Point Cloud Oversegmentation with Graph-Structured Deep Metric Learning

Loic Landrieu[1], Mohamed Boussaha[2]
Univ. Paris-Est, IGN-ENSG, LaSTIG, [1]STRUDEL, [2]ACTE, Saint-Mandé, France
loic.landrieu@ign.fr,mohamed.boussaha@ign.fr

Abstract

In [15], we propose the first supervized learning framework for oversegmenting 3D point clouds into superpoints. We cast this problem as learning deep embeddings of the local geometry and radiometry of 3D points, such that the border of objects presents high contrasts. The embeddings are computed using a lightweight neural network operating on the points' local neighborhood. Finally, we formulate point cloud oversegmentation as a graph partition problem with respect to the learned embeddings.

This new approach allows us to set a new state-of-the-art in point cloud oversegmentation by a significant margin, on a dense indoor dataset (S3DIS) and a sparse outdoor one (vKITTI3D). Our best solution requires over five times fewer superpoints to reach similar performance than previously published methods on S3DIS. Furthermore, we show that our framework can be used to improve superpoint-based semantic segmentation algorithms, setting a new state-of-the-art for this task as well.

1. Introduction

The interest of segmenting point clouds into sets of points known as superpoints—the 3D equivalent of superpixels— as a preprocessing step to their analysis has been extensively demonstrated [19, 3]. However, these unsupervized methods rely on the assumption that segments which are geometrically and/or radiometrically homogeneous are also semantically homogeneous. This assertion should be challenged, especially since the quality of any further analysis is limited by the quality of the initial oversegmentation. Our objective in this paper is to formulate a supervized framework for oversegmentating 3D point clouds into semantically pure superpoints in order to facilitate their semantic segmentation.

Although superpixel-based methods and deep learning have both been around for a long time in computer vision, convolutional neural networks have only recently been used for superpixel oversegmentation. Notably, [14] introduced a loss function emulating oversegmentation metrics, and which is compatible with graph-based clustering methods. [9] propose a fully differentiable spatial clustering methods. Both approaches have shown promising results, displaying significant improvement upon methods relying on handcrafted descriptors. In this paper, we build upon these ideas, albeit in the 3D setting.

We propose formulating point cloud oversegmentation as a deep metric learning problem structured by an adjacency graph defined on an input 3D point cloud. We introduce the *graph-structured contrastive loss*, a loss function which learns to embed 3D points homogeneously within objects and with high contrast at their interface. The resulting segmentation is defined as the piecewise-constant approximation of such learned embedding in the adjacency graph.

We show that our approach can be integrated with the superpoint graph approach of [11] to significantly improve the resulting semantic segmentation.

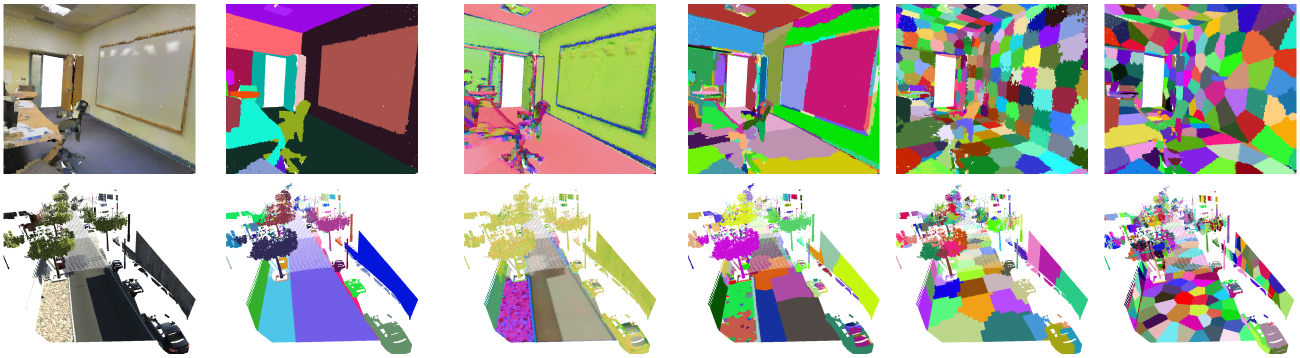
2. Method

Our objective is to associate to each point i of a 3D point cloud C a compact m-dimensional embedding e_i with high contrast along objects borders. We constrain such embeddings to be within the m-unit sphere $\mathbb{S}_m$ to prevent collapse during the training phase. To this end, we introduce the Local Point Embedder (LPE), a lightweight network inspired by PointNet [17]. However, unlike PointNet, LPE does not try to extract information from the whole input point cloud, but rather encodes each point based on purely local information.

