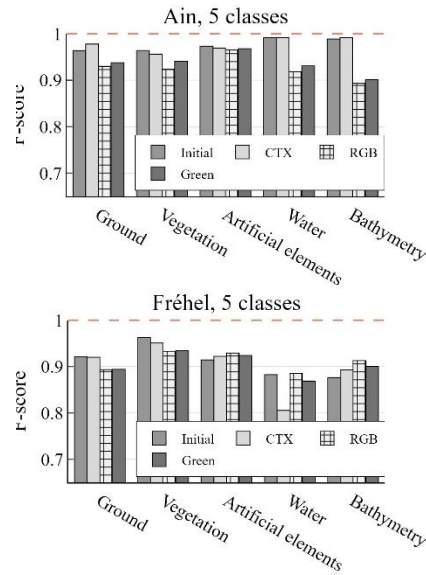


Figure 12: F-score obtained for each class depending on the experiment. Initial = optimal classifier obtained with the initial set of predictors. CTX = optimal classifier obtained when adding contextual features. RGB = optimal classifier obtained with RGB features added. Green = optimal classifier obtained using only green features

the only new features selected. In the Ain, point-based echo ratio, mean return number, proportion of the third PCA eigenvalue and mode and standard deviation of intensity appear. Overall, more scales are used per features and seven and eleven features are selected for Fréhel and the Ain, respectively. The performance decline observed mostly affects *water* and *bathymetry* in the Ain. Although their F-scores drop by 6% and 8% respectively, they remain higher than 90%, showing that a single bathymetric PCs already provides accurate detection of the water column and the riverbed. In Fréhel, the classification of *bathymetry* is even improved when excluding NIR data. In both settings, the distinction of topographic classes is less accurate when discarding NIR information.

6. DISCUSSION

Starting with a set of 88 features computed at 29 scales, we obtained optimized, compact classifiers exploiting at most 30 predictors – scales and attributes – and resulting in good to excellent classification for up to 14 classes. In this section, we discuss these results with respect to existing work on PC classification.

6.1 Classifier optimization and number of predictors used

Through 3DMASC, we obtain classifications of TB scenes with OAs over 90%, using light classifiers that harvest a maximum of 30 predictors (Table 2). Average prediction confidence is high, and accompanied by high accuracy, synonym of an efficient classifier training. Low values of confidence can be linked with classification errors, and used to filter out misclassified points. Table 5 shows the results of applying a confidence threshold below which points are removed. It illustrates that there is necessarily a balance to find between result quality and spatial resolution of the classified PC, as aiming at fewer classification errors means accepting to affect the local density of the data.

The optimization step seems to efficiently balance computational efficiency and high-quality classifications. The low number of predictors makes the models applicable to large datasets, easily explainable with Shapley values and thus accessible to non specialist users. These characteristics allow 3DMASC to be an interesting alternative to current state-of-the-art methods that are 3D deep neural networks. Although they have not yet been applied to similar problems, their use on benchmark datasets demonstrate similar performances in terms of accuracy than those we obtain (Charles R. Qi et al., 2017; Thomas et al., 2019). However, deep neural networks hyperparameters are

| CLASSIFIER | Confidence threshold | OA | Remaining points (%) |
|--------------------|----------------------|-----|----------------------|
| AIN advanced | 0.5 | 95% | 96% |
| | 0.6 | 97% | 92% |
| | 0.7 | 98% | 87% |
| | 0.8 | 98% | 80% |
| FREHEL advanced | 0.5 | 94% | 92% |
| | 0.6 | 96% | 84% |
| | 0.7 | 97% | 76% |
| | 0.8 | 97% | 67% |

Table 5: Overall Accuracy depending on the confidence threshold applied to filter the predictions.

harder to optimize without expert knowledge, and require much more intensive training in terms of labelled data and computing power, while our approach relies on fewer samples. We chose to experiment on datasets containing 2000 labelled points per class, but when randomly subsampling the labelled data, we observe that high accuracies are already possible with a few hundreds of ground truth points per class, as featured in Table 6. Neural networks are also more abstract and thus harder to decipher, contrary to 3DMASC thanks to feature importance and Shapley values.

| Samples per class | Overall accuracy | | | |
|-------------------|------------------|--------|--------|--------|
| | AIN | | FREHEL | |
| | 5 cl. | 11 cl. | 5 cl. | 13 cl. |
| 1600 | 98% | 95% | 91% | 91% |
| 1200 | 98% | 95% | 91% | 91% |
| 800 | 97% | 95% | 91% | 90% |
| 400 | 96% | 95% | 90% | 90% |
| 100 | 94% | 93% | 89% | 90% |

Table 6: Classification accuracy depending on the number of training samples used. Tests are performed using the complete set of 3DMASC features.

6.2 Dominant scales

Taking advantage of the explainability of the method, we identify typical characteristics of OMS classifications. First, **a typical set of scales emerges from the experiments**, including small and medium sphere diameters ranging between 1.5 m and 6 m and one larger scale around 14 m (Table 3). The global range of scales selected does not vary between primary and advanced classifiers, except for the Ain where we can expect that the introduction of smaller scale objects - *vehicles*, *swimming pools*, *intermediate vegetation* – penalizes very large scales. Advanced classifiers rather add scales within the core range, reducing the step between two options. Second, **the exact optimal scales that arise are specific to each environment**, which questions the possibility to identify optimal neighborhoods without analyzing their application context. For example, out of four experiments, three different optimal NN neighborhoods stand out: 10, five, and one (see Table 3), contrasting with the conclusions of Niemeyer et al. (2011) that select an optimal scale of seven NN for their different experiments, and with the results of Dong et al. (2017) who find that five NN are the most often selected neighborhood. Furthermore, the fact that each selected scale is not used for each feature tends to be consistent with the work of Dong et al. (2017), choosing to optimize each feature’s neighborhood rather than identifying a global optimal scale. Third, **scales selection results are consistent with intra-feature correlations** we observed in Figure 5. These suggested that less scales were needed above 6 m than below, which is in line with the fact that we only obtain one large scale. This large scale also outlines the necessary **trade off between classification accuracy and classification resolution**. If we investigate the role of this much larger scale, we find that, though it helps to mitigate some errors linked to larger scale roughness in the PCs – for example confusion of rocks with buildings – it also smoothes out the results, blurring classes borders and even missing smaller objects like cars. In Figure 13, cars can be identified in the PC, but many of them are missed and labelled as ground when large scales are used. Limiting the range of scales to 7 m produces a result in which

these cars are correctly detected, but the ground incorporates false *building* labels.

| Classifier | Confidence threshold | OA | Remaining points (%) |
|--|----------------------|-----|----------------------|
| FREHEL advanced (Max scale = 7 m) | 0.5 | 94% | 91% |
| | 0.6 | 96% | 84% |
| | 0.7 | 97% | 76% |
| | 0.8 | 98% | 66% |

Table 7: Overall Accuracy depending on the confidence threshold for a reduced set of possible scales

Our observations thus **question the relevance of large scales**, which appear to be selected for certain point types as they pass the score filtering selection, but end up penalizing the global classifier application through several aspects. Table 7 illustrates the confidence filtering analysis obtained on the Fréhel advanced classifier optimized on scales within 1 to 7 m only.

