Improving Point Cloud Shape Analysis of a Point Cloud Transformer Using Curve Aggregation

Christian Hiebl
christian.hiebl@tum.de

Atilla Alpay Nalcaci
atilla.nalcaci@tum.de

Fabian Seiler
fabian.seiler@tum.de

Cavit Cakir
cavit.cakir@tum.de

Abstract

Local feature aggregation is an operation that assembles point features of a given key point set, computes the position encodings of the subject point and the neighboring points, and passes the results into relevant transformation and aggregation modules in furtherance of local feature extraction. Even though these operations are feasible for depicting relative local patterns, they are inept with regard to long-range point relations. To that extent, the aggregation strategy introduced by Xiang et al. [7] proposes a new long-range feature aggregation method, namely curve aggregation, for point clouds shape analysis. The initiative of our project [1] is to implement the curve aggregation method upon the Point Cloud Transformer (PCT) of Guo et al. [2], replacing the local neighbor embedding strategy.

1. Introduction

Point clouds are fundamental data structures for the majority of computer vision applications that require sampling and analysis of the 3D environment. Example areas include autonomous driving, LiDAR segmentation, augmented reality, and 3D laser scanning. At any rate, effective analysis techniques are highly utilized on account of point clouds having no surface connectivity information. In general, points are in irregular and unordered forms, thus creating design challenges in terms of feature aggregation and grouping schemes.

To that extent, Guo et al. [2] proposes a novel framework, namely *Point Cloud Transformer* (PCT), that performs point feature learning through an attention medium. PCT eliminates the need for defining order for point clouds through the use of inherent order invariance. In terms of point cloud segmentation, PCT follows a prediction model through the concatenation of global features with point-wise features, subsequently predicting part labels for each point.

In the context of point feature aggregation, Xiang et al. [7] propose a novel feature aggregation paradigm, namely *Curve Aggregation*, which generates continuous sequences of point segments in order to improve point cloud geometry learning. They further posit that these continuous descriptors (denoted as *curves*) provide a more adequate scheme for depicting point cloud geometry. Comparisons with traditional local and non-local methods depict highly enhanced part segmentation and normal estimation scores.

To this end, our project aims to improve the point cloud segmentation method used in PCT by modifying and replacing the curve aggregation approach proposed in CurveNet with the purpose of enhancing object part segmentation accuracy and precision. While extensive experimentation shows that PCT achieves state-of-the-art performance pertaining to classification, part segmentation, and normal estimation, transition, and implementation of curve aggregation method to the PCT exhibited enhanced performance with respect to part segmentation task. The paper is organized as follows. Section 2 presents the related work in tandem with our project. Section 3 summarizes the transition and implementation of the curve aggregation method to PCT. Section 4.1 examines the data sets that are used. In Section 4.2, we provide the baseline and enhanced results with further evaluations. Finally, Section 5 concludes the paper.

2. Related Work

2.1. PCT: Point Cloud Transformer

Point Cloud Transformer [2] is a framework that employs point cloud learning in reference to *Transformer* by Vaswani et al. [5], a dominant framework in the area of natural language processing which has been applied to numerous image vision tasks [1]. In summary, Transformer comprise of a decoder-encoder structure that contains three

