University of Lethbridge Department of Mathematics and Computer Science

CPSC 4990 – Point Cloud Segmentation using Classification and Clustering
Fall2019

Justin Petluk

December 16, 2019

1 Introduction

Point clouds are Points in 3D Space, usually in conjunction with other information such as colour and/or surface normals. This information is often used to recreate objects and can be captured using Light Detection and Ranging (LIDAR) and RGB-D depth sensors. There are many uses for point clouds, and being able to analyze what's contained in them is important. In complex environments or within complex objects, methods of classification and clustering have been designed to automate the process of defining what components exist within the Point Clouds. This is known as segmentation. Here, we explore two types of neural networks are used for segmentation, Pointnet++ (or Pointnet2) [3] and PointCNN [2]. Two clustering-based segmentation techniques are also explored. These are known as Constrained Planar Cuts [5] segmentation (CPC) and Region Growing segmentation [6].

2 Methods

2.1 Collecting Data

In this study, point cloud data of a taxidermy raccoon was captured using two Intel Realsense D415 RGB-D depth cameras 90 centimetres apart, captured outdoors. The data has been captured every 30 centimetres, and only data up to 3 meters was used due to camera accuracy. This process was done for both the left side and right side of the raccoon. A total of 86 usable scans were able to be used for the dataset. This dataset thus consists of two capture angles on two perspectives along a range of 3 meters providing as much variance for the static target as possible while maintaining the ability to distinguish body parts for labelling. As the data was captured, Librealsense [1], the software provided for camera control, allowed spatial and temporal post-processing. The

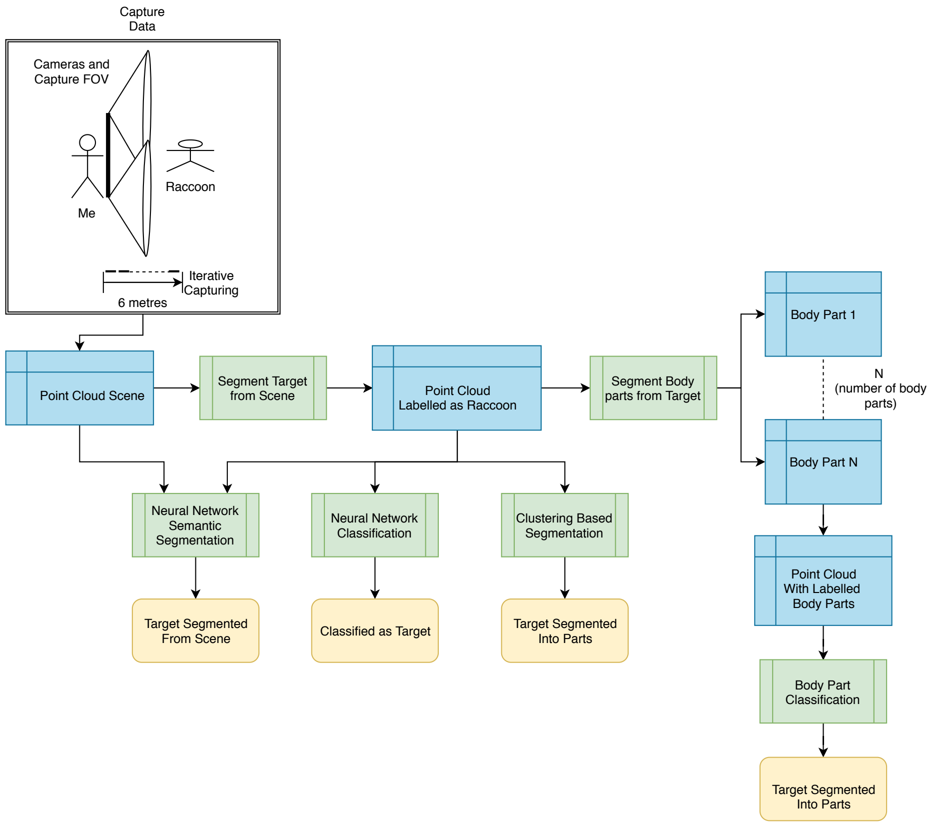

Figure 1: The workflow of data collection for various segmentation techniques.

default spatial filter was applied, improving the edge preservation of the data. The default temporal filtering was applied improving the smoothness of data over several frames. Finally, a 2x2 decimation filter was applied, lowering the resolution and averaging the values.