It shows that, without the possibility to select larger scales, the classification reaches similar accuracies and confidences. However, they clearly affect the computation efficiency. The advanced Fréhel OMS classifier obtained on a scales up to 7 m incorporates ten more features, but the computation time is divided by three (3450 points per second versus 1102 points per second). Suppressing large scales may thus improve classification speed, while maintaining high OAs.

6.3 Computation time

| | 1 m | 4 m | 7 m | 10 m |
|--------------------------------------|----------------|------------------|-------------------|------|
| Single scale (s/1000pts) | 28082 | 7337 | 2109 | 928 |
| | 1 – 4 m | 1 – 7 m | 1 – 10 m | |
| Multi-scale (pts/s) | 4069 | 854 | 275 | |
| | | Up to 7 m | Up to 15 m | |
| Optimised multi-scale (pts/s) | | 3450 | 1102 | |

Table 8: Time necessary to compute features at different scale sets.

Computational efficiency is an important aspect of 3DMASC. The computation of the spherical neighborhoods is the main bottleneck of the workflow, similarly to what is observed in other studies (Hackel et al., 2016; M. Weinmann et al., 2015) and sometimes even drives the choice of the neighborhood type. Table 8 illustrates the time necessary to compute all implemented features at different single or combined scales. It shows how crucial scale selection is: without optimization, computing scales from 1 m to 10 m is five times slower than between 1 and 6 m. Considering the observations made in section 6.2 about the performance differences between models using medium and large scales, the choice of the scales range may be significant on the practical deployment of a classifier.

We ran these test computations on a computer equipped with a 24 core CPU and 128 Go of memory. The test file contained 106 410 018 green points, 61 043 388 NIR points, and 5 700 844 core points. The computation can last up to 5 hours and 45 minutes when exploiting ten scales reaching up to 10 m, which corresponds to a computation speed of 275 points per second. This computation speed could be increased by implementing pyramidal computation into the 3DMASC plugin, which consists in subsampling the data when increasing the neighborhood size as made in Thomas et al.,

(2018). Our implementation currently does not rely on such processes but rather on octrees, future development could thus improve this aspect. Additionally to the selection of a scale range, the number of different diameters within the interval, and the number of features to compute for each neighborhood also has an impact – though less significant – on the processing time. Table 8 shows that the optimised descriptor set relying on scales up to 7 m is three times faster to compute than the complete set of features on scales between 1 and 7 m. Consequently, although predictor selection is not crucial for classification performance (see Table 3), it is essential to the practical applicability of the method.

6.3 Class-wise results: dominant features

Out of the maximal 12 features needed to perform basic classification, five are common to both experiments. These are *multi-echo features computed on both PCs, vertical*

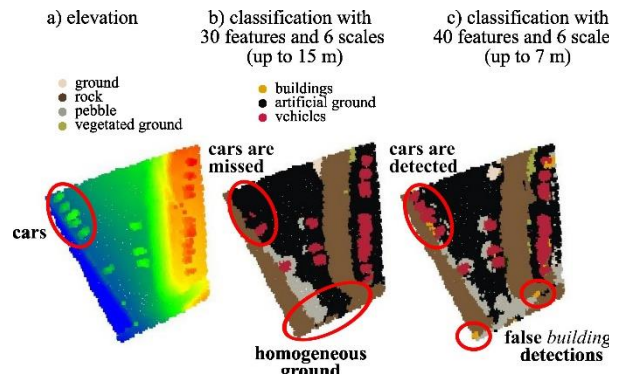


Figure 13: Extracts of classification results obtained depending on the maximal scale included.

distance of green points to their NIR neighbors (KNN), and mean green intensity. They are then combined with site specific attributes. In Fréhel, the optimal predictor set retains mainly multi-echo attributes and intensity-derived information. In the Ain, multi-echo features and height-derived parameters dominate. However, **classical features of 3D data interpretation** such as PCA eigenvalues or covariance eigenvectors ratios delineating the shape of local PCs (Brodu and Lague, 2012; Gross and Thoennessen, 2006; Vandapel et al., 2004; Weinmann et al., 2013) **are almost unused**. They only become more prominent when complexifying the number and types of classes to detect. This is also certainly linked to the fact we analyze airborne lidar data, while these features were designed in priority to describe terrestrial and mobile laser scanning, that include a greater diversity of surface orientations. **Point-based attributes are also completely absent** of the optimized classifications. Newly introduced features based on statistical operators applied to multi-echo features or intensity values systematically outperform them in terms of contribution. Such operators had been tested on height derived values (Antonarakis et al., 2008; Dong et al., 2017) but never applied to other types of features. Yet, **the use of statistical operators seems particularly informative and able to decouple the informative power of point-based characteristics**, namely multi-echo attributes, that never particularly stood out in existing PC classification literature but appear essential to the success of our experiments. They compensate the inability of

decision trees to consider spatial relationships between points, by giving an insight into spatial consistence of considered attributes. These **operators also limitate bias linked to intensity values, that are unavoidable** to classify diverse environments (Song et al., 2002; Yan et al., 2015). Intensity median, mode, skewness or ratio values constitute half of the primary predictors in Fréhel, and are prominent in both advanced models. The statistical operators potentially attenuate spatially inconsistent distributions of radiometric information, mainly in densely vegetated areas. Standard deviation and skewness also mitigate the limitations of this measure, which varies with the acquisition conditions and does not constitute an absolute estimation of surface reflectance (Kashani et al., 2015). Overall, **the features we present seem to better describe natural environments**. We compared classifications of the Ain obtained with 3DMASC’s single cloud features and with features used in Thomas et al. (2018), Hackel et al. (2016) and Chehata et al. (2009). These approaches rely mainly on features derived from the covariance matrix of the core point’s neighborhoods, on height-based parameters, and, less frequently on echo-based parameters. Due to the unavailability of waveform data on our validation areas, we did not include waveform-derived attributes originally exploited in Chehata et al. (2009). We computed each feature set on the green PC only, and at multi-scale spherical neighborhoods with diameters of 2, 3, 4, 5, 6, and 7 m. Details about the features used in each experiment are provided in Supplementary Materials. Overall Accuracies obtained on the validation set for 5 classes in the riverine area by each approach are summed up in Table 8. They show that on natural environments, using our features produces systematically higher results than other existing features.

| 3DMASC | Thomas et al. (2018) | Hackel et al. (2016) | Chehata et al. (2009) |
|--------------|----------------------|------------------------------|--|
| 93.6% | 76.7% | 82.6% | 84.9% |
| | Covariance-based | Covariance- and height-based | Covariance-, height-, echo-, plane-based |

Table 8: Classification accuracies obtained with different types of features on the 5 classes of the Ain dataset.

