Feature Extraction for Point Clouds

Jared May
University of Cape Town
Department of Computer Science
Cape Town, South Africa
myxjar002@myuct.ac.za

ABSTRACT

The widespread use of point clouds in multiple fields including Cultural Heritage and Computer Vision requires that point clouds be analysed and interpreted, however due to the vast difference in the applications, acquisition methods and quality of 3D point clouds this is a non-trivial exercise. Many approaches to deal with these challenges and efficiently process and analyse point clouds have been explored. Feature extraction is integral to methodologies to analyse point clouds. This literature review aims to explore feature extraction in the context of 3D Point clouds, and look at how features and feature extraction are used in modern applications.

KEYWORDS

Point-clouds, Feature extraction, Machine Learning, Deep Learning

1 INTRODUCTION

The acquisition and interpretation of 3D point cloud data has become an important topic in computer vision, robotics, photogrammetry, remote sensing and cultural heritage preservation. 3D point clouds are associated with 3D terrestrial, mobile and aerial laser scanning (TLS/MLS/ALS) and photogrammetry of physical objects and environments. Our ability to interpret point cloud data has led to advanced uses in computer vision and robotics, urban [5] and natural scene classification, modelling of city spaces [15], and the preservation of cultural heritage sites [30] to name a few.

The analysis of complex 3D scenes like those acquired by the Zamani Project [30] for cultural heritage preservation, and other similar campaigns remains a challenge. 3D point clouds data may be noisy, have varying point densities and contain other irregularities or complex surfaces remains. Thus much research has looked into automatic and semi-automatic approaches to manual and time consuming processes like point cloud cleaning [22], semantic segmentation and classification of 3D point clouds. This literature review will focus on one part of the pipeline generally followed to automatically and semi-automatically interpret point clouds - **feature extraction**.

Points in 3D point clouds are at their most basic (X, Y, Z) coordinates in euclidean space, but may contain other descriptive vectors like intensity or colour. We can describe point features by drawing comparisons and relationships between points or groups of points. We call these descriptions features. By extracting specific features of point clouds we are able to understand the scene without manually applying semantic labels to points in the point cloud. Feature extraction is an important part of the classification pipeline which generally has three steps that are taken to describe the 3D scene represented by the point cloud so that it can be classified. i) The recovery of a local neighbourhood for each 3D point, ii) The extraction of geometric or other features based on all 3D points within the local neighbourhood, and finally iii) the classification of all 3D points based on the respective features [32]. Much research has been conducted on point cloud feature extraction, and classical problems of: dealing with noisy data, varying point densities, neighbourhood selection, feature selection, and efficiency of feature extraction.

The documentation of Cultural Heritage (CH) monuments and sites with point clouds or meshes is differs widely from campaign to campaign. Given the recent evolution of technologies and digital tools, the need for automated and reliable methods to classify point clouds is of great importance. The analysis of point clouds, which includes feature extraction, as well as classification and clustering methodologies are being revisited. The development of Machine Learning (ML) and Deep Learning (DL) methodologies especially in the realm of point cloud analysis methodologies where feature extraction specifically is a key step in ML pipelines.

1.1 Background

1.1.1 Point Clouds: Point clouds are a 3D data type of unstructured points acquired from 3D scanning techniques like terrestrial and mobile laser scanning (TLS/MLS) and aerial photogrammetry. The popular use of point clouds has grown in recent years with multiple applications from Cultural Heritage, to Computer Vision. This growth has led to much research into point cloud interpretation.

Point clouds in their most basic form are a set of disjoint points, with each point in the set containing 3 co-ordinates (X, Y, Z) in 3 dimensional (3D) euclidean space. In some cases additional data like colour and intensity are added to points. Points in the set are disjoint and thus have no explicit connectivity or ordering between other points in the cloud. Points have no normal values for orientation and points in a point cloud may not distributed evenly in space, leading to varying point densities. These complexities - variance in point density, lack of oriented descriptors, as well as noise from data capture techniques like terrestrial laser scanning, can make analysing and interpreting point clouds a non-trivial task. Especially in comparison to 3D voxel and 2D pixel data.

Many methods have been developed to interpret point cloud data, however interpretation remains a challenge due to the fact that point clouds are a very primitive data type. They are unstructured sets of disjoint points which may have no distinct connectivity. This means that we must derive methods to determine vectors such as local surface normals points which are hard to adequately describe. There has especially been an uptick in research focusing on Machine Learning (ML) and Deep Learning approaches to point cloud segmentation, feature extraction, and classification.

