Literature Review: Classification of Point Clouds

Leah Gluckman
University of Cape Town
Cape Town, South Africa
glclea001@myuct.ac.za

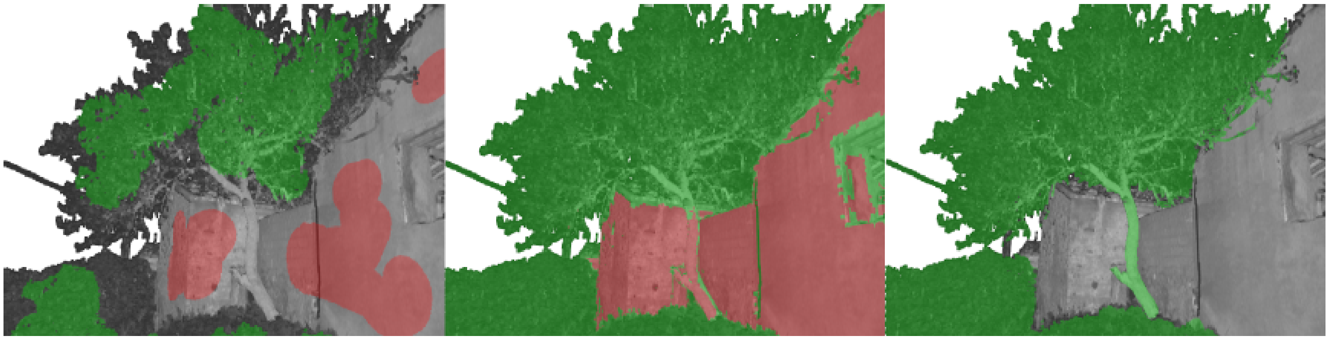


Figure 1: Graphical Depiction of the Point Cloud Classification Pipeline, showcasing the phases of segmentation (leftmost image), classification (middle image), and the correction phase that evaluates the automated classifier's misclassification and improves the result (rightmost image) [14]

ABSTRACT

Point Cloud Classification seeks to automate the process of assigning class labels to points in a point cloud. Point clouds are created from laser scans but contain a lot of noise that one needs to remove, such as foliage, walls, people, etcetera. The classification task comprises labelling each point in the point cloud as noise or some semantically meaningful class. This process is currently mainly manual, labour-intensive and time-consuming. After classification, one can construct a 3D model for use in many areas, including Cultural Heritage preservation to record the three-dimensional structure of historical sites. This literature review evaluates various Machine Learning classification models to automate the process of point cloud cleaning. Several papers show that the Random Forest (RF) classifier can successfully and accurately classify point clouds. One implementation of an RF achieved an 87% prediction accuracy in classifying structural elements in noisy environments. Another used a binary point labelling system to incrementally train an RF and achieved a 98% classification accuracy. Clustering is an unsupervised approach to classification that is suitable for point cloud classification where there is a limited amount of labelled training examples. The papers discussed in this review show that clustering can produce favourable results and effectively classify point clouds whilst being simple to implement. This review also analyses Support Vector Machines (SVM) as an approach for point cloud classification but concludes that they are computationally expensive. In addition, it finds that SVMs achieve a lower accuracy than Instance-Based, Probabilistic and Ensemble classifiers.

CCS CONCEPTS

• **Computing methodologies → Machine learning approaches**; **Classification and regression trees**.

KEYWORDS

Semantic Segmentation, Point Clouds, Machine Learning, Classification

1 INTRODUCTION

The construction of 3D models from laser scans of Cultural Heritage (CH) sites is a vital but challenging area of research for the preservation of CH. Many researchers globally, including the Zamani group at UCT [20], have been researching this for years, particularly the issue of automating the process of point cloud cleaning. This process aims to classify 3D points in laser scans as either relevant or not (i.e., background clutter). Currently, this cleaning process is labour intensive, time-consuming, and mainly manual. This project aims to explore the application of machine learning to the semantic segmentation of 3D point clouds. In this case, semantic segmentation is when a classifier assigns a point to a meaningful class, such as walls, scaffolding, foliage, and more. The remaining groups of points receive the label 'not of interest'. Many issues complicate the classification of point clouds. These issues include uncertainty around the number of classes, the degree of variance within each group, and the class definitions [2]. One must select a classification algorithm that will appropriately and accurately predict the class of points within a point cloud.

2 POINT CLOUDS

Point clouds are a set of data points in three-dimensional space, defined by three spatial coordinates: x, y and z. These points can contain additional information such as colour or intensity. Point clouds can convert into 3D digital models and thus, are an effective and popular mode of representing structures or objects from the real world. A Light Detection And Ranging (LiDAR) scanner creates point clouds from objects. RGB-D sensors augment RGB images with additional depth data [10]. These sensors, along with LiDAR scanners, enable point clouds to be easily and directly obtained. Point clouds contain various pieces of information that describe different properties of points, called features. These features are invariant to transformations [18] and can be global or local. Features develop from a 3D object's specific local or global geometric characteristics [8], and descriptive features make it possible to perform segmentation and classification on point clouds [17]. Researches apply point clouds to many situations, including Cultural Heritage preservation [13], autonomous driving [23] and vehicle detection [5].

