4-2015

Volume Component Analysis for Classification of LiDAR Data

Nina M. Varney
University of Dayton

Vijayan K. Asari
University of Dayton, vasari1@udayton.edu

Volume component analysis for classification of LiDAR data

Nina M. Varney and Vijayan K. Asari[a]

[a]University of Dayton, 300 College Park, Dayton, OH, USA

ABSTRACT

One of the most difficult challenges of working with LiDAR data is the large amount of data points that are produced. Analysing these large data sets is an extremely time consuming process. For this reason, automatic perception of LiDAR scenes is a growing area of research. Currently, most LiDAR feature extraction relies on geometrical features specific to the point cloud of interest. These geometrical features are scene-specific, and often rely on the scale and orientation of the object for classification. This paper proposes a robust method for reduced dimensionality feature extraction of 3D objects using a volume component analysis (VCA) approach.[1]

This VCA approach is based on principal component analysis (PCA). PCA is a method of reduced feature extraction that computes a covariance matrix from the original input vector. The eigenvectors corresponding to the largest eigenvalues of the covariance matrix are used to describe an image. Block-based PCA is an adapted method for feature extraction in facial images because PCA, when performed in local areas of the image, can extract more significant features than can be extracted when the entire image is considered. The image space is split into several of these blocks, and PCA is computed individually for each block.

This VCA proposes that a LiDAR point cloud can be represented as a series of voxels whose values correspond to the point density within that relative location. From this voxelized space, block-based PCA is used to analyze sections of the space where the sections, when combined, will represent features of the entire 3-D object. These features are then used as the input to a support vector machine which is trained to identify four classes of objects, vegetation, vehicles, buildings and barriers with an overall accuracy of 93.8%

Keywords: LiDAR, automatic perception, scene understanding, VCA, volume component analysis

1. INTRODUCTION

LiDAR is one of the fastest growing technologies for creating 3D models. Applications include surveying, remote sensing and surveillance. One of the many challenges of LiDAR data, especially aerial LiDAR, is the task of scene understanding.[2–5] Because aerial LiDAR is able to create dense 3D scenes, the result is a large amount of points which requires extensive time for analysts to sort through and recognize targets or objects of interest. Because of this challenge one of the main tasks in LiDAR processing is the task of automatic scene understanding. Scene understanding can be thought of in two basic tasks, the first being individual object segmentation and the second being a classification of those objects into meaningful categories.

In our particular application we identify the need to separate objects into 5 separate subclasses; ground, vegetation, buildings, vehicles and power lines.[6,7] By separating the objects into these basic subclasses, we can provide the end user with immediate contextual information about the scenes, significantly reducing the amount of time that it takes to both analyze the scene and identify objects of interest.

In order to perform the automatic perception on the scene we break the process down into several steps. The first of these steps is a ground estimation, next is an octree segmentation, followed by a volume component analysis(VCA) for feature extraction, preformed on each individual segment, lastly a cascade of support vector machine (SVM) classifiers is used to sort the segments into our five previously identified classes. Overall the proposed method is robust, correctly identifying an aerial urban LiDAR scene with an accuracy of 93.8%

2. METHODOLOGY

The proposed method can be broken down into four processes, a RANSAC-based ground estimation, octree segmentation, VCA feature extraction and SVM cascade of classifiers.
The overall scheme is shown in Fig. 1.

Optical Pattern Recognition XXVI, edited by David Casasent, Mohammad S. Alam,
Proc. of SPIE Vol. 9477, 94770F · © 2015 SPIE · CCC code: 0277-786X/15/$18
doi: 10.1117/12.2179268