2.2 Preparing the Data

The data was then labelled using Cloud Compare for the ease of interface and use of the segmentation tool. In all of the scenes, the taxidermy raccoon was outlined and essentially cut out from the background and exported into a point cloud. The name of each file was done in such a way it could be referenced from the scene it was extracted from. At this stage, the data can be used for semantic segmentation, the identification of the target within a scene. This data can also be used for evaluating the performance of part segmentation with CPC and Region Growing cluster-based segmentation algorithms. Once this was completed, the next step was aimed for part segmentation, the classification of user-defined parts of an object. The raccoon's body parts labelled were the head including the neck and ears, the arms including the shoulders on the left perspective and only to the elbow on the right, the tail, and body including the thighs and legs. The labels were the filename thus allowing the part ID of all points within each point cloud, and a folder described the same as the original data. All the files within the folders were then grouped into one file containing all parts, seemingly reversing the manual segmentation process, except now each point contains an ID.

When the dataset was being prepared for classification, only the three dimensions of the point cloud data were extracted, ignoring the RGB information for each point. For Pointnet++ part segmentation, the point normals were calculated using Point Cloud Library's Surface Normal Estimation method. Specifically, the data contained the third dimension direction of the normals and not the magnitude. The dataset was split 66 percent training and 33 percent testing for classification.

The clustering dataset was identical to the dataset immediately before the segmented data. This contained the entire Raccoon with colours. CPC leveraged the use of colours in its clustering methods, while Region Growing did not.

3 Evaluation

3.1 Evaluation Methods

Both classification and clustering were evaluated on the same dataset, but in different ways. The two classification methods, Pointnet++ and PointCNN were evaluated by calculating the mean accuracy of labelled body parts. This was measured by calculating the average number of points that are correctly predicted in each raccoon of the entire dataset, for each epoch/iteration. Pointnet++ evaluates over epochs while PointCNN evaluates over iterations, each with a batch size of 8. Region growing and CPC clustering-based segmentation methods were evaluated using the labelled raccoon segments as ground truth. Specifically, homogeneity, completeness and a v-measure score were used [4]. Homogeneity measures how well the cluster points exist as members within the class, and completeness is how well every cluster point exists in a class. V-Measure is the relation of both homogeneity and completeness.

3.2 Clustering Parameters

It is important to note the parameters used during the evaluation for each of the methods. CPC segmentation weighted one part to colour, and three parts to the spatial components of the points. Region growing used a smoothness threshold of 2.5, and a curvature threshold of 1.0. These determine the sensitivity of where the clusters should merge or not.

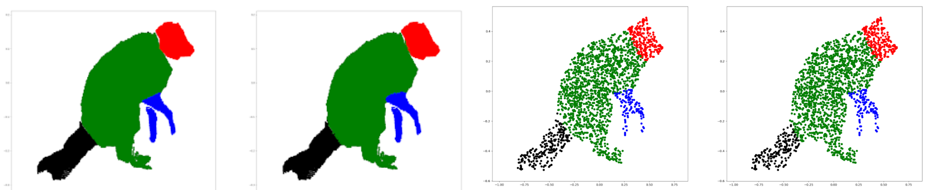

Figure 2: Visaul accuracy of both networks. The left pair is the result of PointCNN, and the right pair is the result of Pointnet++. The left of each pair shows the actual labels, and the right of each pair is the predicted labels.