1.1.2 Point Cloud Analysis: Point clouds need to be analysed as they are an important data type in multiple fields, including Cultural Heritage, Computer Graphics, Architecture, Robotics, and

Computer Vision. Many methods for their analysis and interpretation have been devised, and research continues in this field as new methods are investigated to provide ever more efficient, accurate and useful solutions [35]. One particular are of interest is the automation of point cloud analysis using machine learning and deep learning [4] methods. Point cloud classification - applying semantic labels to point cloud data - is one such area and due irregular sampling, noise, density and context is non-trivial.

A basic framework for point cloud analysis adapted from [32] can be described by - (i) selection of point neighbourhoods of optimal size, (ii) extraction of point features like geometric 3D or 2D features, (iii) strategic feature selection based on use case, and (iv) an application of a suitable classification or clustering scheme or mixture of both.

1.1.3 Features: A feature is a description of a data point or group data points in a data set. Feature extraction is the extraction of useful features from data sets such as text, images, 3D models, or point clouds. In 2D imagery this might be the colour or coordinates of a pixel or the descriptions of the arrangement of a group of pixels as a line, curve or object edge in an image.

In point clouds a feature is a descriptor of a point or group of points that gives a basic description of that portion of the data. Features extracted from point clouds, describe local or global features of a 3D scene described by a point cloud. On its own feature extraction is a powerful form of basic heuristic extraction as well as a basic step in more complex analyses of point clouds. Feature Extraction is an initial step in the processing of Point clouds for machine learning (ML) and deep learning pipelines as well as being traditionally used for 3D modelling meshing and surface reconstruction of point clouds.

1.2 Paper Overview

This literature review is broken down into 3 main sections, excluding the Discussion - Section 5 and Conclusion - Section 6

- Section 2 - Feature Extraction
- Section 3 - Applications of Point Features
- Section 4 - Machine and Deep Learning

2 FEATURE EXTRACTION

Feature Extraction in point clouds has multiple use cases from surface reconstruction for 3D models to intermediate feature sets for ML pipelines for semantic segmentation and classification. Features are are foundational to the classification of laser scanning campaign classification frameworks. It is therefore important to select features that are relevant to both the given data set and classification task. Feature extraction is the selection and labelling of core features of point cloud data, including singular points or point neighbourhoods.

2.1 Point Features

Point features are used in 3D modelling for surface and object reconstruction, or in cultural heritage preservation, computer vision and robotics for semantic segmentation and classification of 3D scenes. Point features can be analysed in a global or local context - we call these global and local features. Global features are useful for applications in meshing and 3D modelling, while local features or a mixture of both are useful in the segmentation and classification of point clouds especially those acquired by terrestrial laser scanning (TLS) which commonly have varying point densities, noise and different environments from scanning campaign to campaign. Features give context that help correct for these complexities, as for example TLS scans are relative to the plane of the scanner and point clouds may be made up of multiple scans, point clouds thus have no true normal - a local feature example is the local point or surface normal, which describes the direction of the vector perpendicular to a point or point neighbourhood's surface i.e. the normal value. Point features are generally derived from the essential information of points in a point cloud which is a vector coordinate $< X, Y, Z >$ in 3D space. As additional vectors like intensity and colour may not be included in the point cloud. However it is possible to extract meaningful features from point clouds, while some features use only a single point's information, most features are based the point and other nearby points i.e. the point neighbourhood. A common way to find neighbourhoods is to search for the k closest points using the k-nearest neighbour (kNN) algorithm. There are also spherical and cylindrical neighbourhoods, formed by all points found within a sphere or cylinder of a fixed radius r [34]. Depending on the density of the point cloud and the k or r parameter, the speed of neighbourhood extraction varies greatly[28].

2.2 Covariance and Geometric Features

In *Feature Extraction from Point Clouds* [11], early applications of feature extraction to conduct principle component analysis (PCA) for feature description from point clouds was investigated. These features called, covariance or eigen features for the classification of points, form the basis of many later works on feature extraction, analysis of local points and and point neighbourhoods, and feature selection. The aim of the paper was to present a novel method for feature extraction from point clouds. Their method (i) constructs a neighbour graph, (ii) analyses the neighbourhood of each point and estimates a classification of each point (iii) extracts neighbourhood and individual point features based on analysis.

In (ii) a set of neighbour points are gathered from the neighbour graph, the number of neighbour points analysed is dependent on factors such noise in the point cloud. From the set of neighbours two descriptors can be described the centre location c_i and the correlation matrix C_i. The eigenvectors $\{e_0, e_1, e_2\}$ of the correlation matrix together with the corresponding eigenvalues $\{\lambda_0, \lambda_1, \lambda_2\}$, where $\lambda_0 \leq \lambda_1 \leq \lambda_2$, define an important metric - the correlation ellipsoid - that adopts the general form of the neighbour points and is used to describe their features. From this, local surface features can be extracted such as local surface normals as well as other surface features like curvatures and creases. As mentioned these geometric, covariance, or eigen features are important to point feature extraction in many applications. Essentially covariance features describe the shape of point neighbourhoods. Examples of many covariance features can be seen in the table - 2, and are analyzed in detail by [32, 13, 27].