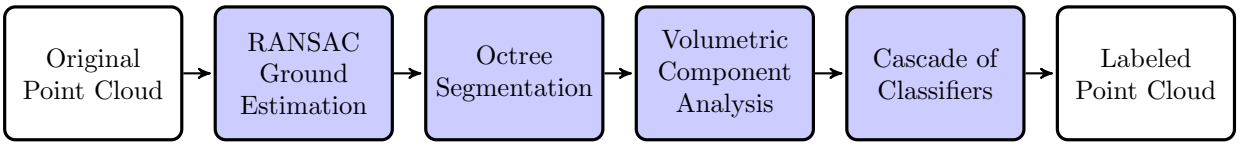

Figure 1: Point cloud classification methodology

2.1 RANSAC Ground Estimation

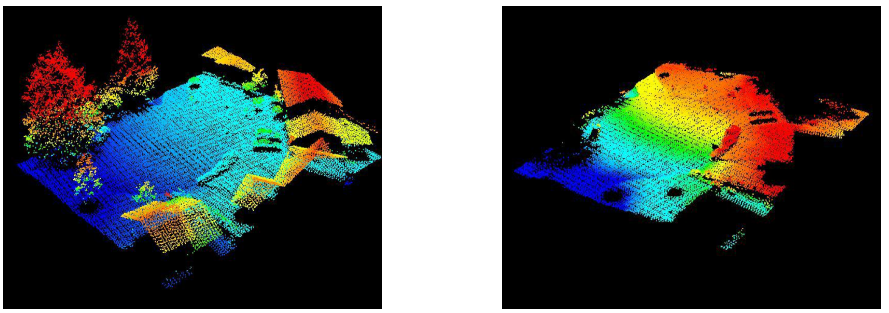

Figure 2: RANSAC Ground Estimation

The first step is identifying the ground points in the LiDAR model.[8–10] In order to do this we use a RANSAC method to estimate the ground plane.[11] In any given aerial scene, the ground points make up the majority of points. Because of this, we can think of the ground points as inliers, lying on the same plane and the object points as outliers, varying in distance and density from the ground plane. Our goal is to use RANSAC to identify the points lying on this underlying plane.

First, the scene is clustered into smaller subsections. This will allow us to capture subtle variations in the ground plane. The size of these sub-sections varies, but a good rule of thumb is to choose a subsection which is about two times larger than the largest object you are trying to identify. Next, each of these subsections is considered individually and a RANSAC plane estimation is performed.

As per the usual RANSAC method, a randomly chosen subset of points is selected and the model parameters are calculated. The model of the plane is described by a system of linear equations, shown in Eq. 1, where x^T is a set of N points, randomly selected for each iteration and θ is the parameter vector which instantiates the plane containing the points $x(1), ...x(N)$, shown in Eq. 2.

$$\begin{cases} \theta_1 x_1^{(1)} + \theta_2 x_2^{(1)} + \theta_3 x_3^{(1)} + \theta_4 \\ \vdots \\ \theta_1 x_1^{(N)} + \theta_2 x_2^{(N)} + \theta_3 x_3^{(N)} + \theta_4 \end{cases} \tag{1}$$

$$\begin{vmatrix} (x^(1))^T & 1 \\ \vdots & \vdots \\ (x^(N))^T & 1 \end{vmatrix} \theta = A\theta = 0 \tag{2}$$

θ can be solved using singular value decomposition. Once the model of the plane, M, is identified, we can estimate the error using Eq. 4.

$$M(\theta) = x \in R^3 : [x^T 1]\theta = 0 \tag{3}$$

$$e(x, M(\theta))^2 = \frac{([x^T 1]\theta)^2}{||\theta_(1:3)||^2} \tag{4}$$

This process is repeated, with another randomly chosen subset of points and the model which contains the smallest error will be used. We then calculate each points distance from the chosen model and identify

outliers, or object points, based on a distance threshold from each point to the plane. An example of a scene and its identified ground plane can be seen in Fig. 2.