3 MACHINE LEARNING

Machine Learning (ML) is an area of Artificial Intelligence which focuses on creating algorithms that gradually improve their accuracy in completing a task on unseen data. ML algorithms adapt to learning data, similarly to how humans learn, and make decisions using empirical and training data [7]. They aim to generalise from experience. ML can divide into two categories: supervised and unsupervised learning. The distinction between these learning approaches depends on whether the algorithm is given labelled training examples as input or not. ML approaches to point clouds comprise processes of segmentation and classification.

3.1 Segmentation

Segmentation is a process that divides a data set, such as a point cloud, into many homogenous classes with similar properties. Each resulting group has unique, differentiable, and measurable attributes, and adjacent groups hold points with unrelated properties [8]. Feature extraction must occur before point cloud segmentation and classification. Efficient analysis of these partitions can then take place in the classification component.

3.2 Classification

Classification, also referred to as semantic segmentation, is a process that categorises a dataset into a set of classes with similar properties. It is the task of assigning labels to the regions produced by segmentation. After segmentation divides a 3D dataset into groups, classification assigns a class label to each group, providing each one with some semantics (see Figure 1) [8]. It uses contextual and geometric features, including area, orthogonality, orientation, proximity, coplanarity and similarity. They are accumulated in feature vectors and processed by a pre-trained classification algorithm to predict the labels [2]. These algorithms are called classifiers.

3.3 Supervised vs Unsupervised Classification

Classification typically involves using some supervised learning algorithm to classify data points within a data set (i.e., points within a cloud). However, one can also use unsupervised learning techniques for classification.

3.3.1 Supervised Learning. Supervised learning is an approach whereby an algorithm predicts target variables, given a set of labelled training examples defined by an attribute set and assigned to a class. First, a classifier learns from a set of labelled data. Then, the trained classifier provides a semantic segmentation of all the data in the set [8]. Supervised classification algorithms have the disadvantage of requiring labelled training data, which is not always accessible. This inaccessibility is particularly true in the case of point cloud classification, where the labelling process is complex and where there is a large amount of data with many different types of objects. The Random Forest classifier, discussed later, is one of the most used supervised learning algorithms for the task of classification [2, 7, 8, 15, 17, 22]. There are, however, many other supervised approaches to point cloud classification including Support Vector Machines [2, 19, 25], Artificial Neural Networks and Conditional and Markov Random Fields [16].

3.3.2 Unsupervised Learning. Unsupervised learning algorithms find patterns in unlabelled data. They aim to predict a target variable without ever seeing examples. Feature learning is an example of unsupervised learning. It is an algorithm that discovers features required to describe data [17]. Another example is clustering, which partitions data into homogenous subgroups called clusters, such that each cluster contains objects that bear more similarity to one another than those in other clusters [8]. A shortcoming is that unsupervised classification does not always produce semantically meaningful classes.

4 POINT CLOUD CLASSIFICATION

Manually removing unwanted or unnecessary data points and noise from point cloud data, such as LiDAR scans, is called point cloud cleaning [15]. In the case of Cultural Heritage (CH) sites [20], point cloud cleaning allows one to remove foliage, scaffolding, and other objects picked up from the scan, to create a 3D model of the CH site. This step is incredibly time-consuming, and an automated classification approach is required. Point Cloud Classification is a machine learning approach to predicting the class of unseen points in point clouds [17]. A common strategy is to use a supervised classifier, such as a Random Forest classifier, as seen in several papers [2, 7, 8, 15, 25] or a Support Vector Machine [19, 25]. In this approach, according to Pocock [17], the classifier trains on a set of features extracted from labelled training data points in a subset of campaign laser scans selected for the training phase. This method is a campaign-based approach whereby two subsets of campaign scans are chosen from a single location and used for the training and prediction phases, respectively. It is beneficial for classifying terrestrial scan campaigns that often contain structures unique to the site, such as in the paper by Mulder and Marais [15]. Another approach is to train general classifiers on external datasets. However, this approach may not generalise well and may struggle to classify terrestrial scans that contain unseen objects. This struggle is particularly true for CH campaigns since there is little consistency amongst different campaigns. In addition, each scene differs from the next, ranging from strong-standing structures in urban spaces to eroding underground ruins. After the classifier has completed the training phase, it can predict the class of points in the campaign's unseen scans. The accuracy and speed of this classification component depend on the extracted features, the amount of detail in the training examples, and the type of classifier. The following sections explain various types of classifiers for classifying datasets.