Geometric features are computed from a points 3D co-ordinates (X, Y, Z). For example, [28] a point's Z-value describes its height if the scanner is perpendicular to the XY plane while recording.

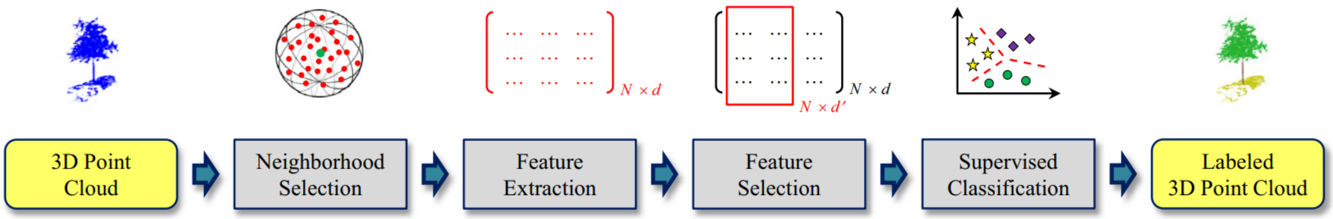

Figure 1: Common pipeline for point feature extraction and classification - Weinmann et al. 2015 [32]

The difference Δ between the smallest and largest Z, as well as the standard deviation σ of Z, in the point's local neighbourhood can help distinguish points belonging to short and tall objects. Other height-based features like vertical range Z_maxZ_min, height below ZZ_min and height above Z_maxZ [13] are extracted from cylindrical neighbourhoods to describe thin vertical objects well.

It should be noted that - X and Y co-ordinates from raw point clouds are not suitable features for classification as they are relative to the scanner's position, and can therefore vary depending on the orientation of individual scanners.

Additionally, features such as the radius of the sphere encapsulating a point's k-nearest neighbours, as well as the density of the neighbourhood, can also be extracted [32]. These are useful for differentiating, for example, scanner noise (large radius/low density) from valid points (small radius/high density)[28].

Name	Symbol	Definition
Covariance/shape features		
Verticality	V	$1-\lvert\langle[001],e_3\rangle\rvert$
Linearity	L_λ	$(\lambda_1-\lambda_2)/\lambda_1$
Planarity	P_λ	$(\lambda_2-\lambda_3)/\lambda_1$
Curvature	C_λ	$\lambda_3/(\lambda_1+\lambda_2+\lambda_3)$
Sphericity	S_λ	λ_3/λ_1
Omnivariance	O_λ	$\sqrt[3]{\lambda_1\cdot\lambda_2\cdot\lambda_3}$
Anisotropy	A_λ	$(\lambda_1-\lambda_3)/\lambda_1$
Eigenentropy	E_λ	$-\sum_{i=1}^{3}\lambda_i\cdot\ln(\lambda_i)$
1st Order Moment 1	M_1	$\sum_{i\in P}\langle P_i-p,e_1\rangle$
1st Order Moment 2	M_2	$\sum_{i\in P}\langle P_i-p,e_2\rangle$
2nd Order Moment 1	M_3	$\sum_{i\in P}\langle P_i-p,e_1\rangle^2$
2nd Order Moment 2	M_4	$\sum_{i\in P}\langle P_i-p,e_2\rangle^2$
Sum of EVs	$\Sigma_{\lambda 3D}$	$\lambda_1+\lambda_2+\lambda_3$
Sum of EVs (2D)	$\Sigma_{\lambda 2D}$	$\lambda_{2D_1}+\lambda_{2D_2}$
Ratio of EVs (2D)	$R_{\lambda 2D}$	$\lambda_{2D_2}/\lambda_{2D_1}$
Geometric features		
Radius	r_{3D}	$\mathrm{dist}(p,P_k)$
Density	D_{3D}	$(k+1)/(\frac{4}{3}\pi {r_{3D}}^3)$
Radius (2D)	r_{2D}	$\mathrm{dist}(p_{2D},P_{2D\,k})$
Density (2D)	D_{2D}	$(k+1)/(\pi {r_{2D}}^2)$
Height features		
Height difference	ΔH	$z_{\max}-z_{\min}$
Height std. deviation	σH	$\sqrt{\sum_{i=1}^{k}(z_i-\overline{z})^2/(k-1)}$
Vertical range (cylinder)	H_{range}	$z_{\max}-z_{\min}$
Height above (cylinder)	H_{above}	$z_{\max}-z$
Height below (cylinder)	H_{below}	$z-z_{\min}$