2.2 Octree Segmentation

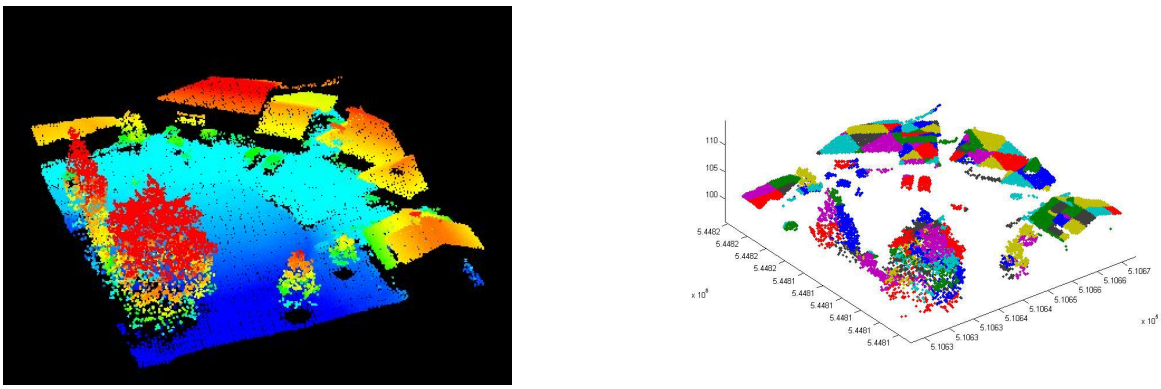

Figure 3: Octree Segmentation Results

After the ground points are identified we consider only the object points and perform an octree segmentation.[12,13] The process is as follows;

First, for each point we estimate a surface normal. A radius around the point is chosen and we examine all the points within that radius. Next we use principal component analysis to estimate a plane which describes that set of points. From this plane we can identify a surface normal which sits perpendicular to the plane. This surface normal is assigned to the point and this process is repeated for each point.

After the surface normal calculation, we perform an octree based segmentation. The entire point cloud area is split into 8 equally spaced bins, $2 \times 2 \times 2$. Next we check the surface normals within each bin. If the variance of the surface normals in each bin is significantly low we stop the segmentation. If the variance of the surface normals is above a certain threshold then we split the bin into 8 additional bins. This process is repeated until each bin contains only points with uniform surface normals. A point cloud and its equivalent octree segmentation are shown in Fig. 3, where each individual color represents a different bin.

2.3 Volume Component Analysis Feature Extraction

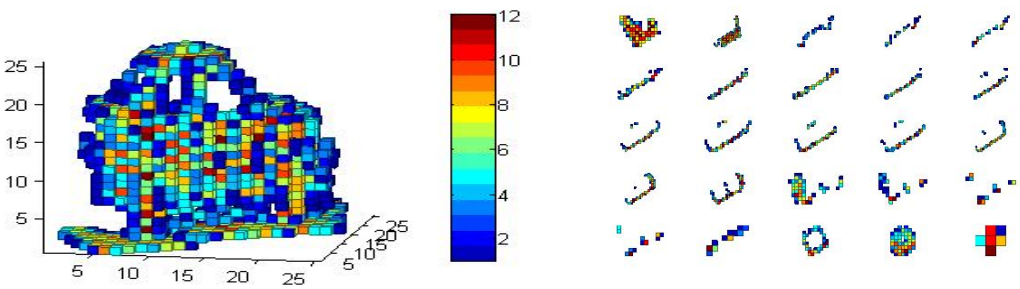

Figure 4: Volume component layers

After the initial segmentation, a VCA feature extraction is performed. Each segmented object is considered individually, the object is normalized into a $25 \times 25 \times 25$ voxel space, this can be represented by a matrix where each cell represents the number of points contained within the cell. Each vertical layer of the matrix is considered and PCA is performed on each layer.[14,15]

The mean of each slice is computed and subtracted from the layer, a covariance matrix, H, is constructed

according to Eq. 5 and eigen analysis is performed, from this eigen analysis, weights are computed and a set of weights from each layer are concatenated together to form the feature vector. The feature vectors are then normalized.