2.1. The Generalized Minimal Partition Problem

Once the embeddings are computed, we define the superpoints with respect to an adjacency graph $G = (C, E)$ derived from the point cloud C. As proposed by [7], we define the superpoints as the constant connected components in G of a piecewise-constant approximation of the embeddings $e \in \mathbb{S}_m^C$. This approximation is the solution $f^\star$ of the following optimization problem:

$$f^\star = \underset{f \in \mathbb{R}^{C \times m}}{\arg\min} \sum_{i \in C} \|f_i - e_i\|^2 + \sum_{(i,j) \in E} w_{i,j} \left[f_i \neq f_j\right], \quad (1)$$



Input cloud Ground truth objects Learned embeddings SSP (ours) VCCS [16] Lin in [13]

Figure 1: Illustration of the oversegmentations produced by our framework (SSP), and by competing algorithms with an equivalent superpoint count (around 400). In the first row, we show a scene from the S3DIS dataset with 58 objects. In the second row, we show a scene from the vKITTI3D dataset with 233 objects. The embeddings in third column are projected into a 3-dimensional space to allow color visualization.

with $w \in \mathbb{R}_+^E$ the edges' weight and $[x \neq y]$ equal to 0 if $x = y$ and 1 otherwise. To encourage splitting along high contrast areas, we define the edge weight as $w_{i,j} = \lambda \exp\left(\frac{-1}{\sigma}\|e_i - e_j\|^2\right)$, with parameters $\lambda, \sigma \in \mathbb{R}^+$.

Problem (1), known as the *generalized minimal partition* (GMP) and introduced by [10], is neither continuous, differentiable, nor convex, and therefore the global minimum cannot be realistically retrieved. However, the ℓ_0-cut pursuit algorithm [10] allows for fast approximate solutions. However, the noncontinuity prevents us from backpropagating directly from the approximated partition.

2.2. Graph-Structured Contrastive Loss

We introduce a surrogate loss called *graph-structured contrastive loss* focusing on correctly detecting the borders between objects. To this end, we define E_{intra} (resp. E_{inter}) the set of *intra-edges* (resp. *inter-edges*) as the set of edges of G between points within the same object (resp. point from different adjacent objects).

In the spirit of the original contrastive loss [4], our loss encourages embeddings of vertices linked by an intra-edge to be similar, while rewarding different embeddings when linked by an inter-edge:

$$\ell(e)=\frac{1}{|E|}\left(\sum_{(i,j)\in E_{\text{intra}}}\phi\left(e_i-e_j\right)+\sum_{(i,j)\in E_{\text{inter}}}\mu_{i,j}\psi\left(e_i-e_j\right)\right),$$

with ϕ (resp. ψ) a function minimal (resp. maximal) at 0, and $\mu_{i,j} \in \mathbb{R}^{E_{\text{inter}}}$ a weight on inter-edges. A point embedding function minimizing this loss will be uniform within objects and have stark contrasts at their interface. Consequently, the components of the piece-wise constant approximation of (1) should follow the objects' borders. This loss differs from the triplet loss [8, 20], as it involves all vertices within a graph (or a sub-graph) at once, and not just an anchor and related positive/negative examples. In this way, it bypasses the problem of example picking altogether. Indeed, the positive and negative examples are directly given by the adjacency structure set by E_{intra} and E_{inter}.

We chose ϕ, the function promoting intra-object homogeneity as $\phi(x) = \delta(\sqrt{\|x\|^2/\delta^2+1}-1)$ with $\delta = 0.3$. This means that the first term of ℓ is the (pseudo)-Huber graph-total variation on the E_{intra} edges [2].

With $\psi(x) = \max(1 - \|x\|, 0)$, the second part of ℓ is the opposite of the truncated graph-total variation [23] on the inter-edges. It penalizes similar embeddings at the border between objects. Conscious that our embeddings are restricted to the unit sphere, we threshold this function for differences larger than 1 (corresponding to a 60 degree angle). In other words, ψ encourages vertices linked by an inter-edge to take embeddings with an euclidean distance of 1, but does not push for a larger difference.