Figure 2: Extracted point feature names, symbols and definitions. γ_i and e_i are the ith eigenvalue and eigenvector derived from a covariance tensor of the points P in a point p's k-nearest neighbourhood. Features are extracted from 3D space unless stated 2D. [28]

2.3 2D Features

2D features are extracted from the projection of the point cloud onto the XY plane. These features are useful in scenes with symmetrical or orthogonal objects like buildings. When projected onto a plane, walls becomes a line which can be easily described by shape features. 3D geometric features described above such as radius and local point density can be adapted into features extracted from circular 2D neighbourhoods [32] and, covariance features can also be adapted to use the eigenvalues derived from the circular neighbourhood's covariance tensor [28].

2.4 Single-Scale vs. Multi-Scale Features

Given a 3D point p_i and its local neighbourhood, geometric features may be derived from the arrangement of the 3D points within the neighbourhood N. From these neighbourhoods we can sample and extract geometric relations such as distances, angles and angular variations between 3D points within the local neighbourhood. However, individual feature vectors of singular points or point neighbourhoods may not always be useful or they may provide too much or too little information and computational overhead in the process of feature extraction. Consequently, their have been investigations focusing on deriving useful features [32, 5]. When deriving features at a single scale, one has to consider that a suitable scale (in the form of either fixed or individual 3D neighbourhoods) is required in order to obtain an appropriate description of the local 3D structure - discussed below.

Alternatively, we may also derive features at multiple scales and subsequently involve a classifier in order to define which combination of scales allows the best separation of different classes [3]. In this context, features may even be extracted by considering different entities such as points and regions [36] or by involving a hierarchical segmentation. However, multi-scale approaches result in feature spaces of higher dimension, which is more computationally expensive making appropriate feature selection schemes even more important in order to gain predictive accuracy while at the same time reducing the extra computational burden in terms of both time and memory consumption [12].

2.5 Neighbourhood Selection

Neighbourhood selection is the selection and size of points around a given point p_i. Neighbourhood selection and feature extraction are connected issues, as the distinctiveness of geometric features strongly depends on the neighbourhood encapsulating those 3D points which are taken into consideration for feature extraction [32].

2.5.1 Defining Neighbourhoods: Describing the local 3D structure around a given 3D point p_i via geometric features, a neighbourhood definition that encapsulates all considered 3D points is required. Generally, different strategies are applied for defining the local neighbourhood N around a given 3D point p_i. Among these, the most commonly applied neighbourhood definitions are represented by:

- *A spherical neighbourhood definition* N_s: Where the neighbourhood is formed by all 3D points in a sphere of fixed radius $r_s \in \mathbb{R}$ around the point p_i [19].
- *A cylindrical neighbourhood definition* N_c: Where the neighbourhood is formed by all those 3D points whose 2D projections onto a plane (e.g. the ground plane) are within a circle of fixed radius $r_c \in \mathbb{R}$ around the projection of p_i [8].
- *A neighbourhood definition* N_k: Based on a fixed number of the $k \in \mathbb{N}$ nearest neighbours of p_i in 3D [20] or in 2D [26].

2.5.2 Fixed vs. Individual Neighbourhoods: Respective local neighbourhoods are usually defined by the definitions described above. The definition based on the k nearest neighbours offers greater flexibility with respect to the absolute neighbourhood size and is more adaptive to varying point density [35]. All these neighbourhood definitions, however, rely on a scale parameter, like either a radius r or k, which is commonly selected to be identical for all 3D points and determined via heuristic or empiric knowledge on the scene. As a result, the derived scale parameter is specific for each data set [35].

In order to obtain a solution taking into account that the selection of a scale parameter depends on the local 3D structure as well as local point density, an individual neighbourhood size can be determined for each 3D point. The most approaches rely on a neighbourhood consisting of the k-nearest neighbours and thus focus on optimising k for each individual 3D point. This optimisation may for instance be based on the local surface variation [27, 1], iterative schemes relating neighbourhood size to curvature, point density and noise of normal estimation [23, 16], dimensionality-based scale selection [7] or eigenentropy-based scale selection [34]. In particular, the latter two approaches have proven to be suitable for point cloud data acquired via mobile laser scanning, and a significant improvement of classification results can be observed in comparison to the use of fixed 3D neighbourhoods with identical scale parameter [34]. Alternatively, a different method of increasing context re-samples the neighbourhood's search space [27] in order to describe the local 3D structure across varying scales. In doing this, k remains fixed but the neighbourhoods capture a larger context as the search space becomes sparser.

2.6 Feature Selection

Point clouds differ from campaign to campaign, in noise level, and density. It is thus essential to select features that best describe the data set to minimise computational complexity and time when classifying 3D scenes.