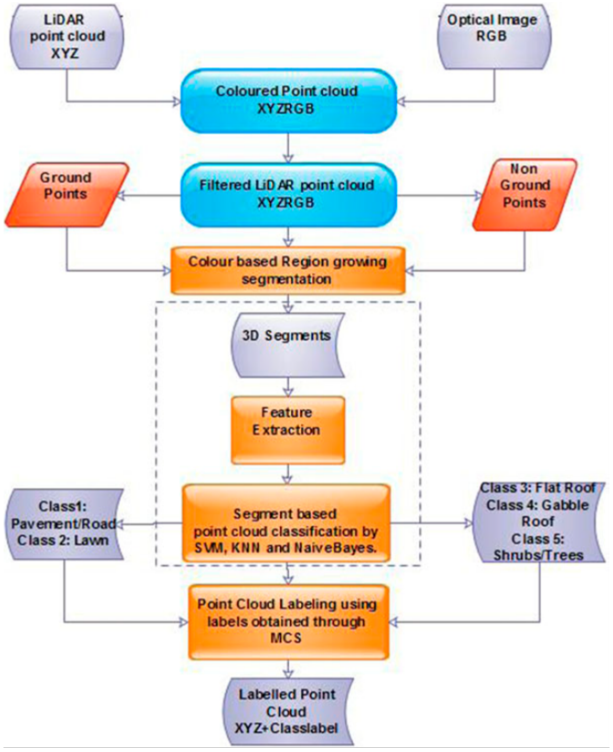

LiDAR point cloud XYZ
Optical Image RGB
Coloured Point cloud XYZRGB
Ground Points
Filtered LiDAR point cloud XYZRGB
Non Ground Points
Colour based Region growing segmentation
3D Segments
Feature Extraction
Class1: Pavement/Road Class 2: Lawn
Segment based point cloud classification by SVM, KNN and NaiveBayes.
Class 3: Flat Roof Class 4: Gabble Roof Class 5: Shrubs/Trees
Point Cloud Labeling using labels obtained through MCS
Labelled Point Cloud XYZ+Classlabel

Figure 2: Flow Diagram depicting the Pipeline of Point Cloud Classification for classifying urban land as either 'ground' or 'non-ground' using an SVM, *k*-NN and Naïve Bayesian Classifier [19]

5 CATEGORIES OF CLASSIFIERS

Many different classifiers perform diverse functions, each with advantages and limitations. Weinmann et al. [21] compare and evaluate different types of classifiers. This section identifies several fundamental classifiers and provides a more in-depth analysis.

5.1 Instance-Based Classifiers

Instance-based classifiers use a lazy-learning approach in that a classifier does not learn a model. The classifier directly compares vectors within the training set to unseen feature vectors [21]. The model classifies data samples according to a distance or similarity metric, such as the Euclidean distance metric [4]. This metric labels feature vectors nearby as more similar to one another than feature vectors further away from one another. Unseen samples classify as the class label of the closest training example. According to Pocock [17], the main advantage of this type of classification is that there is no training phase, which saves resources such as computational power and time. The classifier can also effortlessly adapt to new data since instances can be added or removed from the dataset, rather than training a new model again, as is done in Model-based learning. However, classification has a high cost and can delay at runtime as the dataset and its consequent complexity increase. In addition, the classifier can overfit to noise in the dataset if the vector within the closest proximity is noise. An example of this type of classifier is the *k*-Nearest Neighbour (*k*-NN) classifier, which chooses the *k* nearest samples in the training data and classifies them using the majority vote of their respective class labels [21]. The implementation by Ramiya et al. [19] of *k*-NN classifies objects in an urban landscape as ground or non-ground. It achieves an overall classification accuracy of 96,67% and 84,48% for each respective class. The classifier outperforms the Support Vector Machine in this paper and the one by Bassier et al. [2] on classifying point cloud data of building objects.

5.2 Rule-Based Classifiers

Rule-based classifiers decide which class to assign a data sample to based on a set of rules [17]. These rules are easily interpretable and similar to *if statement* rules since they often are in the format of binary decisions. They evolve as the classifier receives more training examples so they can apply to unseen data. An example of this type of classifier is a decision tree. A decision tree represents data in binary decisions and has a hierarchical tree structure. Decision trees construct using a top-down strategy, and at each step, the variable that splits the given training set best is chosen [21]. Eum et al. [5] use a decision tree algorithm with vertical and horizontal features of vehicle segments to classify a model for vehicle detection from airborne LiDAR point clouds. This approach improves previous vehicle detection methods by more than 13% for recall and 30% for precision. However, rule-based classifiers perform poorly in the study conducted by Weinmann et al. [21] compared to other types of classifiers. Their overall accuracy percentage is lower than instance-based, ensemble, max-margin, probabilistic and Deep Learning classifiers.

5.3 Probabilistic Classifiers

Probabilistic classifiers predict a probability distribution over a set of classes to which a given unseen data sample could belong. These classifiers predict a point's class as being the one with the highest probability. The Naïve Bayesian (NB) classifier is a probabilistic classifier built on Bayes' theorem. It assumes that all features are conditionally independent [21]. Lai and Fox [11] implement this classifier to recognise objects in various environments (both outdoor and indoor) for robotic navigation and achieve favourable results. The classier outperforms other classification methods, including Support Vector Machines and boosting. Similarly, the implementation by Ramiya et al. [19] of a Naïve Bayesian classifier (see Figure 2) performs well, with an accuracy of over 90% and close to 80% for predicting ground and non-ground classes, respectively. It aims to classify urban land and performs on par with an SVM but with lower accuracy than the *k*-NN classifier.