$$H = (X - \dot{X})(X - \dot{X})^T \tag{5}$$

2.4 SVM Classification

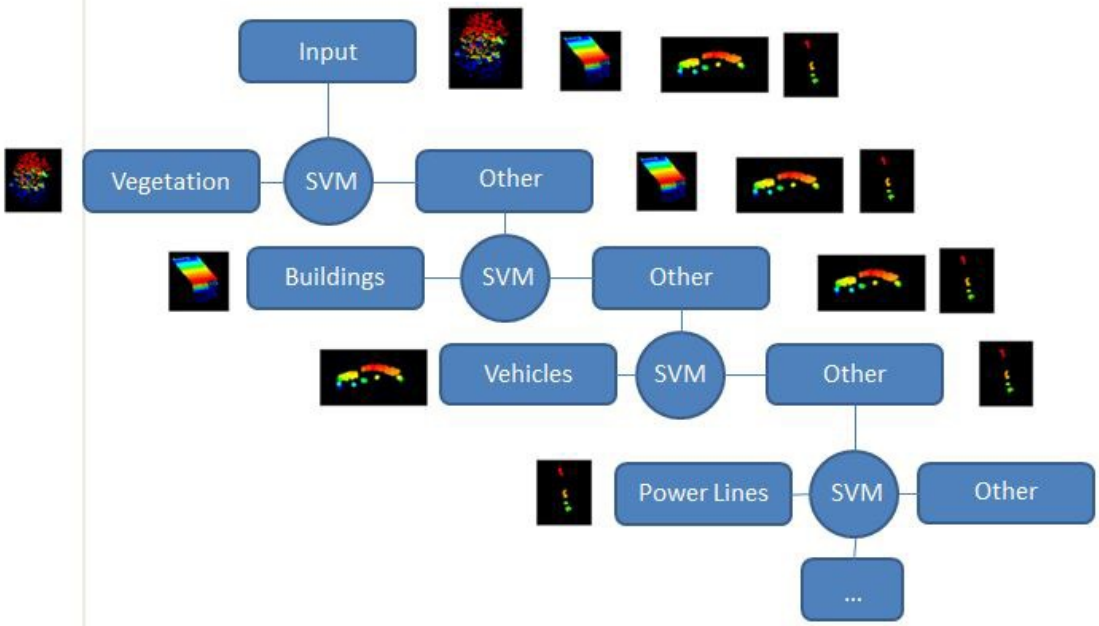

Figure 5: Cascade of Classifiers

Once the feature vectors are calculated they are used as inputs to a support vector machine.[16,17] The support vector machines are set up in a cascade of classifiers approach. Four linear SVMs are trained for the vegetation, buildings, vehicle and power-line classes. The feature vectors corresponding to each object are used as inputs to the vegetation classifier. All vectors which are not identified as vegetation are then used as inputs into the building SVM, this process is repeated until all SVM's have been applied. Those feature vectors which have not been classified by the four SVM's are identified as objects outside the four classes. The overall classifier scheme can be seen in Fig. 5.

3. EXPERIMENTAL RESULTS AND DISCUSSION

The performance of the proposed feature extraction technique has been evaluated on a set of segmented and voxelized point cloud objects in a subset of the aerial LiDAR data from Surrey, British Columbia, which was made available through the Open Data Program. The point cloud resolution is 15.6 pts/m^2. The proposed method was examined on 11 scenes, ranging from 59,740 to 3,705,537 total points. To perform RANSAC ground estimation, the subset size was chosen to be 30,000 points, with a distance threshold of 0.3 meters. Octree segmentation was performed and we examined bins with a variance less than 0.3. Volume component analysis was performed and the size of the normalized voxel space was chosen to be $25 \times 25 \times 25$

The results were achieved using MATLAB on a standard desktop computer, with a 2GHz processor and 12 GB of RAM. Processing time for the largest point cloud, containing 3,705,537 points, and measuring 98.9

Table 1: Accuracy

	Vegetation	Buildings	Vehicles	Power lines
Accuracy	**96.2**	**92.1**	**95.2**	**92.0**
Accuracy[4]	91.8	91.5	92.0	84.2

MB and spanning approximately $500m^2$, was 128 minutes.