To paraphrase Weinmann et. al [32]. The interpretation of a 3D scene described by a point cloud usually involves the following - (i) the selection of a local neighbourhood for each point, (ii) the extraction of geometric features based on all points in the local neighbourhood, and (iii) the classification of all 3D points based on their respective features. It is common practise to use as many features as possible in 3D scene analysis due to a lack of understanding of which features are most relevant or useful. Research addressing the selection of meaningful features as additional step between feature extraction and classification has been explored by in multiple texts [5, 21, 33, 34]. However, in all these examples, the challenge of dealing with the complexity of 3D scenes - irregular point sampling, varying point density and distinct differences across campaigns and objects, as well as, the computational complexity of dealing with large 3D point clouds and many available features has to be taken into account. In other words feature selection is important but non-trivial.

An example of a feature selection paper is [5] where the use of 21 LiDAR point cloud features split into 5 groups - height-based, echo-based, eigenvalue-based, plane-based and full-waveform - are explored for urban scene classification using a random forest classifier. A key focus of their research is the idea of relevant features. Here the authors devise methods to derive a minimal feature vector that includes the features that are most relevant to a specific class. Here they determine that in some classes certain feature sets notability the echo-based features are irrelevant but in other classifiers they maybe relevant.

3 APPLICATIONS OF POINT FEATURES

3.1 Classic Applications

3.1.1 3D modelling and surface reconstruction: Point clouds have been used for the building of 3D models and meshes for decades [11, 27] etc. geometric and linear features have been used to reconstruct 3D models from laser scanning campaigns, RGB-D data from depth imaging sensors like Microsoft Kinect and photogrammetry.

3.1.2 Semantic segmentation: Point clouds are widely used in cultural heritage preservation [30, 22], computer vision, engineering and surveying for semantic segmentation. i.e. for deriving meaningful data or labels from point clouds. This can be labelling building facades, or for cleaning erroneous data [22].

3.2 Machine Learning Pipeline

As mentioned there is a need for efficient processing and the automation of the processing of point cloud data in many contexts. One are is the classification of point clouds. Point features are used as input for the automated classification of point clouds. This may be as an input for semantic segmentation or for direct classification. Point cloud classification typically uses supervised learning to label groups of points in the cloud. Point cloud classification has become an active area of research and an important task in computer vision.

Various approaches with different machine learning methods at their core have been proposed these are discussed below. While the performance of these methods greatly depends on the problem's context and the quality of the training data, certain methods have been shown to perform better than others in most conditions leading to a rise in the popularity of ML in the analysis of point clouds. The basic pipeline can be seen in 1 above.

4 MACHINE LEARNING

Machine learning has become an important part of the processing of point clouds for analysis a task that without automation is a manual process [30, 22, 25]. Feature extraction is an important part of the pipeline for the automated analysis of of the point clouds in cultural heritage preservation and point features are used as a step in both supervised and unsupervised machine learning techniques as a step towards semantic segmentation, classification and clustering techniques.

Machine and Deep Learning (ML/DL) has allowed the development of algorithms that let machines take decisions based on a set of training data. Deep Learning can be considered an evolution of Machine Learning. Its algorithms are structured in layers to create an artificial neural network that can learn and make intelligent decision on its own. The use of Machine Learning techniques for point cloud classification has been successfully investigated in the last decade [9] looking at the built and natural environment by [26, 35, 4, 37], while in the Cultural Heritage (CH) field ML and DL methods have been explored by [10, 22, 13] to name a few.

4.1 Machine Learning Paradigms

The two main categories that machine learning algorithms typically fall into are supervised and unsupervised [24]. This distinction depends on whether the training data is labelled or not.

4.1.1 Supervised learning: With supervised learning, the algorithm has access to both the input data and their desired output labels. In the case of image recognition a supervised algorithm would learn from a set of images, with each image containing a label such as cat or dog. The supervised algorithm would then predict the class of unseen images after this training stage. This is an example of a binary classification since there are only two possible labels - this is different from multi-class classification where there are three or more possible labels [28].

4.1.2 Unsupervised learning: Unsupervised learning methods are not given labelled data sets. Instead are used to recognise patterns in unlabelled data. An example of this is feature learning where an algorithm discovers features needed to describe samples, or cluster analysis where unlabelled samples are assigned to groups according to some similarity metric [28].

4.1.3 Deep Learning: In recent years, deep learning methods have made major advances in solving difficult problems and have seen a significant increase in popularity [28]. Deep-learning [18] methods learn a representation of training data through multiple levels of transformation. Each level transforms the data into a higher and slightly more abstract level, obviating the need for feature engineering as features are learned from training data directly. While deep-learning methods have achieved state of the art results in many benchmarks, most implementations require very large amounts of training data and computation time.