5.4 Max-Margin Classifiers

Max-margin classifiers maximise the distance between data samples belonging to different classes in the dataset. Support Vector Machines (SVM) are max-margin classifiers which classify unstructured and high-dimensional data accurately [17]. SVMs, discussed later, are designed to separate a pair of classes linearly and do this by creating a single or set of hyperplanes in high-dimensional feature space [21]. In cases where it is impossible to separate the feature space linearly, the classifier utilises a kernel function to implicitly map the training data into a new, higher-dimensional feature space where linear separation of the data is possible [21]. Eum et al. [5] introduce a method of identifying vegetation in point clouds. It uses an SVM classifier to classify regions within a point cloud as either vegetation or non-vegetation based on the region's features. The results are successful - especially when using eigenvalue histogram features for the classification component. The max-margin classifiers evaluated in the paper by Weinmann et al. [21] perform with a high degree of accuracy. However, for that case, the SVM proves not to be the most efficient classifier since SVMs are computationally expensive, and the classification requires additional training time for parameter turning. Consequently, the SVM takes longer than other evaluated classifiers, despite its degree of accuracy.

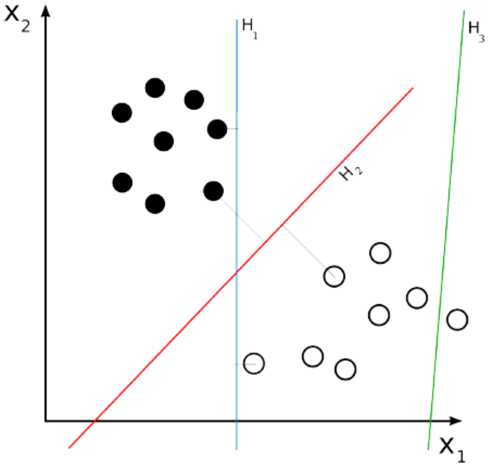


Figure 3: Graphical Depiction of Support Vector Machine Hyperplanes (blue = H_1, red = H_2, green = H_3) separating training data samples. H_2 maximises the distance between the nearest data samples from each cluster. [17]

5.5 Ensemble Classifiers

Ensemble classifiers combine several weak classifiers (in the form of decision trees) into a single accurate classifier. Approaches to ensemble classifiers include boosting and bagging. Bagging is often in the form of the Random Forest classifier. It involves training the weak learners on a randomly chosen subset of the training data and predicting the class of an unseen sample by using the majority vote of those learners [17]. This process results in a set of weak learners that are randomly diverse and, consequently, no correlation between the individual hypotheses. The output (which is the majority vote over all the hypotheses) is greatly generalised and robust [21]. Boosting is the approach of incrementally producing a set of weak classifiers over distributions of the training data and consecutive iterations [21]. The ensemble is built incrementally by feeding instances that models achieved low accuracy as input to newer models during the training phase [17]. After several iterations, the majority vote over all hypotheses is determined [21]. The synthesis of a diverse set of weak learners is the main advantage of ensemble classifiers. Ensemble classifiers successfully reduce bias and are less susceptible to overfitting because each learner is training with different training data [17]. Random Forests improve on this by randomly choosing the training data for each learner in the set. The random selection further diversifies the ensemble and increases its robustness. The disadvantages of ensemble classifiers are that manually selecting training examples is time-consuming, and accuracy can be negatively affected by choosing poor training samples [17]. In addition, it is challenging to interpret ensemble classifiers visually.

5.6 Deep Learning

In Deep Learning, classifiers learn new representations of training samples through several transformation levels. Data is modified at each level and fed as input to a higher and increasingly abstract level [17]. These transformations produce complex data descriptions. Deep learning classifiers are very accurate. In addition, by learning features directly from training data, classifiers learn faster than previously mentioned learners. One disadvantage is that supervised Deep Learning requires more labelled data to produce successful output than other classifiers. Thus, Deep Learning is not the preferred classifier choice when there is limited labelled data.

One Deep Learning classifier is a Multi-Layer Perceptron (MLP). An MLP contains an input layer, one or more hidden layers, and an output layer. Layers are fully connected, and a weight factor specifies connections [21]. These weights are learned during the training phase using the technique of backpropagation. PointNet [23] uses Deep Learning and MLPs to classify point clouds by separately performing feature extraction on each point. The classifier reads point cloud data directly and then learns several optimisation functions that choose the most descriptive data points [17]. When predicting the class of an instance, a symmetric function accumulates these functions' output to assign class labels in the final layer [17]. PointNet is simple to implement but efficient and achieves permutation invariance [25]. It attains a 76,82% overall accuracy for the task of semantic segmentation and significantly outperforms an MLP method. However, PointNet's inability to represent local structures produced by the metric of the feature space is limiting. This limitation influenced the creation of other models that improve on PointNet, such as PointNet++, PointCNN and PointHop [25].