The choice of $\mu_{i,j}$ plays a crucial role in the efficiency of the graph-structured contrastive loss. Indeed, small errors in the former can have drastic consequences in the latter. Indeed, a single missed edge can erroneously fuse two large superpoints covering different objects. Therefore, we need to incorporate the induced partition's purity into the loss.

Inspired by [14], we introduce the cross-partition weighting strategy to chose the weight $\mu_{i,j}$. This strategy allows us to directly take into account the error induced by a given embedding in its induced partition compared to a ground-truth segmentation.

3. Numerical Experiments

3.1. Point Cloud Oversegmentation

We evaluate our approach on two datasets of different natures, represented in Figure 1. The first one is S3DIS [1], composed of dense indoor scans of rooms in an office setting. The second one is vKITTI3D [5], an outdoor dataset of urban scenes that mimics sparse LiDAR acquisitions.

In Figure 2, we report the performance of our algorithm according to three segmentation metrics: OOA, BR, and BP. OOA denotes the Oracle Overall Accuracy, *i.e.* the OA of the oracle classification algorithm associating the majority label to each segment of the proposed partition. BP (resp. BR) denotes the precision (resp. recall) of the predicted transition edges compared to the true transition edges between objects, with a tolerance of one edge.

In all our experiments, we set m the dimension of our embeddings to 4. We choose a light architecture for the LPE, with less than $15,000$ parameters.

We denote our method by **SSP** for *Supervized Super-Points*. We also implemented **SSP-SEAL**, a version in which the cross-partition strategy is replaced by the SEAL, and **SSP-cluster** our implementation of the soft partition approach of [9] to the 3D setting. We also compare our method to the graph-based method introduced by [7] solving (1) on handcrafted features instead of learned ones. Finally, we used available implementation of two state-of-the-art point cloud segmentation methods: the octree-structured cluster-based method **VCCS**[16], and **Lin *et al.*** , the adaptive resolution graph-based method introduced by [13].

We observe that our approach significantly outperforms the other approaches on all metrics. In particular, we remark that **SSP** only requires under 350 superpoints to reach a performance comparable with **VCCS** with over $1,800$ superpoints on S3DIS. Furthermore, the quality of the border is unmatched in our range of superpoints. The improvement is less significant on vKITTI3D, which could be due to the difficulty of constructing an adjacency graph on such a sparse acquisition. The performance is degraded further without color information, as some transition are not predictable with purely from the geometry. **Geom-Graph** performs well on the accuracy, but not on the boundary. This is expected as the handcrafted geometric features cannot detect some borders, such as adjacent walls. **SSP-Cluster** performs better than the unsupervized cluster-based method of Lin *et al.* , but still suffer from the typical limitations of clustering methods, such as sensitivity to initialization.

In terms of computational speed, the embeddings can be computed very efficiently in parallel on a GPU with over 3 million embeddings per second on a 1080Ti GPU. The bottleneck remains solving the graph partition problem in (1), which can process around $100,000$ points per second.

3.2. Semantic Segmentation

In Table 1 and Table 2, we show how our point cloud oversegmentation framework can be successfully used by the superpoint-based semantic segmentation technique of [11][1] (**SPG**). We replace the unsupervized superpoint computation with our best-performing approach, **SSP**. We evaluate the resulting semantic segmentation using standard classification metrics: overall accuracy (OA), mean per-class accuracy (mAcc) and mean per-class intersection-over-union (mIOU). We observe a significant increase in the performance of **SPG**, beating concurrent methods on both datasets. In particular, we observe that our method allows for better retrieval of small objects (see detailed IoU in the appendix), which translates into much better per-class metrics, although the overall accuracy is not necessarily better than the latest state-of-the-art algorithms.