In order to measure the effectiveness of the proposed method we used the measure of accuracy. Accuracy is defined as the ratio of correctly predicted positive and negative cases over the total number of predictions. Table 1 shows the accuracy of our proposed method over the method discussed in.[4]

Additionally, Fig 6 and 7 show the classification results of several sub-scenes within the data set. Once the points were identified, we assigned them RGB color codes according to their class, for the ground information the color was scaled based on the intensity from the original LiDAR data, for additional contextual information. In the scenes below, ground is represented in brown, buildings are in purple, vehicles are in cyan, vegetation is in green and power lines are shown in pink.

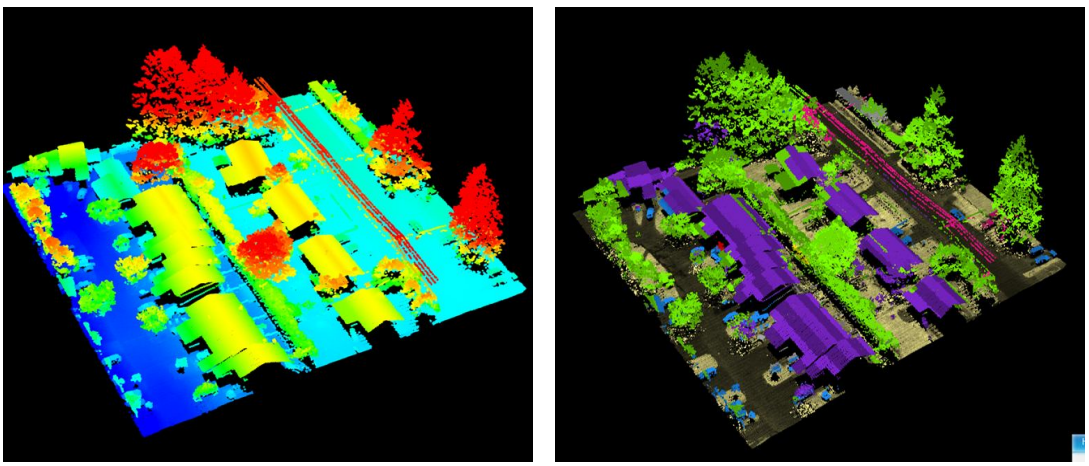

Figure 6: Scene 1 and its resulting classification

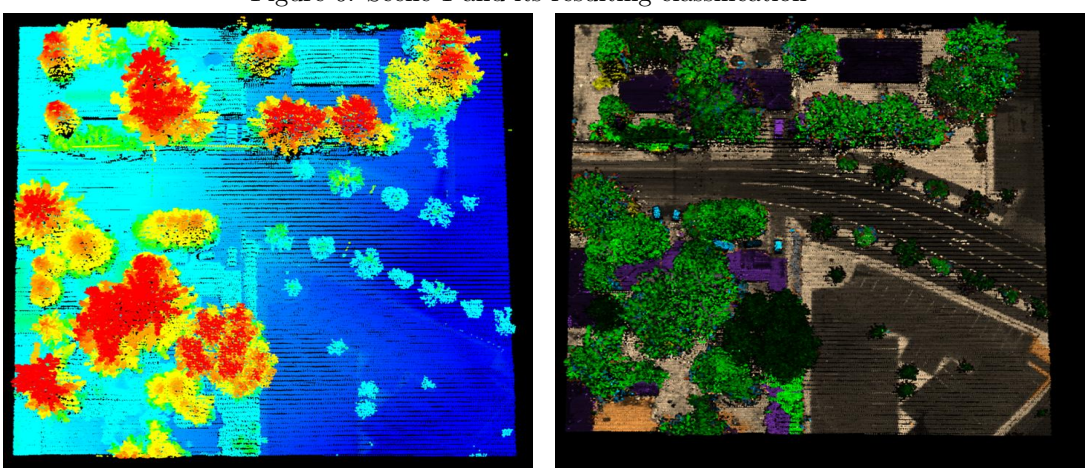

Figure 7: Scene 2 and its resulting classification

4. CONCLUSION

We proposed a method of automatically extracting volume-based features using our novel volume component analysis (VCA) approach. The proposed method was found to have an overall accuracy of 93.87%. Overall,