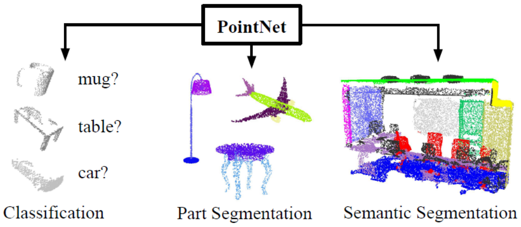


Figure 4: Illustration of the PointNet Deep Learning Classification Pipeline [18]

Whilst there are successful Deep Learning applications for point cloud classification, like PointNet [18], such algorithms have disadvantages. These include an expensive use of computational resources, long training times, and difficulty interpreting visually [24].

Moreover, they require large amounts of training data and often the labelled kind, which is not always plausible. PointHop++ [24] is a 3D point cloud classification technique that performs on par with Deep Neural Network algorithms, such as the Deep Learning classifier PointNet, through its use of unsupervised feature extraction. The unsupervised feature extraction component is tree-structured. Feature vectors are associated with leaf nodes in the tree. In addition, features are ordered hierarchically according to their discriminant power [24].

6 MACHINE LEARNING MODELS FOR POINT CLOUD CLASSIFICATION

Whilst there are many classifiers and approaches to point cloud classification, three Machine Learning classification models appear relevant and applicable to the problem of point cloud classification for CH preservation: Support Vector Machines, Random Forests and Clustering.

6.1 Random Forest

Random Forest (RF) classifiers construct a set of decision trees, each created from permuted versions of the training data and combined into a single tree that performs accurate predictions. Each constructed tree is given the same parameters for training. However, their training sets are selected randomly, with replacement, from the dataset and are the same size as the original dataset [17]. Thus, each training set differs. Training sets can contain more than one copy of the same feature vector, and they can also have zero copies of a particular vector. In the prediction phase, a data sample is given to each tree within the forest to classify. The majority vote in the RF determines the final predicted output, as illustrated in Figure 5. RFs have classification applications in many data samples, including floors and ceilings [2], vegetation and buildings. However, the highly cluttered environments cause problems, and several classification errors occur because of the increased variance of feature vectors within some classes [2].

RFs overcome many decision tree learning problems, including some faced by rule-based classifiers. For instance, RFs solve issues of overfitting the training data by selecting random subsets of the original dataset and a random group of input features for each subset [7]. Several papers [2, 7, 8, 15, 25] implement RFs with feature vectors (both contextual and geometric) for the semantic labelling of points within point clouds. Mulder and Marais [15] use a boosted RF classifier to classify points within a point cloud based on coarse user labelling, and Bassier et al. [2] achieved 87% prediction accuracy when using an RF to predict structural elements in noisy environments. Whilst these approaches reduce the time required to classify points compared to manual classification processes and achieve some degree of accuracy during the prediction phase, the amount of time needed to adjust point labels to attain high accuracy remains long [15]. In addition, these approaches involve learning class labels for every point that partitions the point cloud into classes (ground cover, walls, trees). This learning requires a large amount of labelled training data containing instances of every object, including all variations of that object. Having such a large amount of labelled training data is unrealistic, especially for point cloud classification.

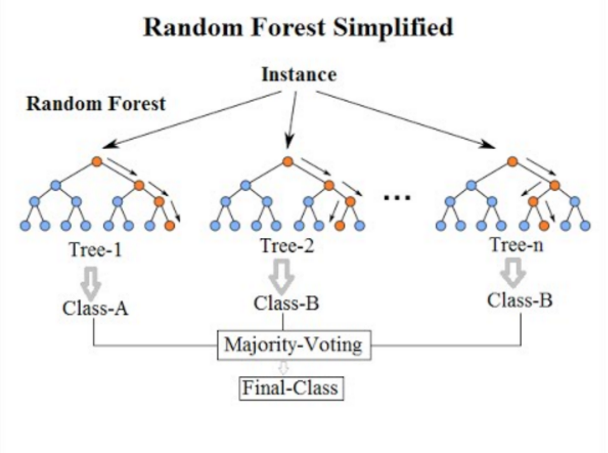


Figure 5: Graphical depiction of Random Forest Prediction that illustrates how multiple weak classifiers are combined into a single, accurate classifier and depicts how the majority vote forms to decide the final class label of an instance [9]

The most successful implementation of Random Forests for point cloud classification is that approached by Marais et al. [13]. A semi-automated point cloud cleaning implementation trained an RF classifier incrementally on a binary labelling system, starting from a manually cleaned scan. This binary point labelling system was restricted to 'keep' and 'discard' labels. The classifier predicted the class label for the following scan in the sequence and achieved a 98% average accuracy. This approach is the first time a point cloud classification task used only two intuitive class labels.