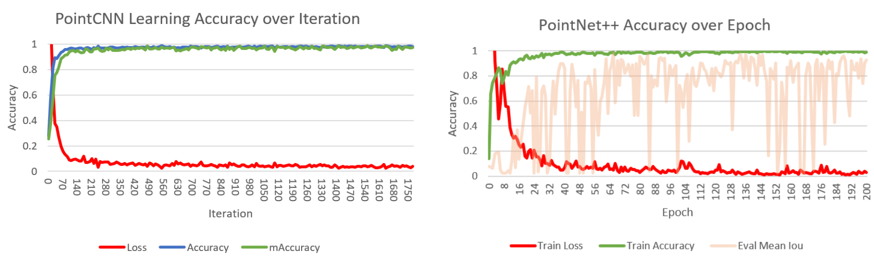

Figure 3: The accuracy curves for both networks over iteration and epoch for respective networks. PointCNN does not contain the mean evaluation of the curve, the reason being how it output the information during the process.

3.3 Classification Results

Both Pointnet++ and PointCNN performed quite well given the dataset at hand as each was able to reach 98 percent accuracy while learning as seen in Figure 3, meaning that it was able to learn from the dataset. Both learning curves for PointCNN and Pointnet++ are similar, though PointCNN seems to reach a more stable learning accuracy earlier in the process. Looking at the performance of Pointnet++, it either does quite well or simply not correct. PointCNN, on the other hand, does not experience this drop as PointNet++ does as seen in Figure 3. When testing the results, it seemed that PointCNN succeeded in labelling the edges of each body part and did not underestimate the edges as severely as Pointnet++.

3.4 Clustering Results

The results of clustering with Region Growing and CPC was not comparable to the classification results. The curves of the V-Measure scores show that Region growing has a more consistent success of the homogeneity and completeness, while CPC has more success with completeness throughout all of the samples as shown in Figure 4. Figure 5 shows how well the two compete, and where one is more successful. Region Growing again shows a more stable ac-

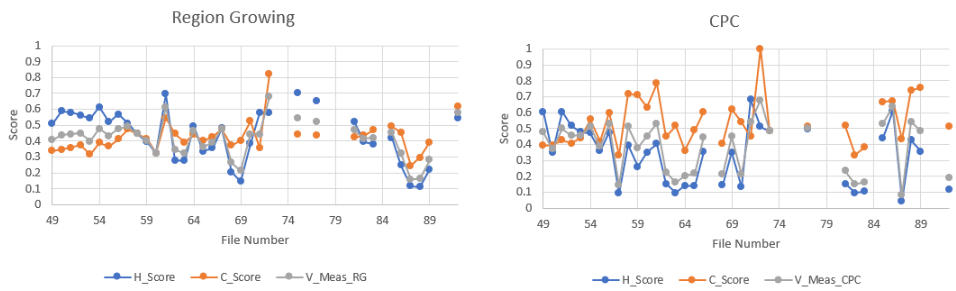

Figure 4: The score curves for Homogeneity, Completeness and V-Measure for both Region Growing and CPC.

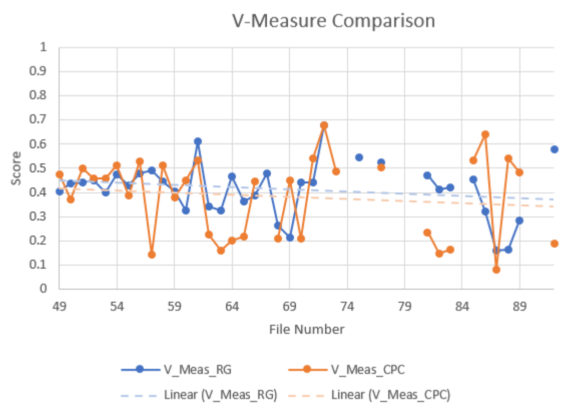

Figure 5: A comparison between Region Growing and CPC V-Measure scores, and their linear trends.

curacy when compared to Region Growing in Figure 5, and maintains a better score when CPC cannot, resulting in a slightly better trend than CPC, both with a score of around 0.4.