They also show that using solely covariance-based features produces the lowest OA on our riverine environment, while it generated more precise classifications of urban environments (Thomas et al., 2018), highlighting the need for methods adapted to the different types of 3D data currently in use.

We also **introduce new measures of the reflective behaviour in 3D PCs**, that were mostly estimated through mean intensity, and propose new inter-channel ratios to complement existing multispectral attributes (Morsy et al., 2017b; Wichmann et al., 2015). Previous studies analyzing multispectral lidar faced the difficulty of linking points to their equivalents in PCs of other wavelengths, since they are never in strictly identic positions due to the sensor configuration (Lague and Feldmann, 2020). These new ratios, along with our dual-cloud features compensate the limits of point matching, used in existing multispectral lidar analysis work (Morsy et al., 2017b) when they are used on datasets with correct geometrical and radiometric calibration (Kashani et al., 2015; Yan et al., 2012). **Dual-cloud features systematically stand out among highly contributive features**. Their lower inter-scale correlation

likely contribute to their more informative character, along with their ability to compensate the limits of shallow learning classifiers, that are unable to learn features, and thus to bring out and use connections between features. For example, difference of roughness between NIR and green PC is particularly high for points belonging to water column, and much lower for the riverbed of the bottom of swimming pools, due to the full reflection of NIR laser on the water and scattering of the green light in the water column. The same optical phenomenon explains the higher difference of elevation modes between PCs in swimming pools and river. The inherent points position differences of TB sensors, illustrated in Figure 1, explain the varying vertical distances between green and NIR PCs in vegetated ares, and their systematically negative value in bathymetric zones. Similarly, **the use of a previously classified ground PC as contextual feature** allows to improve the labelling of points at the limit between ground and above ground features, namely building walls and lower tree branches, explaining the improvement observed when they are included, and the smaller scales needed to capture the signature of such variations.

Using these observations, we **recommend the following set of features to use on topo-bathymetric environments**: the NIR and green *number of returns* and *echo ratios*, the green *return number*, the *vertical distance* to the 1 and 10 *nearest neighbors of the core points in the NIR PC*, the *mode* of the green and NIR intensities, the *skewness* of the NIR *intensity*, the *ratio of median intensities*, the NIR and green *elevations’ standard deviation*, the *difference of elevation modes*, the NIR *roughness* and the *difference of roughness*, the NIR *dip*, and the green PC *sphericity*. With these 19 features computed at scales between 1.5 and 14 m, we observed OAs of 98% and 91% for 5 classes on the riverine and coastal datasets, respectively, and 94% and 90% on their 11 and 13 classes versions.

7. CONCLUSION

In this paper, we have introduced 3DMASC, a method for explainable machine learning multispectral point cloud classification. 3DMASC operates directly on sets of unordered, unstructured points and predicts a label for each, with a confidence index and information on the origin of the decision, through feature importance. It differs from previous point cloud classification methods in its capacity to handle multiple clouds simultaneously and describe the spatial and statistical repartition of point cloud attributes, introducing indirect context consideration in the model and new multispectral feature ratios. 3DMASC also stands out from state-of-the-art 3D classification methods with its accessibility: it is explainable using Shapley values, usable without dedicated GPUs, and easy to handle for thematic specialists such as geomorphologists, ecologists, or cartographers. We focus on providing an optimized approach in terms of computation cost, processing time, and results. We demonstrate the performance of the approach on two different airborne lidar use cases: the detection of land and sea covers in (1) a riverine environment and (2) a coastal area. Results show that the method produces highly accurate classifications of basic or detailed categories of points. Furthermore, models excel in TB environments thanks to the newly introduced features and require a limited number of training points (\leq

2000 per class), scales, and attributes. We also implemented a feature selection framework that allows us to draw three main conclusions about the definition of the predictor's vector: (1) statistics of point-based attributes are more informative than classical dimensionality or geometrical features on this type of data, (2) multi-echo features, vertical distance between the two PCs and mean intensity appear to constitute an essential base of features to use and (3) dual cloud features are highly contributive to separate ground, artificial elements and vegetation. Our results also stress the superiority of multi-cloud classification compared to single-cloud, especially for bi-spectral lidar. We release our source code through an open-source plugin in CloudCompare (Girardeau-Montaut, 2022), hoping it will help applications of 3D remote sensing for earth observation and conservation. Although our paper illustrates specific use cases of the workflow on topo-bathymetric lidar datasets, 3DMASC can be extended to PC time series analysis, and 3D data interpretation in general. It may be applied to terrestrial laser scanning data, to SFM PCs, or even to UAV lidar sensors, which are still under development, enlarge the access to lidar surveys, but are too compact to incorporate dual-wavelength lidar sensors.

Acknowledgments : This research was partially funded by the Saur Group and the Region Bretagne, who the authors thank for their support. The Titan DW sensor, operated by the Nantes-Rennes Lidar Platform has been funded by the Region Pays de la Loire with funding of the RS2E-OSUNA programs and the Region Bretagne with support from the European Regional Development Fund. Patrick Launeau is greatly acknowledged for his contribution in the acquisition of the Titan DW sensor. We thank Cyril Michon, Emmanuel Gouraud, William Gentile from Geofit-Expert company, and Laurence Hubert-Moy for their contribution in the over-all operation of the Titan DW sensor. We thank Electricité De France (A. Barillier, A. Clutier) for commissioning the acquisition of the Ain River survey and providing access to the data.

Appendix A: Cloudcompare (CC) q3DMASC plugin implementation and operation

Using the q3DMASC plugin for classifier training or inference requires a labelled core point file and up to 3 accessory point clouds used to compute the features around each core point: PC1 (e.g., green channel), PC2 (e.g., NIR channel) and CTX (e.g., a point cloud with a populated classification field). For single point cloud classification, only one accessory point cloud is needed. A text file contains the description of point clouds, scales and features to be used for training. Upon training completion, a classifier file is saved and can be subsequently used with q3DMASC to apply the classifier to other point clouds.