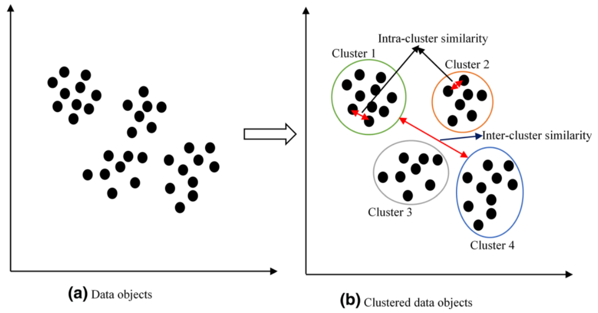
6.2 Support Vector Machines

A Support Vector Machine (SVM) is a supervised max-margin classifier that finds a hyper-plane within the high-dimensional feature space that maximises the distance between the closest feature vectors from each class. In this way, it classifies the point cloud. It maps samples to the hyper-space and assigns a class label based on which side of the hyper-plane they belong to (see Figure 3). SVMs have successfully classified outdoor and indoor point cloud data [2, 19, 25]. Ramiya et al. [19] introduce a multiclass SVM to classify the dataset. This method considers the problem a multiple binary classification task, and the SVM performs well at segmenting a dataset into 'ground' and 'non-ground' classes. Bassier et al. [2] and Zhang et al. [25] compare SVMs to RF, and in both cases, the SVM does not perform as well and achieves a lower accuracy than the RF.

6.3 Clustering

Clustering is a typically unsupervised ML approach. It involves grouping similar samples into clusters to classify data and detect outliers. In all the clusters there is maximised intra-cluster similarity and minimised inter-cluster similarity. Several clustering algorithms have applications in point cloud classification [8]. K-means is a popular clustering algorithm which partitions the dataset into k clusters. It is easy to understand and fast to implement. However, one must specify the number of clusters (k) in advance, and the algorithm does not necessarily account for outliers.

The fact that clustering is an unsupervised classification approach is highly advantageous to point cloud classification, particularly that of Cultural Heritage preservation, because of the limited labelled data. Zhang et al. [26] used a multi-level point clustering algorithm to classify large and small objects in point clouds. The clustering algorithm significantly improved the classification precision, especially for the classification of small objects, compared to other tested clustering methods. It also used correlations between multi-level point clusters to its advantage to accurately predict the classes of unseen instances.



(a) Data objects

(b) Clustered data objects

Figure 6: Illustration of clustering a set of data points into clusters, showcasing inter-cluster and intra-cluster similarity [6]

One of the main disadvantages of classifying point clouds using clustering-based approaches is clustering methods cannot always discern meaningful and relevant clusters in data sets with substantial density discrepancies or high-dimensional data – both of which are common to point clouds. To combat this, Aljumaily et al. [1] created a fully automatic and scalable clustering-based approach for point cloud classification that is not reliant on a pre-processing step and utilises the 3D coordinate information only. A DBSCAN clustering algorithm based on point density effectively clusters and classifies points within a point cloud whilst discarding noise. However, the algorithm requires a large amount of memory to perform accurately.

Belton et al. [3] and Maligo and Lacroix [12] address the problem of point cloud classification using Gaussian Mixture Model (GMM) clustering algorithms. These algorithms are unsupervised and parametric. A GMM assumes the overall distribution is a combination of a finite number of Gaussian distributions. It then determines the attributes of the respective distributions so that the final combination best models the distribution of points in the data set. Each resulting Gaussian model represents a cluster of data points within the point cloud [3].

Belton et al. [3] use a GMM to cluster and classify trees from point clouds extracted from laser scans of forestry landscapes. Maligo and Lacroix [12] introduce a semi-supervised classification approach which involves training a GMM in an unsupervised manner and then classifying a dataset into semantic classes. The latter requires supervised training of the grouping based on the output of the GMM. Using an unsupervised approach eliminates the need for input data to be manually labelled. This method achieves a data set generalisation. These two papers [3, 12] detect differentiation within the data using unsupervised learning. They address clutter and nonuniform sampling and successfully classify the data into semantically interpretable classes. Most notably, they are simple to implement as they do not require pre or post-processing steps. However, both papers suffer the defect of computational complexity and training noise. The presence of randomness in the GMM training creates training noise which negatively affects the classification performance. In addition, the computational complexity of the GMM by Maligo and Lacroix [12] is analogous to linear complexity. Thus, it is worse than that of state-of-the-art supervised classification approaches.

7 CONCLUSIONS

The current manual process of point cloud classification is labour intensive and time-consuming. There is a need to automate this process and create an algorithm that can accurately and efficiently predict the class labels of points within a point cloud. Several studies have applied many different Machine Learning approaches to point clouds to solve this issue. These include supervised algorithms, which receive a set of labelled training data, and unsupervised algorithms, which receive no labelled examples. Through an analysis and comparison of various classifiers, this literature review has shown that each classifier has advantages and limitations that contribute to its classification performance.