Method	OA	mAcc	mIoU
6-fold cross validation			
PointNet [17] in [5]	78.5	66.2	47.6
Engelmann *et al.* in [5]	81.1	66.4	49.7
PointNet++ [18]	81.0	67.1	54.5
Engelmann *et al.* in [6]	84.0	67.8	58.3
SPG [11]	85.5	73.0	62.1
PointCNN [12]	**88.1**	75.6	65.4
SSP + SPG (ours)	87.9	**78.3**	**68.4**
Fold 5			
PointNet [17] in [6]	-	49.0	41.1
Engelmann *et al.* in [6]	84.2	61.8	52.2
PointCNN [12]	85.9	63.9	57.3
SPG [11]	86.4	66.5	58.0
PCCN [21]	-	67.0	58.3
SSP + SPG (ours)	**87.9**	**68.2**	**61.7**

Table 1: Performance for the semantic segmentation task on the S3DIS dataset. The top table is for the 6-fold cross validation, the bottom table on the fifth fold.

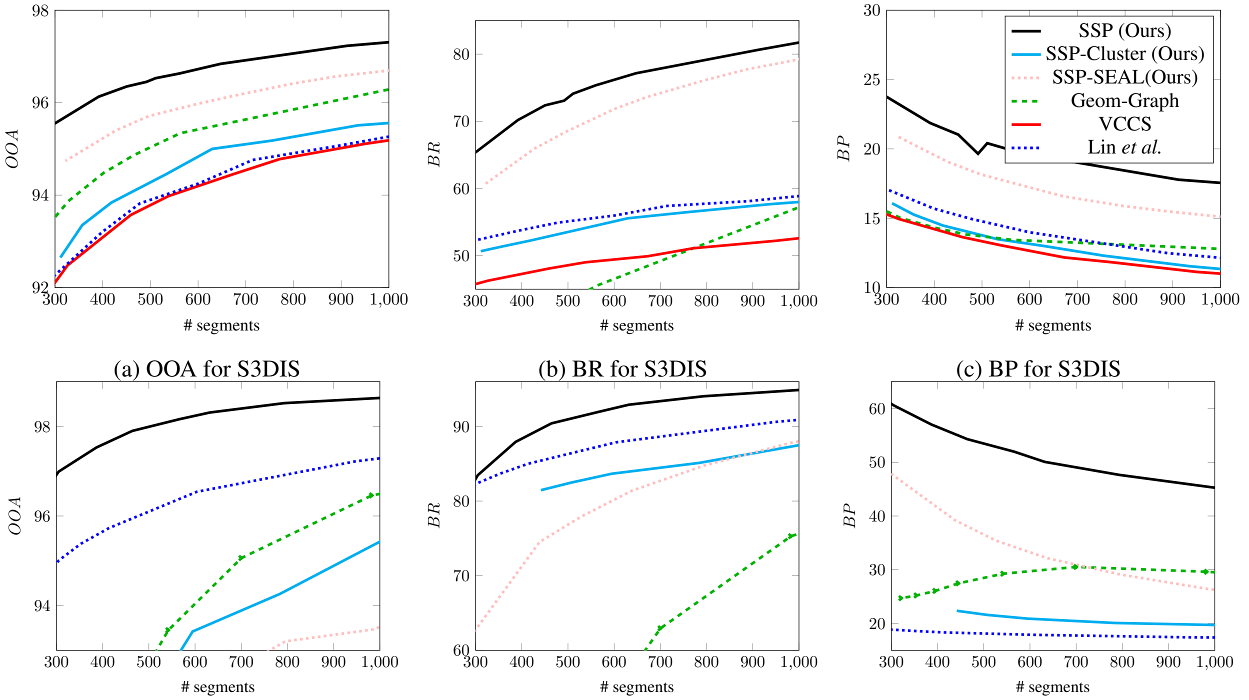
Method	OA	mAcc	mIoU
PointNet [17]	79.7	47.0	34.4
Engelmann *et al.* in [6]	79.7	57.6	35.6
Engelmann *et al.* in [5]	80.6	49.7	36.2
3P-RNN [22]	**87.8**	54.1	41.6
SSP + SPG (ours)	84.3	**67.3**	**52.0**

Table 2: Performance for the semantic segmentation task on the vKITTI3D dataset with 6-fold cross validation.

4. Conclusion

In this paper, we presented the first supervized 3D point cloud oversegmentation framework. Using a simple point embedding network and a new graph-structured loss function, we were achieved significant improvements compared to the state-of-the-art of point cloud oversegmentation. When combined with a superpoint-based semantic segmentation method, our method sets a new state-of-the-art of semantic segmentation as well. A video illustration is accessible at `https://youtu.be/bKxU03tjLJ4`. The source code will be made available to the community as well as trained networks in an update to the superpoint-graph repository[1]. Future work will focus on adapting this work to other data structure.