Here are the main characteristics of the q3DMASC plugin implemented in the open source software CloudCompare (CC):

Accessibility: the q3DMASC plugin has been designed to be usable without programming language knowledge (e.g., Python) directly in the CC GUI. As such it makes a great introductory tool for non-specialists, for teaching and for quick tests without having to setup a complete programming environment. We have also modified the CC scissor tool to

allow direct interactive labelling of 3D data, and introduced a tool to automatically split point clouds according to classes, and a new plugin for labelling data in 3D has just been released (QCloudLayers by Wiggins Tech). These simple tools associated with the neat 3D visualization of CC greatly facilitate the creation of labelled 3D data for training.

Speed: (CC) written in C++ has a well proven, fast and fully parallelized 3D neighbourhood search essential for fast computation of spherical neighbourhood or KNN search. While not critical during the training phase as a limited number of samples is necessary, this is essential during application and production phases to compute features on several millions of points.

Scalability: the q3DMASC plugin can be used in command line mode without GUI in order to apply the classifier in batch mode for large point cloud projects that would not fit in the computer RAM. For instance, we have been able to use it routinely to process projects with more than 10 billions points using tiling strategies.

Non data source specific: while some features of 3DMASC are specific to Airborne LiDAR (e.g., multi-echo features), many geometric features can be used for any type of high resolution 3D point cloud created, for instance, from terrestrial LiDAR, Structure From Motion, Satellite Stereo Photogrammetry and multibeam sonar. There are in particular provision to use RGBNIR information that can be essential for SFM.

Flexibility in feature creation: to generate complex single or dual cloud features over several scales, the user has to create a text file containing the description of the various point clouds, the scales to be used and the features to be computed. Complex single cloud features can be generated using the following formalism:

FEAT_SC#_STAT_PC#

in which FEAT corresponds to a predefined list of features (e.g., *intensity*, *z*, *number of returns*, *sphericity*, ...), SC# indicates the scale at which they will be calculated, STAT is a statistical descriptor for point-based features sampled within the spherical neighbourhood (*mean*, *mode*, *median*, *std*, *range*, *skew*), PC# indicate the point cloud to be used for calculation around the core point. Dual cloud features are generated with this formalism:

FEAT_SC#_STAT_PC#_PC\$_MATH

In which PC\$ indicates the second cloud to be used and MATH is a operator (*minus*, *plus*, *divide*, *multiply*). For instance the Z mode difference (fig. 2) between the green channel (PC1) and the NIR channel (PC2) calculated at all possible scales is written Z_SCx_MODE_PC1_PC2_MINUS. Contextual features are constructed using the following formalism:

DZk_SC0_PC#_CTX#,

In which DZk (resp. DHk) indicates the vertical (resp. horizontal) distance to the *k* nearest neighbours, PC# indicates the PC considered and CTX# the number in the classification field to consider (e.g., 2 for ground, 5 for vegetation...). For instance the average vertical distance to the 3 nearest ground

points of the NIR channel (PC2) that holds a valid classification field is DZ3_SC0_PC2_CTX2.

Explainability: we use a random forest algorithm that combines a good performance on many attributes, simplified feature selection, and robustness to overfitting. After training, the GUI version of 3DMASC outputs the overall accuracy, RF feature ranking and allows to manually remove features that are less contributing. After training completion, users can directly visualize feature values in 3D to understand why they contribute directly or not to classification success.

For training purposes, we chose the cross-platform OpenCV library (Bradski, 2000) implementation of Random Forests as it allows classifiers created in Cloudcompare to be used in Python and vice-versa. The downside of the C++ implementation of OpenCV is that the training is not parallelized, and is consequently much slower than the RF implementation, e.g., of scikit-learn (Pedregosa et al., 2011). RF training is thus the main bottleneck during classifier creation in the CC version. Classifier application is extremely fast, and feature calculation becomes the main bottleneck. Expert users can directly train their classifier in python with their favourite algorithm.

Appendix B: Complete list of features used in this study

Point-based features and single/dual cloud features constructed from them in spherical neighborhood

| Name | Single cloud features stat descriptors (532 nm or 1064 nm) | Dual cloud features (532 and 1064 nm) | |
|-------------------------|--|---------------------------------------|--------------------|
| | | Subtraction | Division |
| Elevation* | Std, Skew | Mean, Median, Mode, Std, Skew | - |
| Intensity | X | Std, Skew | Mean, Median, Mode |
| Return number | Mean | - | - |
| Numb. of returns | Mean | - | - |
| Echo Ratio | Mean | - | - |
| R, G, B | Mean, Mode, Median | - | - |

*: not used as a point based feature

Dimensionality based features computed in spherical neighborhood

| Name | Formulation from eigenvalues | Dual cloud features (532 and 1064 nm) |
|--|---|---------------------------------------|
| PCA1* | $\lambda_1/(\lambda_1 + \lambda_2 + \lambda_3)$ | subtraction |
| PCA2* | $\lambda_2/(\lambda_1 + \lambda_2 + \lambda_3)$ | subtraction |
| PCA3/Surf variation⁺ | $\lambda_3/(\lambda_1 + \lambda_2 + \lambda_3)$ | subtraction |
| Sphericity⁺ | λ_3/λ_1 | subtraction |
| Linearity⁺ | $(\lambda_1 - \lambda_2)/\lambda_1$ | subtraction |
| Planarity⁺ | $(\lambda_2 - \lambda_3)/\lambda_1$ | subtraction |

*: Brodu and Lague (2012); + : Weinmann et al., (2013)

Geometry based features computed in spherical neighborhood

| Name | Information | Dual cloud features (532 and 1064 nm) |
|---------------------|--|---------------------------------------|
| Verticality* | Varies between 0 (horizontal) and 1 (vertical) | subtraction |

| | | |
|----------------------------|--|-------------|
| Detrended Roughness | Std of distance between points and best fitting plane | subtraction |
| Curvature | Mean curvature in CC= average of principal curvatures | subtraction |
| Nb of points | - | subtraction |
| Anisotropy | Ratio of distance to center of mass and radius of sphere | subtraction |
| First Order Moment* | Hackel et al. (2016) | subtraction |

*: Demantké et al., 2012;

Height based metrics computed in spherical neighborhood

| Name | Formulation | Dual cloud features (532 and 1064 nm) |
|---------------|-----------------------|---------------------------------------|
| Zrange | $Z_{\max} - Z_{\min}$ | subtraction |
| Zmin | $Z - Z_{\min}$ | subtraction |
| Zmax | $Z_{\max} - Z$ | subtraction |

z is the core point elevation, Z_{\max} and Z_{\min} are the maximum and minimum elevation in the spherical neighbourhood, respectively.