Instance-based classifiers require no training phase, thereby saving time and computational resources. However, classification is expensive, and the classifier can overfit to noise. The Probabilistic and k-NN classifiers achieve favourable results and outperform the Support Vector Machine, but the k-NN classifier has the higher overall classification accuracy of the two. Rule-based classifiers, such as the decision tree, perform poorly and with lower accuracy than all the other evaluated classifiers. Max-margin classifiers, including the SVM, are very accurate but computationally expensive and require long training times. They are less efficient than other examined classifiers and achieve a lower accuracy than the RF classifier.

The Random Forest Ensemble classifier proves to be the most successful method for classifying point clouds. RFs solve the problem of overfitting the training data since each weak learner in the forest is trained with a different, randomly selected subset of the data. By doing so, RFs reduce bias and increase the robustness and diverseness of the classifier. Deep learning and RF classifiers accurately classify point clouds. The former train faster than other evaluated classifiers since they learn features directly from the training data. Despite their high classification accuracy, these supervised classification approaches (RF and Deep Learning) are difficult to interpret visually and have a time-consuming training phase. They also require a lot of labelled training data. This requirement is not very plausible for point cloud classification and especially not for Cultural Heritage Preservation. Point clouds of CH sites have a limited amount of labelled data available. In addition, CH sites are not consistent, with each comprising a different set of objects, ruins, buildings, etcetera.

Clustering is an unsupervised Machine Learning approach that does not require any labelled training data and is, therefore, an advantageous approach to the problem of point cloud classification. This paper has reviewed several clustering methods, including *k*-means, DBSCAN and Gaussian Mixture models and can conclude that such an approach would be appropriate for this project. Clustering algorithms are easy to understand, and their lack of a pre-processing step makes them simple to implement. The main disadvantage of clustering algorithms is that they cannot always find meaningful clusters in datasets with high-dimensional data or density discrepancies. In addition, *k*-means requires one to specify the number of partitions in advance, DBSCAN requires a large amount of memory to achieve accuracy, and Gaussian Mixture Models suffer from training noise and computational complexity. However, when these requirements are satisfied, these clustering methods can effectively cluster and classify point cloud data into semantically meaningful classes whilst discarding noise.

In conclusion, there are several approaches to point cloud classification, and it is possible to automate the process. The efficiency and accuracy of the classification task depend on the chosen classifier. One can conclude that the Random Forest classifier is the most accurate and successful supervised approach. An unsupervised approach such as Clustering is more appropriate for point cloud classification of CH sites where a limited amount of labelled training data is available.

REFERENCES

[1] Harith Aljumaily, Debra F Laefer, and Dolores Cuadra. 2017. Urban Point Cloud Mining Based on Density Clustering and MapReduce. *Journal of Computing in Civil Engineering* 31, 5 (2017), 04017021. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000674

[2] Maarten Bassier, Bjorn Van Genechten, and Maarten Vergauwen. 2019. Classification of Sensor Independent Point Cloud Data of Building Objects using Random Forests. *Journal of Building Engineering* 21 (2019), 468–477. https://doi.org/10.1016/j.jobe.2018.04.027

[3] David Belton, Simon Moncrieff, and Jane Chapman. 2013. Processing Tree Point Clouds using Gaussian Mixture Models. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* 2, 5 (November 2013), 43–48. https://doi.org/10.5194/isprsannals-II-5-W2-43-2013

[4] Pedro Domingos. 1995. Rule Induction and Instance-Based Learning: A Unified Approach. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence (IJCAI-95)*. Morgan Kaufmann, Montreal, Canada, 1226–1232.

[5] Junho Eum, Minho Bae, Junbeom Jeon, Heezin Lee, Sangyoon Oh, and Minsu Lee. 2017. Vehicle Detection from Airborne LiDAR Point Clouds Based on a Decision Tree Algorithm with Horizontal and Vertical Features. *Remote Sensing Letters* 8, 5 (2017), 409–418. https://doi.org/10.1080/2150704X.2016.1278310

[6] Absalom E Ezugwu, Amit K Shukla, Moyinoluwa B Agbaje, Olaide N Oyelade, Adán José-García, and Jeffery O Agushaka. 2021. Automatic Clustering Algorithms: A Systematic Review and Bibliometric Analysis of Relevant Literature. *Neural Computing and Applications* 33, 11 (2021), 6247–6306. https://doi.org/10.1007/s00521-020-05395-4

[7] E. Grilli, D. Dininno, G. Petrucci, and F. Remondino. 2018. From 2D to 3D Supervised Segmentation and Classification for Cultural Heritage Applications. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-2 (2018), 399–406. https://doi.org/10.5194/isprs-archives-XLII-2-399-2018

[8] Eleonora Grilli and Fabio Remondino. 2019. Classification of 3D Digital Heritage. *Remote Sensing* 11, 7 (2019), 1–23. https://www.mdpi.com/2072-4292/11/7/847

[9] William Koehrsen. 2017. *Random Forest Simple Explanation.* Retrieved April 22,2022 from https://williamkoehrsen.medium.com/random-forest-simple-explanation-377895a60d2d