[1] https://github.com/AtillaA/PCT-CIC

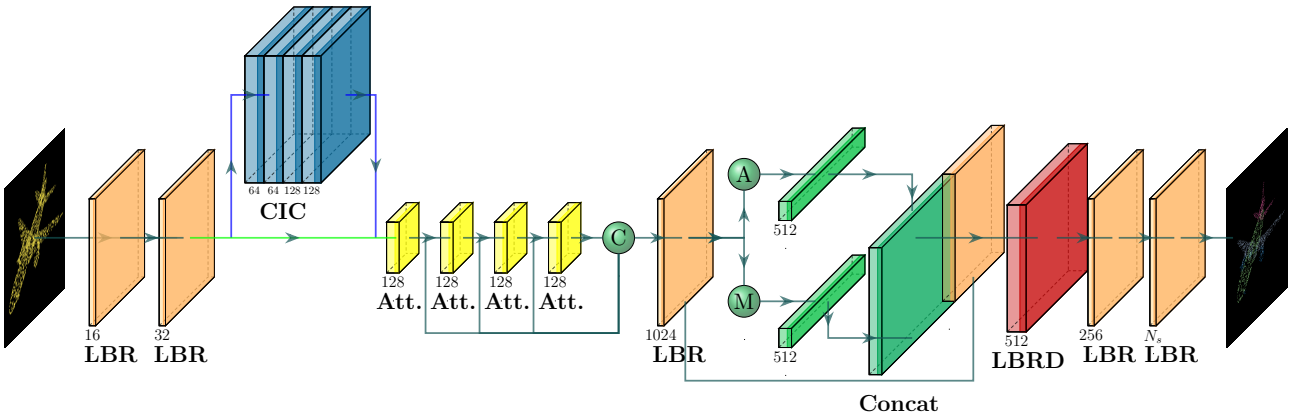


Figure 1. PCT architecture in conjunction with 4 CIC layers shown. Model layers are abbreviated in the following way: LBRD - Linear, BatchNorm, ReLU, Dropout; LBR - Linear, BatchNorm, ReLU; Att. - Attention; C - Concatenation of the Attention layers; M - Max Pooling; A - Average Pooling. For the regular PCT structure the blue arrowed path to the CIC layers is not followed; instead, the green arrow directly to the attention layers is chosen.

KNN	Instance IoU
5	85.14
10	85.04
20	**85.33**
40 (*default*)	84.93
80	84.99

Table 1. PCT [2] with various k-nearest neighbors evaluating with instance average IoU after 100 epochs.

modules for input embedding, order encoding, and self-attention which being the core component anent generating attention features. All operations of respective modules are parallelizable and order-independent, essentially qualifying the architecture as a replacement to a convolution operation in a convolutional neural network with increased versatility.

PCT aims to conduct feature learning by the use of attention mechanism via invalidating the prerequisite of defining order for point cloud data. This is achieved using the inherent order invariance feature of Tranformer to employ point cloud learning in preference to Transformer's original implementation of NLP tasks. In the context of point cloud segmentation, point features are concatenated with global part features to yield the corresponding part label predictions per point.

2.2. CurveNet: Learning Curves for Point Clouds Shape Analysis

CurveNet [7] aims to improve point cloud geometry learning through generating continuous sequences of point segments. These continuous descriptors are denoted as curves. Essentially, curves describe preset paths for an undirected graph where point clouds are regarded as undirected graphs with discrete points serving as nodes and neighbor point connections serving as edges.

In summary, operation of curve aggregation is encapsulated in a Curve Intervention Convolution (CIC) block, consisting of farthest point sampling algorithm, curve grouping and curve aggregation operations, respectively. At the outset, relative point differences are projected to a higher dimension through local point-feature aggregation layer, which are then fed into the farthest point sampling algorithm to sample from the rest points that is the farthest from the set of sampled points. Subsequently, curves are first grouped using a predefined learnable π function that finds the optimal walk on the graph to determine the next state of the respective curve, and then aggregated by scaling inter- and intra-channel feature varieties of points to all point features.

2.3. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation

As one of the pioneer direct approaches, PointNet/PointNet++ [3, 4] by Qi et al. utilizes shared MLPs to learn pointwise features of input point clouds. In particular, the architecture performs feature learning on point clouds by use of multi-layer perceptrons (MLPs), maxpooling operators and rigid transformations in order to ensure invariance under permutations and rotation for a given orientation. Our project does not employ technical specifics with regard to this architecture, but most of the baseline model experiments and comparisons are carried out in contrast to this architecture, on account of being one of the avant-garde network in the context of part segmentation and semantic parsing.

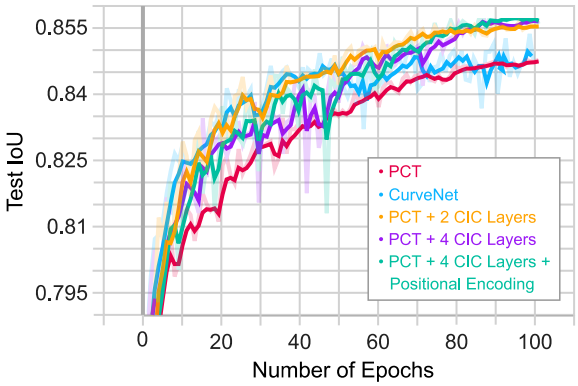


Figure 2. Plot of the average test IoU of various model configurations over 100 epochs. Transparent lines represent the respective unsmoothed graph of a model.