Contextual features in the NIR channel

| Name | Formulation | Target class |
|------------------|--|----------------|
| DZ to KNN | Mean vertical distance to k nearest neighbor | 1064 nm ground |
| DH to KNN | Mean horizontal distance to k nearest neighbor | 1064 nm ground |

REFERENCES

- Aggarwal, C.C., 2014. Educational and software resources for data classification, in: Data Classification: Algorithms and Applications. pp. 657–665. <https://doi.org/10.1201/b17320>
- Antonarakis, A.S., Richards, K.S., Brasington, J., 2008. Object-based land cover classification using airborne LiDAR. Remote Sens Environ 112, 2988–2998. <https://doi.org/10.1016/J.RSE.2008.02.004>
- Atzmon, M., Maron, H., Lipman, Y., 2018. Point Convolutional Neural Networks by Extension Operators. ACM Trans Graph 37. <https://doi.org/10.48550/arxiv.1803.10091>
- Bradski, G., 2000. The OpenCV Library. Dr. Dobb's Journal of Software Tools.
- Breiman, L., 2001. Random Forests. Machine Learning 2001 45:1 45, 5–32. <https://doi.org/10.1023/A:1010933404324>
- Brodu, N., Lague, D., 2012. 3D terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology. ISPRS Journal of Photogrammetry and Remote Sensing 68, 121–134. <https://doi.org/10.1016/j.isprsjprs.2012.01.006>
- Chehata, N., Guo, L., Mallet, C., 2009. AIRBORNE LIDAR FEATURE SELECTION FOR URBAN CLASSIFICATION USING RANDOM FORESTS.

- Chen, B., Shi, S., Gong, W., Zhang, Q., Yang, J., Du, L., Sun, J., Zhang, Z., Song, S., 2017. Multispectral LiDAR Point Cloud Classification: A Two-Step Approach. *Remote Sensing* 2017, Vol. 9, Page 373 9, 373. <https://doi.org/10.3390/RS9040373>
- Chen, T., Guestrin, C., 2016. XGBoost: A scalable tree boosting system, in: *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, pp. 785–794. <https://doi.org/10.1145/2939672.2939785>
- Chust, G., Grande, M., Galparsoro, I., Uriarte, A., Borja, Á., 2010. Capabilities of the bathymetric Hawk Eye LiDAR for coastal habitat mapping: A case study within a Basque estuary. *Estuar Coast Shelf Sci* 89, 200–213. <https://doi.org/10.1016/j.ecss.2010.07.002>
- Dai, W., Yang, B., Dong, Z., Shaker, A., 2018. A new method for 3D individual tree extraction using multispectral airborne LiDAR point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing* 144, 400–411. <https://doi.org/10.1016/J.ISPRSJPRS.2018.08.010>
- Dash, M., Liu, H., 1997. Feature Selection for Classification. *IDA ELSEVIER Intelligent Data Analysis* 1, 131–156.
- Demantke, J., Mallet, C., David, N., Vallet, B., Dimensionality Based, B.V., Demantké, J., 2011. DIMENSIONALITY BASED SCALE SELECTION IN 3D LIDAR POINT CLOUDS.
- Demantké, J., Vallet, B., Paparoditis, N., 2012. STREAMED VERTICAL RECTANGLE DETECTION IN TERRESTRIAL LASER SCANS FOR FACADE DATABASE PRODUCTION.
- Dong, W., Lan, J., Liang, S., Yao, W., Zhan, Z., 2017. Selection of LiDAR geometric features with adaptive neighborhood size for urban land cover classification. *International Journal of Applied Earth Observation and Geoinformation* 60, 99–110. <https://doi.org/10.1016/J.JAG.2017.04.003>
- Fernandez-Diaz, J.C., Glennie, C.L., Carter, W.E., Shrestha, R.L., Sartori, M.P., Singhania, A., Legleiter, C.J., Overstreet, B.T., 2014. Early results of simultaneous terrain and shallow water bathymetry mapping using a single-wavelength airborne LiDAR sensor. *IEEE J Sel Top Appl Earth Obs Remote Sens* 7, 623–635. <https://doi.org/10.1109/JSTARS.2013.2265255>
- Girardeau-Montaut, D., 2022. CloudCompare (version 2.12.4) [GPL software]. (2022). Retrieved from <http://www.cloudcompare.org/>.
- Gong, W., Sun, J., Shi, S., Yang, J., Du, L., Zhu, B., Song, S., 2015. Investigating the Potential of Using the Spatial and Spectral Information of Multispectral LiDAR for Object Classification. *Sensors* 2015, Vol. 15, Pages 21989–22002 15, 21989–22002. <https://doi.org/10.3390/S150921989>
- Gross, H., Thoennessen, U., 2006. EXTRACTION OF LINES FROM LASER POINT CLOUDS.
- Guan, H., Yu, J., Li, J., Luo, L., 2012. RANDOM FORESTS-BASED FEATURE SELECTION FOR LAND-USE CLASSIFICATION USING LIDAR DATA AND ORTHOIMAGERY.
- Guenther, G.C., Cunningham, A.G., Larocque, P.E., Reid, D.J., Service, N.O., Highway, E., Spring, S., 2000. Meeting the Accuracy Challenge in Airborne Lidar Bathymetry. *EARS eProceedings* 1, 1–27.
- Guyon, I., Elisseeff, A., 2003. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research* 3, 1157–1182.
- Hackel, T., Savinov, N., Ladicky, L., Wegner, J.D., Schindler, K., Pollefeys, M., 2017. SEMANTIC3D.NET: A NEW LARGE-SCALE POINT CLOUD CLASSIFICATION BENCHMARK, in: *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*. pp. 91–98. <https://doi.org/10.5194/isprs-annals-IV-1-W1-91-2017>
- Hackel, T., Wegner, J.D., Schindler, K., 2016. Fast Semantic Segmentation of 3D Point Clouds with Strongly Varying Density. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* III–3, 177–184. <https://doi.org/10.5194/ISPRS-ANNALS-III-3-177-2016>
- Hansen, S.S., Ernstsén, V.B., Andersen, M.S., Al-Hamdani, Z., Baran, R., Niederwieser, M., Steinbacher, F., Kroon, A., 2021. Classification of Boulders in Coastal Environments Using Random Forest Machine Learning on Topo-Bathymetric LiDAR Data. *Remote Sens (Basel)* 13, 4101. <https://doi.org/10.3390/rs13204101>
- Hastie, T., Rosset, S., Zhu, J., Zou, H., 2009. Multi-class AdaBoost. *Stat Interface* 2, 349–360. <https://doi.org/10.4310/SII.2009.V2.N3.A8>
- Hermosilla, P., Ritschel, T., Vázquez, P.-P., Vinacua, À., Ropinski, T., 2018. Monte Carlo convolution for learning on non-uniformly sampled point clouds. *ACM Transactions on Graphics (TOG)*. <https://doi.org/10.1145/3272127.3275110>
- Hsieh, P.-F., Lee, L.C., Chen, N.-Y., 2001. Effect of spatial resolution on classification errors of pure and mixed pixels in remote sensing. *IEEE Transactions on Geoscience and Remote Sensing* 39, 2657–2663. <https://doi.org/10.1109/36.975000>
- Hua, B.S., Tran, M.K., Yeung, S.K., 2018. Pointwise Convolutional Neural Networks, in: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. pp. 984–993. <https://doi.org/10.1109/CVPR.2018.00109>
- Im, J., Jensen, J.R., Hodgson, M.E., 2013. Object-Based Land Cover Classification Using High-Posting-