[10] Hema Koppula, Abhishek Anand, Thorsten Joachims, and Ashutosh Saxena. 2011. Semantic Labeling of 3D Point Clouds for Indoor Scenes. In *Advances in Neural Information Processing Systems*, J. Shawe-Taylor, R. Zemel, P. Bartlett, F. Pereira, and K.Q. Weinberger (Eds.), Vol. 24. Curran Associates, Inc., 1–9. https://proceedings.neurips.cc/paper/2011/file/9872ed9fc22fc182d371c3e9ed316094-Paper.pdf

[11] Kevin Lai and Dieter Fox. 2010. Object Recognition in 3D Point Clouds Using Web Data and Domain Adaptation. *The International Journal of Robotics Research* 29, 8 (2010), 1019–1037. https://doi.org/10.1177/0278364910369190

[12] Artur Maligo and Simon Lacroix. 2017. Classification of Outdoor 3D Lidar Data Based on Unsupervised Gaussian Mixture Models. *IEEE Transactions on Automation Science and Engineering* 14, 1 (2017), 5–16. https://doi.org/10.1109/TASE.2016.2614923

[13] Patrick Marais, Matteo Dellepiane, Paolo Cignoni, and Roberto Scopigno. 2019. Semi-automated Cleaning of Laser Scanning Campaigns with Machine Learning. *ACM Journal on Computing and Cultural Heritage* 12, 3 (2019), 1–29. https://doi.org/10.1145/3292027

[14] Rickert Mulder. 2017. *Accelerating Point Cloud Cleaning.* Master's thesis. University of Cape Town.

[15] Rickert Mulder and Patrick Marais. 2016. Accelerating Point Cloud Cleaning. In *Proceedings of Eurographics Workshop on Graphics and Cultural Heritage.* Eurographics Association, Genoa, Italy, 1–4.

[16] Daniel Munoz, Nicolas Vandapel, and Martial Hebert. 2008. Directional Associative Markov Network for 3-D Point Cloud Classification. *Fourth international symposium on 3D data processing, visualization and transmission* 1, 1 (2008), 1–8.

[17] Christopher Pocock. 2019. *3D Scan Campaign Classification with Representative Training Scan Selection.* Master's thesis. Faculty of Science, Department of Computer Science.

[18] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. 2017. Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition (CVPR).* IEEE, San Juan, 652–660.

[19] Anandakumar M Ramiya, Rama Rao Nidamanuri, and Ramakrishnan Krishnan. 2016. Object-Oriented Semantic Labelling of Spectral–Spatial LiDAR Point Cloud for Urban Land Cover Classification and Buildings Detection. *Geocarto International* 31, 2 (2016), 121–139. https://doi.org/10.1080/10106049.2015.1034195

[20] Heinz Rüther, Christoph Held, Roshan Bhurtha, Ralph Schroeder, and Stephen Wessels. 2012. From Point Cloud to Textured Model, the Zamani Laser Scanning Pipeline in Heritage Documentation. *South African Journal of Geomatics* 1, 1 (Jan. 2012), 44–59.

[21] Martin Weinmann, Boris Jutzi, Stefan Hinz, and Clément Mallet. 2015. Semantic Point Cloud Interpretation Based on Optimal Neighborhoods, Relevant Features and Efficient Classifiers. *ISPRS Journal of Photogrammetry and Remote Sensing* 105 (2015), 286–304. https://doi.org/10.1016/j.isprsjprs.2015.01.016

[22] Daniel Wolf, Johann Prankl, and Markus Vincze. 2015. Fast Semantic Segmentation of 3D Point Clouds using a Dense CRF with Learned Parameters. In *2015 IEEE International Conference on Robotics and Automation (ICRA).* IEEE, Washington State Convention Center, Seattle, Washington, 4867–4873. https://doi.org/10.1109/ICRA.2015.7139875

[23] Bin Yang, Wenjie Luo, and Raquel Urtasun. 2018. PIXOR: Real-Time 3D Object Detection From Point Clouds. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE, Salt Lake City, Utah, 7652–7660.

[24] Min Zhang, Yifan Wang, Pranav Kadam, Shan Liu, and C.-C Jay Kuo. 2020. Pointhop++: A Lightweight Learning Model on Point Sets for 3D Classification. In *2020 IEEE International Conference on Image Processing (ICIP).* IEEE, Abu Dhabi, United Arab Emirates, 3319–3323. https://doi.org/10.1109/ICIP40778.2020.9190740

[25] Min Zhang, Haoxuan You, Pranav Kadam, Shan Liu, and C-C Jay Kuo. 2020. Pointhop: An Explainable Machine Learning Method for Point Cloud Classification. *IEEE Transactions on Multimedia* 22, 7 (2020), 1744–1755.

[26] Zhenxin Zhang, Liqiang Zhang, Xiaohua Tong, P Takis Mathiopoulos, Bo Guo, Xianfeng Huang, Zhen Wang, and Yuebin Wang. 2016. A Multilevel Point-Cluster-Based Discriminative Feature for ALS Point Cloud Classification. *IEEE Transactions on Geoscience and Remote Sensing* 54, 6 (2016), 3309–3321. https://doi.org/10.1109/TGRS.2016.2514508