3. Method

Neighbor embedding strategy of PCT improves upon point embedding through assisting the attention module by considering points with semantic information between local groups instead of individual points. While this approach is effective in terms of gathering local geometric information for point cloud learning, long-range feature aggregation is essential for extracting underlying patterns implied by the point cloud shapes.

On the other hand, Curve Aggregation method performs long-range feature aggregation through utilizing continuous sequences of point segments. This strategy enriches point geometries with similar sampling behavior in terms of geometric information by using more relative encoding rules rather than element-wise difference, and provided to be the core finding of this architecture.

Initiative of our project is to implement the Curve Aggregation method of CurveNet upon PCT, replacing the local neighbor embedding strategy. Local feature aggregation methods are feasible for depicting relative local patterns, but they are inept with regard to long-range point relations. Generation of continuous sequences of point segments posit enhancements in terms of point cloud geometry learning and depicting the geometry of point cloud objects, compared to existing local and non-local operators.

The implementation of the proposed architecture is displayed in Figure 1. After processing the point cloud into an input embedding using two subsequent Linear-BatchNorm-ReLU (LBR) layers, CIC blocks are introduced. Depending on the variation, two or fours layers of these blocks are included. Afterwards, the embedding which now contains not just local information but also global information from the curve grouping and aggregation is passed to the attention layers. In here each point learns to attend to relevant other points in four subsequent attention layers. After conducting both max pooling and average pooling to further aggregate global information, it is concatenated with the embedding from the attention layers to obtain a feature rich representation. Further processing is done in the Linear-BatchNorm-ReLU-Dropout (LBRD) and LBR layers until each point is classified as one of the N_s part classes.

Methods	Instance IoU
PCT [2]	84.9
CurveNet [7]	85.4
PCT + 2 CIC + positional embedding	85.5
PCT + 4 CIC	**85.7**
PCT + 4 CIC + positional embedding	85.6

Table 2. Comparing vanilla versions of the architecture in comparison to including CIC layers and also the coordinate information.

4. Experiments

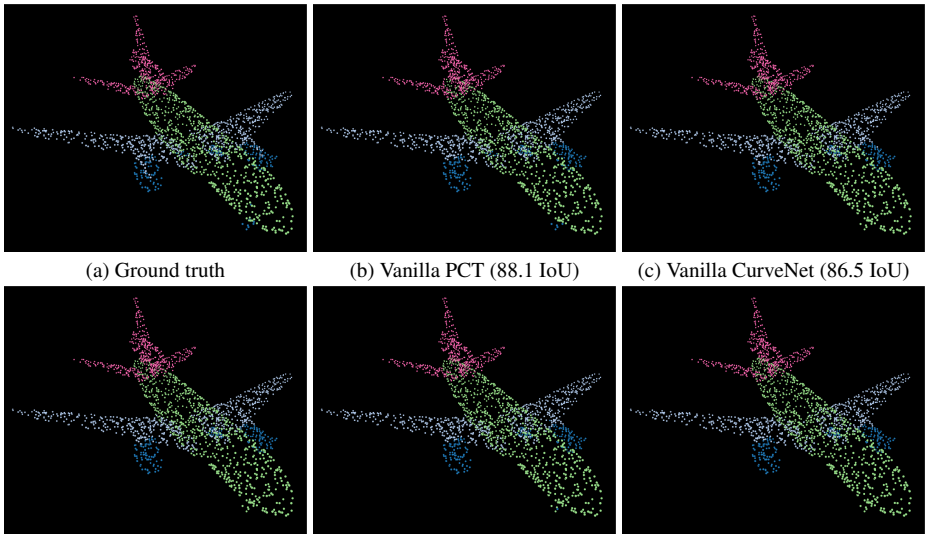
4.1. Benchmarks

In order to evaluate our proposed method, we used the ShapeNet [6] Parts dataset for the task of part segmentation. Both PCT and CurveNet used this dataset as the basis for their model evaluation, enabling us to compare our results with theirs. The dataset contains 16,880 meshes categorized into 16 object classes (i.e. plane, car, chair) with 2-5 parts per object and a total of 50 part classes. As is common in other works, the class label is passed to the model during training and inference to facilitate the network to learn class-specific parts. In the inference setting, the class used for part segmentation can be predicted from the classification head.