- Density LiDAR Data. <http://dx.doi.org/10.2747/1548-1603.45.2.209> 45, 209–228.
<https://doi.org/10.2747/1548-1603.45.2.209>
- Kashani, A.G., Olsen, M.J., Parrish, C.E., Wilson, N., 2015. A Review of LiDAR Radiometric Processing: From Ad Hoc Intensity Correction to Rigorous Radiometric Calibration. *Sensors* 15, 28099–28128.
<https://doi.org/10.3390/s151128099>
- Lague, D., Feldmann, B., 2020. Topo-bathymetric airborne LiDAR for fluvial-geomorphology analysis, in: Tarolli, P., Mudd, S.M. (Eds.), *Developments in Earth Surface Processes, Remote Sensing of Geomorphology*. Elsevier, pp. 25–54.
<https://doi.org/10.1016/B978-0-444-64177-9.00002-3>
- Launeau, P., Giraud, M., Ba, A., Moussaoui, S., Robin, M., Debaine, F., Lague, D., le Menn, E., 2018. Full-Waveform LiDAR Pixel Analysis for Low-Growing Vegetation Mapping of Coastal Foredunes in Western France. *Remote Sens (Basel)* 10, 669.
<https://doi.org/10.3390/rs10050669>
- Lecun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature* 521:7553 521, 436–444.
<https://doi.org/10.1038/nature14539>
- Letard, M., Collin, A., Corpetti, T., Lague, D., Pastol, Y., Ekelund, A., 2022a. Classification of Land-Water Continuum Habitats Using Exclusively Airborne Topobathymetric Lidar Green Waveforms and Infrared Intensity Point Clouds. *Remote Sens (Basel)* 14, 341.
<https://doi.org/10.3390/rs14020341>
- Letard, M., Collin, A., Lague, D., Corpetti, T., Pastol, Y., Ekelund, A., 2022b. USING BISPECTRAL FULL-WAVEFORM LIDAR TO MAP SEAMLESS COASTAL HABITATS IN 3D. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLIII-B3-2*, 463–470.
<https://doi.org/10.5194/ISPRS-ARCHIVES-XLIII-B3-2022-463-2022>
- Letard, M., Collin, A., Lague, D., Corpetti, T., Pastol, Y., Ekelund, A., Pergent, G., Costa, S., 2021. Towards 3D Mapping of Seagrass Meadows with Topo-Bathymetric Lidar Full Waveform Processing, in: 2021 IEEE International Geoscience and Remote Sensing Symposium IGARSS. pp. 8069–8072.
<https://doi.org/10.1109/IGARSS47720.2021.9554262>
- Lundberg, S.M., Allen, P.G., Lee, S.-I., 2017. A Unified Approach to Interpreting Model Predictions. *Adv Neural Inf Process Syst* 30.
- Mandlburger, G., Hauer, C., Wieser, M., Pfeifer, N., 2015. Topo-Bathymetric LiDAR for Monitoring River Morphodynamics and Instream Habitats—A Case Study at the Pielach River. *Remote Sens (Basel)* 7, 6160–6195. <https://doi.org/10.3390/rs70506160>
- McKean, J., Nagel, D., Tonina, D., Bailey, P., Wright, C.W., Bohn, C., Nayegandhi, A., 2009. Remote Sensing of Channels and Riparian Zones with a Narrow-Beam Aquatic-Terrestrial LiDAR. *Remote Sensing* 2009, Vol. 1, Pages 1065–1096 1, 1065–1096.
<https://doi.org/10.3390/RS1041065>
- Morsy, S., Shaker, A., El-Rabbany, A., 2022. Classification of Multispectral Airborne LiDAR Data Using Geometric and Radiometric Information. *Geomatics* 2022, Vol. 2, Pages 370–389 2, 370–389.
<https://doi.org/10.3390/GEOMATICS2030021>
- Morsy, S., Shaker, A., El-Rabbany, A., 2017a. Clustering of multispectral airborne laser scanning data using Gaussian decomposition, in: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*. pp. 269–276.
<https://doi.org/10.5194/isprs-archives-XLII-2-W7-269-2017>
- Morsy, S., Shaker, A., El-Rabbany, A., 2017b. Multispectral LiDAR Data for Land Cover Classification of Urban Areas. *Sensors* 2017, Vol. 17, Page 958 17, 958.
<https://doi.org/10.3390/S17050958>
- Niemeyer, J., Rottensteiner, F., Soergel, U., 2012. CONDITIONAL RANDOM FIELDS FOR LIDAR POINT CLOUD CLASSIFICATION IN COMPLEX URBAN AREAS. <https://doi.org/10.5194/isprsannals-I-3-263-2012>
- Niemeyer, J., Wegner, J.D., Mallet, C., Rottensteiner, F., Soergel, U., 2011. Conditional Random Fields for Urban Scene Classification with Full Waveform LiDAR Data, in: Stilla, U., Rottensteiner, F., Mayer, H., Jutzi, B., Butenuth, M. (Eds.), *Lecture Notes in Computer Science*. Springer, pp. 233–244.
https://doi.org/10.1007/978-3-642-24393-6_20
- Pal, M., 2007. Random forest classifier for remote sensing classification.
<http://dx.doi.org/10.1080/01431160412331269698> 26, 217–222.
<https://doi.org/10.1080/01431160412331269698>
- Pan, Z., Glennie, C., Hartzell, P., Fernandez-Diaz, J.C., Legleiter, C., Overstreet, B., 2015. Performance Assessment of High Resolution Airborne Full Waveform LiDAR for Shallow River Bathymetry. *Remote Sens (Basel)* 7, 5133–5159.
<https://doi.org/10.3390/rs70505133>
- Parrish, C.E., Dijkstra, J.A., O’Neil-Dunne, J.P.M., McKenna, L., Pe’eri, S., 2016. Post-Sandy Benthic Habitat Mapping Using New Topobathymetric Lidar Technology and Object-Based Image Classification. *J Coast Res* 76, 200–208. <https://doi.org/10.2112/SI76-017>
- Pauly, M., 2003. Point Primitives for Interactive Modeling and Processing of 3D Geometry.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., others, 2011. Scikit-learn: Machine learning in Python. *Journal of machine learning research* 12, 2825–2830.