[1] `https://github.com/loicland/superpoint-graph`



(a) OOA for S3DIS

(b) BR for S3DIS

(c) BP for S3DIS

(d) OOA for vKITTI

(e) BR for vKITTI

(f) BP for vKITTI

Figure 2: Performance of the different algorithms on the 6-fold S3DIS dataset (a, b, c), and the 6-fold vKITTI3D (d, e, f).

References

[1] I. Armeni, O. Sener, A. R. Zamir, H. Jiang, I. K. Brilakis, M. Fischer, and S. Savarese. 3d semantic parsing of large-scale indoor spaces. In *CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*, 2016. 2

[2] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on Image Processing*, 6(2), 1997. 2

[3] J. Chen and B. Chen. Architectural modeling from sparsely scanned range data. *International Journal of Computer Vision*, 78(2-3), 2008. 1

[4] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *CVPR*, volume 1. IEEE, 2005. 2

[5] F. Engelmann, T. Kontogianni, A. Hermans, and B. Leibe. Exploring spatial context for 3d semantic segmentation of point clouds. In *ICCV Workshops*, 2017. 3

[6] F. Engelmann, T. Kontogianni, J. Schult, and B. Leibe. Know what your neighbors do: 3d semantic segmentation of point clouds. In *GMDL Workshop, ECCV*, 2018. 3

[7] S. Guinard and L. Landrieu. Weakly supervised segmentation-aided classification of urban scenes from 3d lidar point clouds. In *ISPRS Workshop*, 2017. 1, 3

[8] E. Hoffer and N. Ailon. Deep metric learning using triplet network. In *International Workshop on Similarity-Based Pattern Recognition*. Springer, 2015. 2

[9] V. Jampani, D. Sun, M. Liu, M. Yang, and J. Kautz. Superpixel sampling networks. In *ECCV*, 2018. 1, 3

[10] L. Landrieu and G. Obozinski. Cut pursuit: Fast algorithms to learn piecewise constant functions on general weighted graphs. *SIAM Journal on Imaging Sciences*, 10(4), 2017. 2

[11] L. Landrieu and M. Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *CVPR*. IEEE, 2018. 1, 3

[12] Y. Li, R. Bu, M. Sun, and B. Chen. PointCNN. *arXiv preprint arXiv:1801.07791*, 2018. 3

[13] Y. Lin, C. Wang, D. Zhai, W. Li, and J. Li. Toward better boundary preserved supervoxel segmentation for 3d point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 143, 2018. 2, 3

[14] W.-C. T. M.-Y. Liu, V. J. D. S. Shao-Yi, C. M.-H. Yang, and J. Kautz. Learning superpixels with segmentation-aware affinity loss. In *CVPR*. IEEE, 2018. 1, 2

[15] M. B. Loic Landrieu. Point cloud oversegmentation with graph-structured deep metric. In *CVPR*. IEEE, 2019. 1

[16] J. Papon, A. Abramov, M. Schoeler, and F. Wörgötter. Voxel cloud connectivity segmentation - supervoxels for point clouds. In *CVPR*, 2013. 2, 3

[17] C. R. Qi, H. Su, K. Mo, and L. J. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *CVPR, IEEE*, 1(2), 2017. 1, 3

[18] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. PointNet++: Deep hierarchical feature learning on point sets in a metric space. In *NIPS*, 2017. 3

[19] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz. Towards 3d point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56(11), 2008. 1

[20] J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, and Y. Wu. Learning fine-grained image similarity with deep ranking. In *CVPR*, 2014. 2

[21] S. Wang, S. Suo, W.-C. M. A. Pokrovsky, and R. Urtasun. Deep parametric continuous convolutional neural networks. In *CVPR*, 2018. 3

[22] X. Ye, J. Li, H. Huang, L. Du, and X. Zhang. 3d recurrent neural networks with context fusion for point cloud semantic segmentation. In *ECCV*, 2018. 3

[23] T. Zhang et al. Some sharp performance bounds for least squares regression with l1 regularization. *The Annals of Statistics*, 37(5A), 2009. 2