4.2 Supervised Learning

4.2.1 Random Forests (RF):. Random forests [2] are an ensemble method that combine multiple decision trees into a powerful predictor. They classify input vectors by giving each tree in the forest the vector for classification, then compare the resulting outputs and using a majority voting mechanism determine the majority vote and the final output. When trees in the forest are trained, they are given the same parameters but use different training sets. These training sets are randomly selected - with replacement - from the original data set and are the same size as the original data set. Nodes are also split for further randomisation. Due to the randomisation and the averaging of loosely correlated decision trees, RFs have lower variance than single decision trees [28]. Random Forests are popular ad have been used in classification of point clouds for example [22] where they were used for keep/discard classification of point clouds.

4.2.2 Support Vector Machines (SVM):. Support Vector Machines (SVMs) were originally limited to non-probabilistic binary classification [6], and could therefore only distinguish between two classes, but several extensions have been made to SVMs to enable multi-class classification. SVMs treat training sample feature vectors as points in space that are mapped such that samples from different classes are as far away from each other as possible. They use a kernel function to map each training sample's feature vector into a higher dimensional space, and find hyper-planes that fit between the training samples. The goal is to find the hyper-plane that maximises the margin between the nearest feature vectors from each class - called support vectors. The support vectors are the only vectors that affect the position of the hyper-plane, and therefore define the decision function. When the SVM receives a test sample, it maps it to the hyper-space and classifies it based on which side of the max-margin hyper-plane it falls into [28]. Multi-class SVM classification approaches can be categorised as indirect or direct. The indirect approach involves training multiple SVMs, either in a one-against-rest or one-against-one manner [31]. Direct or all-together methods consider all classes at once, however their practical use is limited due to their high computational complexity [31].

4.2.3 Multi-Layer Perceptrons (MLP):. Artificial neural networks (ANN), are loosely based on the biological neural networks of brains [14], they are machine learning models formed by collections of connected nodes or neurons. These artificial neurons can receive signals, process them and then generate an output signal for connected neurons to use as input. A common ANN in the point cloud analysis is the Multi-Layer Perceptron (MLP). An MLP consists of one input layer, one output layer and one or more hidden layers. These layers contain one or more neurons that are connected to neurons in the previous and next layer[28]. In recent years, deep neural networks with many hidden layers have been used to learn representations from training data, creating new methodologies that challange traditional feature extraction, and classify point clouds [17, 4].

4.3 Unsupervised and Deep Learning

In Machine Learning it is typical to feed "regular" input data into convolutional neural networks. Since point clouds or meshes are irregular sets of disjoint points, they typically have been transformed into regular data types like 3D voxels arrays or 2D image stacks before feeding them into a neural network. These data transformations however are computationally costly, introduce artefacts that may degrade or obscure the intricacies of the data, and create data that is unnecessarily voluminous, requiring added memory or storage [4].

4.3.1 *Pointnet and Pointnet++:*

As mentioned typically - unstructured data types like 3D point clouds - are not suitable for direct input for ML/DL algorithms. However, Qi et al. [4], developed a deep net that does use point clouds as input resulting in what they termed *PointNets*. In their original work a novel deep neural network *PointNet* that directly consumes point cloud was proposed as well as theoretical analysis and visualisations of their network. Their paper provided a unified solution to multiple 3D point analysis issues: object classification, part segmentation and semantic segmentation. They showed that their deep net performed on par or better results than state of the art methods using benchmarks. Pointnet has three key modules: the max pooling layer as a symmetric function to aggregate information from all the points, a local and global information combination structure, and two joint alignment networks that align both input points and point features. PointNet++, [29] follows up on the groundwork of Pointnet. It is also a neural network architecture for processing point sets sampled in a metric space. However, PointNet++ recursively functions on a nested partitioning of the input point set, and improves on Poitnets handling of local and global features Pointnet++ and is effective in learning hierarchical features with respect to distance. Pointnet++ handles the issue of non-uniform point sampling, by using two novel set abstraction layers that intelligently aggregate multi-scale information according to local point densities. These contributions enable again state-of-the-art performance on challenging benchmarks of 3D point clouds.

5 DISCUSSION

5.1 Feature Extraction

Feature extraction is a vital step in the analysis of 3D point clouds, it is a broadly applicable and well researched topic.

5.1.1 New methodologies for feature extraction: Feature extraction still has room for research especially in deep and unsupervised learning. There are new applications of old methods and new problems to investigate. Machine Learning has room for investigation in terms of improving efficiency, accuracy, and computational complexity of 3D point cloud analysis. Unsupervised learning is an area where there is still space for experimentation especially in CH where there has not been much experimentation. The use of unsupervised methods like clustering could give useful insights.