the results provide an increased level of scene understanding and context which will help analysts when identifying objects of interest, in aerial LiDAR data. Future work includes adding additional levels of scene understanding, specifically the precise object detection and identifying different object types within each class, with a focus on vehicles.

REFERENCES

1. N. Varney and V. Asari, "Volume component analysis for feature extraction in lidar point cloud objects," in *Imaging Systems and Applications*, pp. JTu2C–5, Optical Society of America, 2014.
2. M. Dalponte, L. Bruzzone, and D. Gianelle, "Fusion of hyperspectral and lidar remote sensing data for classification of complex forest areas," *Geoscience and Remote Sensing, IEEE Transactions on* **46**(5), pp. 1416–1427, 2008.
3. H. Li, H. Gu, Y. Han, and J. Yang, "Fusion of high-resolution aerial imagery and lidar data for object-oriented urban land-cover classification based on svm," *Proceedings of the ISPRS Working Group IV/1:Dynamic and Multi-dimensional GIS* , pp. 179–184, 2007.
4. J. Zhang, X. Lin, and X. Ning, "Svm-based classification of segmented airborne lidar point clouds in urban areas," *Remote Sensing* **5**(8), pp. 3749–3775, 2013.
5. T. Korah, S. Medasani, and Y. Owechko, "Strip histogram grid for efficient lidar segmentation from urban environments," in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on*, pp. 74–81, IEEE, 2011.
6. J. Holmgren, Å. Persson, and U. Söderman, "Species identification of individual trees by combining high resolution lidar data with multi-spectral images," *International Journal of Remote Sensing* **29**(5), pp. 1537–1552, 2008.
7. G. Miliaresis and N. Kokkas, "Segmentation and object-based classification for the extraction of the building class from lidar dems," *Computers & Geosciences* **33**(8), pp. 1076–1087, 2007.
8. M. Bartels, H. Wei, and D. C. Mason, "Dtm generation from lidar data using skewness balancing," in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, **1**, pp. 566–569, IEEE, 2006.
9. M. Bartels and H. Wei, "Segmentation of lidar data using measures of distribution," *International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences* **36**(7), pp. 426–31, 2006.
10. M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM* **24**(6), pp. 381–395, 1981.
11. N. Varney and V. Asari, "Volumetric features for object region classification in 3d lidar point clouds," in *Applied Imagery Pattern Recognition*, IEEE, 2014.
12. B. Douillard, J. Underwood, N. Kuntz, V. Vlaskine, A. Quadros, P. Morton, and A. Frenkel, "On the segmentation of 3d lidar point clouds," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pp. 2798–2805, IEEE, 2011.
13. K. L. Damkjer and H. Foroosh, "Mesh-free sparse representation of multidimensional lidar data,"
14. J. Yang, D. Zhang, A. F. Frangi, and J.-y. Yang, "Two-dimensional pca: a new approach to appearance-based face representation and recognition," *Pattern Analysis and Machine Intelligence, IEEE Transactions on* **26**(1), pp. 131–137, 2004.
15. L. Wang, X. Wang, X. Zhang, and J. Feng, "The equivalence of two-dimensional pca to line-based pca," *Pattern Recognition Letters* **26**(1), pp. 57–60, 2005.
16. J. A. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural processing letters* **9**(3), pp. 293–300, 1999.
17. J. A. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, J. Suykens, and T. Van Gestel, *Least squares support vector machines*, vol. 4, World Scientific, 2002.