- Philpot, W., 2019. Airborne Laser Hydrography II.
- Qi, Charles R., Su, H., Mo, K., Guibas, L.J., 2017. PointNet: Deep learning on point sets for 3D classification and segmentation, in: Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017. pp. 77–85. <https://doi.org/10.1109/CVPR.2017.16>
- Qi, Charles R., Yi, L., Su, H., Guibas, L.J., 2017. PointNet++: Deep hierarchical feature learning on point sets in a metric space, in: Advances in Neural Information Processing Systems. pp. 5100–5109.
- Shapley, L.S., 1952. A Value for N-Person Games. A Value for N-Person Games. <https://doi.org/10.7249/P0295>
- Smeeckaert, J., Mallet, C., David, N., Chehata, N., Ferraz, A., 2013. Large-scale classification of water areas using airborne topographic LiDAR data. *Remote Sens Environ* 138, 134–148. <https://doi.org/10.1016/j.rse.2013.07.004>
- Song, J.-H., Han, S.-H., Yu, K., Kim, Y.-I., 2002. Assessing the possibility of land-cover classification using lidar intensity data, in: International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives.
- Sun, Y.-D., Shyue, S.-W., 2017. A HYBRID SEABED CLASSIFICATION METHOD USING AIRBORNE LASER BATHYMETRIC DATA 8.
- Teo, T.A., Wu, H.M., 2017. Analysis of Land Cover Classification Using Multi-Wavelength LiDAR System. *Applied Sciences* 2017, Vol. 7, Page 663 7, 663. <https://doi.org/10.3390/APP7070663>
- Thomas, H., Goulette, F., Deschaud, J.E., Marcotegui, B., Gall, Y. le, 2018. Semantic classification of 3d point clouds with multiscale spherical neighborhoods, in: Proceedings - 2018 International Conference on 3D Vision, 3DV 2018. pp. 390–398. <https://doi.org/10.1109/3DV.2018.00052>
- Thomas, H., Qi, C.R., Deschaud, J.E., Marcotegui, B., Goulette, F., Guibas, L., 2019. KPConv: Flexible and deformable convolution for point clouds, in: Proceedings of the IEEE International Conference on Computer Vision. pp. 6410–6419. <https://doi.org/10.1109/ICCV.2019.00651>
- Tulldahl, H.M., Wikström, S.A., 2012. Classification of aquatic macrovegetation and substrates with airborne lidar. *Remote Sens Environ* 121, 347–357. <https://doi.org/10.1016/j.rse.2012.02.004>
- Vandapel, N., Huber, D.F., Kapuria, A., Hebert, M., 2004. Natural terrain classification using 3-D lidar data. *Proc IEEE Int Conf Robot Autom* 2004, 5117–5122. <https://doi.org/10.1109/ROBOT.2004.1302529>
- Wedding, L.M., Friedlander, A.M., McGranaghan, M., Yost, R.S., Monaco, M.E., 2008. Using bathymetric lidar to define nearshore benthic habitat complexity: Implications for management of reef fish assemblages in Hawaii. *Remote Sens Environ, Applications of Remote Sensing to Monitoring Freshwater and Estuarine Systems* 112, 4159–4165. <https://doi.org/10.1016/j.rse.2008.01.025>
- Weinmann, Martin, Jutzi, B., Hinz, S., Mallet, C., 2015. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing* 105, 286–304. <https://doi.org/10.1016/J.ISPRSJPRS.2015.01.016>
- Weinmann, M., Jutzi, B., Mallet, C., 2013. FEATURE RELEVANCE ASSESSMENT FOR THE SEMANTIC INTERPRETATION OF 3D POINT CLOUD DATA. <https://doi.org/10.5194/isprsannals-II-5-W2-313-2013>
- Weinmann, M., Urban, S., Hinz, S., Jutzi, B., Mallet, C., 2015. Distinctive 2D and 3D features for automated large-scale scene analysis in urban areas. *Comput Graph* 49, 47–57. <https://doi.org/10.1016/J.CAG.2015.01.006>
- West, K.F., Webb, B.N., Lersch, J.R., Pothier, S., Triscari, J.M., Iverson, A.E., 2004. Context-driven automated target detection in 3D data. <https://doi.org/10.1117/12.542536> 5426, 133–143. <https://doi.org/10.1117/12.542536>
- Wichmann, V., Bremer, M., Lindenberger, J., Rutzinger, M., Georges, C., Petrini-Monteferrri, F., 2015. EVALUATING THE POTENTIAL OF MULTISPECTRAL AIRBORNE LIDAR FOR TOPOGRAPHIC MAPPING AND LAND COVER CLASSIFICATION. <https://doi.org/10.5194/isprsannals-II-3-W5-113-2015>
- Wilson, N., Parrish, C.E., Battista, T., Wright, C.W., Costa, B., Slocum, R.K., Dijkstra, J.A., Tyler, M.T., 2019. Mapping Seafloor Relative Reflectance and Assessing Coral Reef Morphology with EAARL-B Topobathymetric Lidar Waveforms. *Estuaries and Coasts*. <https://doi.org/10.1007/s12237-019-00652-9>
- Xu, Y., Fan, T., Xu, M., Zeng, L., Qiao, Y., 2018. SpiderCNN: Deep learning on point sets with parameterized convolutional filters, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). pp. 90–105. https://doi.org/10.1007/978-3-030-01237-3_6
- Yan, W.Y., Shaker, A., El-Ashmawy, N., 2015. Urban land cover classification using airborne LiDAR data: A review. *Remote Sens Environ* 158, 295–310. <https://doi.org/10.1016/J.RSE.2014.11.001>
- Yan, W.Y., Shaker, A., Habib, A., Kersting, A.P., 2012. Improving classification accuracy of airborne LiDAR intensity data by geometric calibration and radiometric correction. *ISPRS Journal of Photogrammetry and Remote Sensing* 67, 35–44. <https://doi.org/10.1016/J.ISPRSJPRS.2011.10.005>

Zavalas, R., Ierodiaconou, D., Ryan, D., Rattray, A., Monk, J., 2014. Habitat Classification of Temperate Marine Macroalgal Communities Using Bathymetric LiDAR. *Remote Sens (Basel)* 6, 2154–2175.
<https://doi.org/10.3390/rs6032154>

Supplementary materials :

Figure 1 presents the results of the experiments made to determine the scale to use for evaluation and the correlation threshold to apply during feature selection.