5.2 Promising Methods and Future Work

5.2.1 Use of Pointnet and Pointnet++ for Feature Extraction: Pointnet and Pointnet++ are promising in their ability to process point clouds directly as well as the potential for their feature aggregation methods to be fed directly into unsupervised learning applications. This could be useful in investigations into clustering of CH datasets, with potential use in point cloud cleaning, and classification.

6 CONCLUSION

This literature review has explored point clouds, feature extraction, and classical and modern applications of features to the analysis of 3D point cloud data - including using machine and deep learning. It has shown the widespread use of point clouds and feature extraction in multiple fields and applications including Cultural Heritage, and semantic segmentation. It has been shown that the analysis of 3d point clouds is non-trivial due to the vast difference in the applications, acquisition methods and quality of 3D point clouds. There are many approaches to deal with these challenges and many methods to efficiently process and analyse point clouds have been explored. This literature review has show that feature extraction is integral to all methodologies in the analysis of point clouds, and that without it 3d point clouds are not very useful.

ACKNOWLEDGMENTS

Thanks to Supervisor: Prof. Patrick Marais and Co-Supervisor: Luc Hayward.

REFERENCES

[1] David Belton and Derek D Lichti. 2006. Classification and segmentation of terrestrial laser scanner point clouds using local variance information. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci* 36, 5 (2006), 44–49.
[2] Leo Breiman. 2001. Random forests. *Machine learning* 45, 1 (2001), 5–32.
[3] Nicolas Brodu and Dimitri Lague. 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 (2012), 121–134.
[4] R. Qi Charles, Hao Su, Mo Kaichun, and Leonidas J. Guibas. 2017. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, Los Alamitos, CA, USA, 77–85. https://doi.org/10.1109/CVPR.2017.16
[5] Nesrine Chehata, Li Guo, and Clément Mallet. 2009. Airborne lidar feature selection for urban classification using random forests. In *Laserscanning*. ISPRS, Paris, France, 207–212. https://hal.archives-ouvertes.fr/hal-02384719
[6] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine learning* 20, 3 (1995), 273–297.
[7] Jérôme Demantke, Clément Mallet, Nicolas David, and Bruno Vallet. 2011. DIMENSIONALITY BASED SCALE SELECTION IN 3D LIDAR POINT CLOUDS. In *Laserscanning*. Calgary, Canada. https://hal.archives-ouvertes.fr/hal-02384758
[8] Sagi Filin and Norbert Pfeifer. 2005. Neighborhood Systems for Airborne Laser Data. *Photogrammetric Engineering and Remote Sensing* 71, 6 (2005), 743–755. https://doi.org/doi:10.14358/PERS.71.6.743
[9] E. Grilli, E. Özdemir, and F. Remondino. 2019. APPLICATION OF MACHINE AND DEEP LEARNING STRATEGIES FOR THE CLASSIFICATION OF HERITAGE POINT CLOUDS. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-4/W18, 447–454. https://www.proquest.com/conference-papers-proceedings/application-machine-deep-learning-strategies/docview/2306826192/se-2
[10] Eleonora Grilli and Fabio Remondino. 2019. Classification of 3D Digital Heritage. *Remote Sensing* 11, 7 (2019). https://doi.org/10.3390/rs11070847
[11] Stefan Gumhold, Xinlong Wang, and Rob Macleod. 2001. Feature Extraction from Point Clouds. In *In Proceedings of the 10th International Meshing Roundtable*. 293–305.
[12] Isabelle Guyon and André Elisseeff. 2003. An introduction to variable and feature selection. *Journal of machine learning research* 3, Mar (2003), 1157–1182.
[13] T. Hackel, J. D. Wegner, and K. Schindler. 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 (2016), 177–184. https://doi.org/10.5194/isprs-annals-III-3-177-2016
[14] Simon Haykin and N Network. 2004. A comprehensive foundation. *Neural networks* 2, 2004 (2004), 41.

[15] Florent Lafarge and Clément Mallet. 2012. Creating Large-Scale City Models from 3D-Point Clouds: A Robust Approach with Hybrid Representation. *International journal of computer vision* 99, 1 (2012), 69–85.

[16] Jean-François Lalonde, Ranjith Unnikrishnan, Nicolas Vandapel, and Martial Hebert. 2005. Scale selection for classification of point-sampled 3D surfaces. In *Fifth International Conference on 3-D Digital Imaging and Modeling (3DIM'05)*. IEEE, 285–292.

[17] Loic Landrieu and Martin Simonovsky. 2018. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4558–4567.

[18] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep learning. *Nature* 521, 7553 (2015).