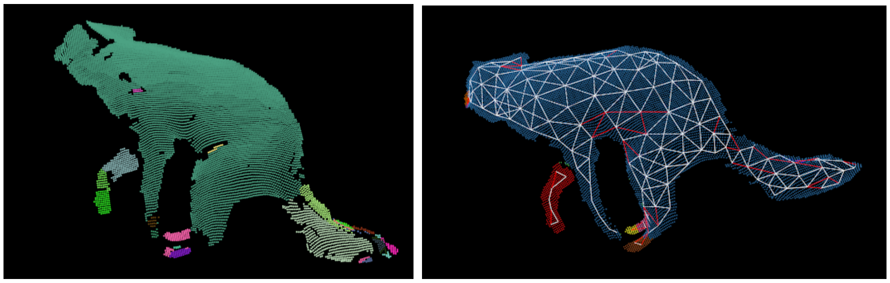

Figure 6: Visuals of Region Growing and CPC segmentation methods. The image on the left is Region Growing, and the image on the right is CPC.

4 Conclusion

4.1 Classification

When analyzing the quality of labelling, the ability to label along the edges of each of the classes seemed to prove the largest source of error. This could be due to the fitness of the training data to the testing data, as well as user label errors. It shows in some samples of PointCNN where the edge along the arm was either straight or curved, though this resulted in a small margin of error (around 3-5 percent) displays the need for consistency during labelling. This reflects the natural variability in the selected labels, and not directly on the classification itself. The most extreme sources of error occurred most often as over-estimating the body label, and under-estimating either the tail or arms. Two things that were different during the study were Pointnet++ had Farthest Point Sampling in the results which were inconsistent during evaluation and Pointnet++ learned the point normals while PointCNN did not. To get a better representation of the results, this should have been consistent. It would have also been nice to directly compare the evaluation accuracy directly side-by-side over the training period, but this was not implemented with PointCNN so this was not completed.

4.2 Clustering

Clustering found some success when compared to the labelled body parts. One of the driving factors for success was the ability to set the parameters of the different methods to suit the dataset. As seen in both of the curves the initial few samples performed better than the later ones as they became more dissimilar from the original. A possible reason for this is how the parameters are set for the best possible segmentation on the first image and maintained for all of the rest. This reflects the nature of a large group of clustering methods and shows to continue to be a hurdle. Nonetheless, it could be possible for these clustering methods may be more successful in more accurate datasets with higher resolution.

4.3 Future Work

Classification seemed very promising for part segmentation given some labels, specifically PointCNN. Both classification networks are capable of semantic segmentation of a scene, labelling an entire target within a scene which is an important step to getting to part segmentation. A larger dataset that contains many different animal poses would always be valuable in testing the robustness of the methods which means expanding to targets which and change pose. It would also be useful to test how training an entire model at once or even artificial models can perform with captured data using Pointnet++ and PointCNN.

Clustering could demonstrate better results in a different situation such as clustering an entire scene. There may be a larger variability in edges and angles which the methods largely depend on, and therefore may perform better. Exploring this may demonstrate which situations are best for these algorithms to perform. It would be useful to know the limits of data quality for clustering methods like Region Growing and CPC to work.

References

[1] *IntelR©RealSenseTMCross Platform API (librealsense)*. 2017. URL: `https://software.intel.com/sites/products/realsens-e/camera/`.

[2] Yangyan Li et al. "Pointcnn: Convolution on x-transformed points". In: *Advances in Neural Information Processing Systems*. 2018, pp. 820–830.

[3] Charles Ruizhongtai Qi et al. "Pointnet++: Deep hierarchical feature learning on point sets in a metric space". In: *Advances in neural information processing systems*. 2017, pp. 5099–5108.

[4] Andrew Rosenberg and Julia Hirschberg. "V-measure: A conditional entropy-based external cluster evaluation measure". In: *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*. 2007, pp. 410–420.

[5] Markus Schoeler, Jeremie Papon, and Florentin Worgotter. "Constrained planar cuts-object partitioning for point clouds". In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 5207–5215.

[6] Anh-Vu Vo et al. "Octree-based region growing for point cloud segmentation". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 104 (2015), pp. 88–100.