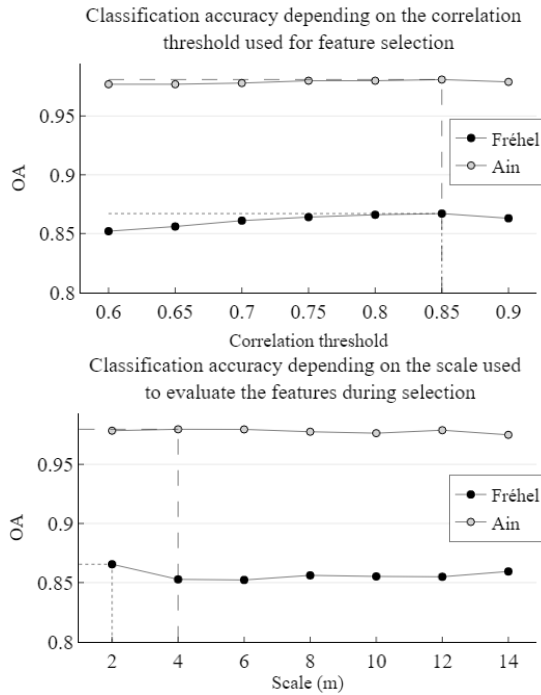


Figure 1: Results of the experiments performed to determine the correlation threshold to apply for feature selection, and the scale at which to evaluate each feature's information gain.

Figure 2 gives more detailed information on the correlation between features computed at different scales.

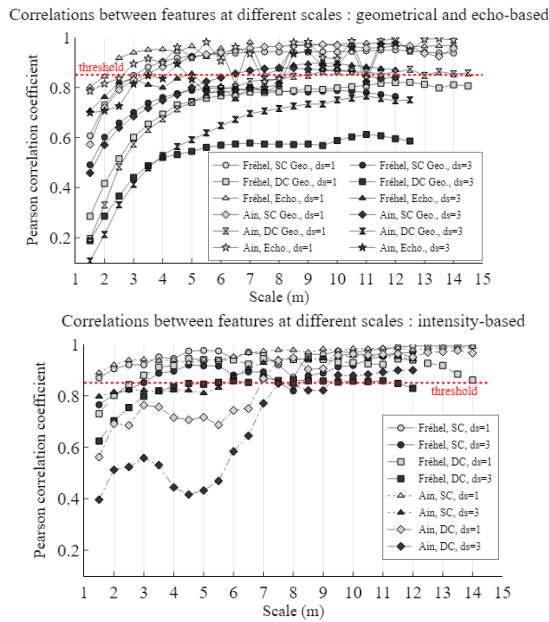


Figure 2: Linear correlation between features computed at scales separated by $dx=1$ m or $dx=3$ m for different families of feature. SC = Single Cloud ; DC = Dual Cloud

Figure 3 presents class-wise metrics obtained by the advanced optimal multi-scale models on both scenes.

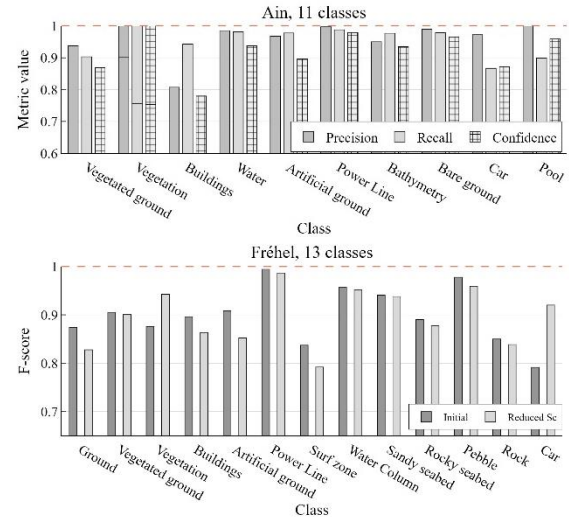


Figure 3: Classification metrics obtained by the optimal multi-scale classifier on each class of the advanced processing of the riverine (Ain) and the coastal (Fréhel) point clouds.

Figure 4 illustrates the impact of large scales on classification accuracy depending on the presence or absence of contextual features (vertical distances to a previously classified ground point cloud). Scales were removed iteratively per decreasing order.

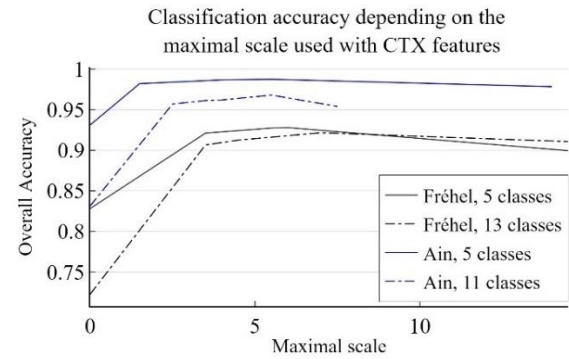


Figure 4: Classification performances depending on the maximal scale kept in the optimal multi-scale predictor set, and in the predictor set augmented with contextual attributes.

Table 1 details the features used to compare 3DMASC to other approaches. The eigenvalues referred to are those obtained on the covariance matrix of spherical neighborhoods. For detailed mathematical expressions of the different attributes, please consult the original papers (Chehata et al., 2009; Hackel et al., 2016; Thomas et al., 2018).

| Name | Thomas et al. (2018) | Hackel et al. (2016) | Chehata et al. (2009) |
|--------------------|----------------------|----------------------|-----------------------|
| Sum of eigenvalues | X | X | |
| Omnivariance | X | X | |
| Eigenentropy | X | X | |
| Anisotropy | | X | X |
| Linearity | X | X | X |
| Planarity | X | X | X |

| | | | |
|---|---|---|---|
| Sphericity | X | X | X |
| Curvature | | | X |
| Surface variation | X | X | |
| Verticality | | X | |
| Verticality based on 1 st eigenvector | X | | |
| Verticality based on 3 rd eigenvector | X | | |
| Vertical moment (1 st order) | X | | |
| Vertical moment (2 nd order) | X | | |
| Number of points | X | | |
| Statistical moments of eigenvectors (1 st and 2 nd order) | X | X | |
| Z range in neighborhood | | X | |
| Difference with minimal Z in neighborhood | | X | X |
| Difference with maximal Z in neighborhood | | X | |
| Standard deviation of Z in neighborhood | | | X |
| Residuals of the fitting of a plane to the neighborhood | | | X |
| Deviation angle of a fitted plan normal to the vertical | | | X |
| Variance of the deviation angles in the neighborhood | | | X |
| Distance to the fitted plan | | | X |
| Number of returns | | | X |
| Normalised return number | | | X |

Table 1: description of the features used in each approach compared to 3DMASC on the Ain dataset.