[19] I. Lee and T. Schenk. 2002. Perceptual Organization of 3D Surface Points. In *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. XXXIV-3A. Graz, Austria, 193–198.

[20] Lars Linsen and Hartmut Prautzsch. 2001. Local Versus Global Triangulations. In *Eurographics 2001 - Short Presentations*. Eurographics Association. https://doi.org/10.2312/egs.20011021

[21] Clément Mallet, Frédéric Bretar, Michel Roux, Uwe Soergel, and Christian Heipke. 2011. Relevance assessment of full-waveform lidar data for urban area classification. *ISPRS Journal of Photogrammetry and Remote Sensing* 66, 6, Supplement (2011), S71–S84. https://doi.org/10.1016/j.isprsjprs.2011.09.008 Advances in LIDAR Data Processing and Applications.

[22] Patrick Marais, Matteo Dellepiane, Paolo Cignoni, and Roberto Scopigno. 2019. Semi-Automated Cleaning of Laser Scanning Campaigns with Machine Learning. *ACM J. Comput. Cult. Herit.* 12, 3, Article 16 (June 2019), 29 pages. https://doi.org/10.1145/3292027

[23] Niloy J. Mitra and An Nguyen. 2003. Estimating Surface Normals in Noisy Point Cloud Data. In *Proceedings of the Nineteenth Annual Symposium on Computational Geometry* (San Diego, California, USA) *(SCG '03)*. Association for Computing Machinery, New York, NY, USA, 322–328. https://doi.org/10.1145/777792.777840

[24] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. 2012. *Foundations of Machine Learning*. The MIT Press. http://www.jstor.org/stable/j.ctt5hhcw1

[25] Rickert Mulder. 2017. *Accelerating Point Cloud Cleaning*. mathesis. University of Cape Town.

[26] Joachim Niemeyer, Franz Rottensteiner, and Uwe Soergel. 2014. Contextual classification of lidar data and building object detection in urban areas. *ISPRS Journal of Photogrammetry and Remote Sensing* 87 (2014), 152–165. https://doi.org/10.1016/j.isprsjprs.2013.11.001

[27] Mark Pauly, Richard Keiser, and Markus Gross. 2003. Multi-scale Feature Extraction on Point-Sampled Surfaces. *Computer Graphics Forum* 22, 3 (2003), 281–289. https://doi.org/10.1111/1467-8659.00675 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/1467-8659.00675

[28] Christopher Pocock. 2019. *3D Scan Campaign Classification with Representative Training Scan Selection*. mathesis. University of Cape Town.

[29] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. 2017. PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space. In *Advances in Neural Information Processing Systems*, I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Eds.), Vol. 30. Curran Associates, Inc. https://proceedings.neurips.cc/paper/2017/file/d8bf84be3800d12f74d8b05e9b89836f-Paper.pdf

[30] 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 (2012), 44–59.

[31] Zhe Wang and Xiangyang Xue. 2014. Multi-class support vector machine. In *Support vector machines applications*. Springer, 23–48.

[32] 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

[33] M. Weinmann, B. Jutzi, and Clément Mallet. 2013. Feature relevance assessment for the semantic interpretation of 3D point cloud data. In *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. II-5/W2. Antalya, Turkey, 313–318. https://doi.org/10.5194/isprsannals-II-5-W2-313-2013

[34] M. Weinmann, B. Jutzi, and Clément Mallet. 2014. Semantic 3D scene interpretation: A framework combining optimal neighborhood size selection with relevant features. In *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, Vol. II-3. Zürich, Switzerland, 181–188. https://doi.org/10.5194/isprsannals-II-3-181-2014

[35] M. Weinmann, A. Schmidt, C. Mallet, S. Hinz, F. Rottensteiner, and B. Jutzi. 2015. CONTEXTUAL CLASSIFICATION OF POINT CLOUD DATA BY EXPLOITING INDIVIDUAL 3D NEIGBOURHOODS. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* II, 3 (2015), 271–278. https://www.proquest.com/scholarly-journals/contextual-classification-point-cloud-data/docview/1756968233/se-2 Copyright - Copyright Copernicus GmbH 2015; Last updated - 2016-01-16.

[36] Xuehan Xiong, Daniel Munoz, J Andrew Bagnell, and Martial Hebert. 2011. 3-d scene analysis via sequenced predictions over points and regions. In *2011 IEEE International Conference on Robotics and Automation*. IEEE, 2609–2616.

[37] E. Özdemir and F. Remondino. 2019. CLASSIFICATION OF AERIAL POINT CLOUDS WITH DEEP LEARNING. *The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-2/W13, 103–110. https://www.proquest.com/conference-papers-proceedings/classification-aerial-point-clouds-with-deep/docview/2585583873/se-2