4.2. Analysis

4.2.1 Baseline Results

In order to compare our proposed architecture, baseline experiments of PCT and CurveNet were conducted. After inspection of the PCT paper and its corresponding code base, ablation studies were done that consist of varying values of k-nearest neighbors algorithm, since this would influence the amount of local information each point will infuse. Table 1 depicts the respective results of the architecture. Default k value in the code base was given as 40, as also shown in the table. Using values different than the default, slightly higher results were achieved, the highest with k value of 20 neighbors where instance intersection over union metric yielding 85.33. Authors of the PCT paper do not mention a reasoning behind their number count selection, and no supplementary material with ablations was provided.



(a) Ground truth (b) Vanilla PCT (88.1 IoU) (c) Vanilla CurveNet (86.5 IoU)

(d) PCT + 2 CIC Layers + Positional Encoding (87.7 IoU) (e) PCT + 4 CIC Layers (88.6 IoU) (f) PCT + 4 CIC Layers + Positional Encoding (89.6 IoU)

Figure 3. Point cloud segmentations of different model architectures on ShapeNet Parts class airplane - instance 70.

4.2.2 Enhanced Results

As done for the verification experiments, the other experiments were trained on a single RTX 3090 GPU with 24GB VRAM (ML3D cluster). Figure 2 delineates the results for the original PCT in red, CurveNet in blue, and our implementations of PCT including the curve intervention convolution layers, and Table 2 outlines the best results by using the CIC layers ingrained into the PCT architecture, correspondingly.

Unsurprisingly, CurveNet performs higher test results than PCT overall. While the vanilla implementation of CurveNet achieved an instance intersection over union of 85.4 and PCT an instance intersection over union of 84.9, it can be observed that the hybrid PCT implementation that is including curve intervention convolution layers perform marginally better than CurveNet, more particularly achieving intersection over union metric of 85,7, compared to 85,4 that is achieved by vanilla CurveNet implementation in the verifications. This PCT-CIC hybrid thus effectively employs the curve grouping and aggregation methods from CurveNet together with the attention layers of PCT architecture.

Even though the PCT-CIC hybrid architectures achieved the highest results, it must be noted that the vanilla implementation of CurveNet in the original paper is reported to achieve an instance intersection over union of 86.8, and the authors of the PCT reported to have achieved an instance intersection over union value of 86.6.

5. Conclusion

In conclusion, original PCT framework with the implementation of varying number of curve intervention convolution layers yields marginally higher performance. In particular, achieving 85.7 intersection over union metric compared to 85.4 of CurveNet architecture, demonstrating the effective use of curve grouping and aggregation methods of CurveNet upon PCT. The resultant hybrid PCT-CIC implementation thus verified to be an efficient candidate in the context of object part segmentation task.

Additionally, ablation studies that are carried out with regard to vanilla PCT has shown divergent results. By assigning nearest neighbor values from 5 to 80 with twofold multiplication in between, test metrics concerning intersection over union achieved the highest result of 85.33 with 20 nearest neighbor value, using anterior k-nearest neighbors method implemented in PCT. Although default nearest neighbors value of 40 yielding 84.93 intersection over union metric, authors of the PCT do not mention the rationale behind their number count selection and no supplementary material with ablations was provided. As a future work, the project can be tested with even more varying neighbor values under increased epoch size in order to unveil distinct performance evaluations of hybrid PCT-CIC under the influence of disparate hyperparameters. Additionally it would be interesting to see if including attention layers after the max and average pooling has an improved effect since the points would contain even more global context information.

References

[1] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. 1

[2] Meng-Hao Guo, Jun-Xiong Cai, Zheng-Ning Liu, Tai-Jiang Mu, Ralph R Martin, and Shi-Min Hu. Pct: Point cloud transformer. *Computational Visual Media*, 7(2):187–199, 2021. 1, 2, 3

[3] Charles R. Qi, Hao Su, Kaichun Mo, and Leonidas J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation, 2016. 2

[4] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017. 2

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 1

[6] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. pages 1912–1920, 06 2015. 3

[7] Tiange Xiang, Chaoyi Zhang, Yang Song, Jianhui Yu, and Weidong Cai. Walk in the cloud: Learning curves for point clouds shape analysis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 915–924, 2021. 1, 2, 3