Omni-supervised Point Cloud Segmentation via Gradual Receptive Field Component Reasoning

Jingyu Gong[1] Jiachen Xu[1] Xin Tan[1] Haichuan Song[2]
Yanyun Qu[3] Yuan Xie[2*] Lizhuang Ma[1,2*]
[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China
[2]School of Computer Science and Technology, East China Normal University, Shanghai, China
[3]School of Informatics, Xiamen University, Fujian, China
{gongjingyu,xujiachen,tanxin2017}@sjtu.edu.cn hcsong@cs.ecnu.edu.cn
yyqu@xmu.edu.cn yxie@cs.ecnu.edu.cn ma-lz@cs.sjtu.edu.cn

Abstract

Hidden features in neural network usually fail to learn informative representation for 3D segmentation as supervisions are only given on output prediction, while this can be solved by omni-scale supervision on intermediate layers. In this paper, we bring the first omni-scale supervision method to point cloud segmentation via the proposed gradual Receptive Field Component Reasoning (RFCR), where target Receptive Field Component Codes (RFCCs) are designed to record categories within receptive fields for hidden units in the encoder. Then, target RFCCs will supervise the decoder to gradually infer the RFCCs in a coarse-to-fine categories reasoning manner, and finally obtain the semantic labels. Because many hidden features are inactive with tiny magnitude and make minor contributions to RFCC prediction, we propose a Feature Densification with a centrifugal potential to obtain more unambiguous features, and it is in effect equivalent to entropy regularization over features. More active features can further unleash the potential of our omni-supervision method. We embed our method into four prevailing backbones and test on three challenging benchmarks. Our method can significantly improve the backbones in all three datasets. Specifically, our method brings new state-of-the-art performances for S3DIS as well as Semantic3D and ranks the 1st in the ScanNet benchmark among all the point-based methods. Code is publicly available at https://github.com/azuki-miho/RFCR.

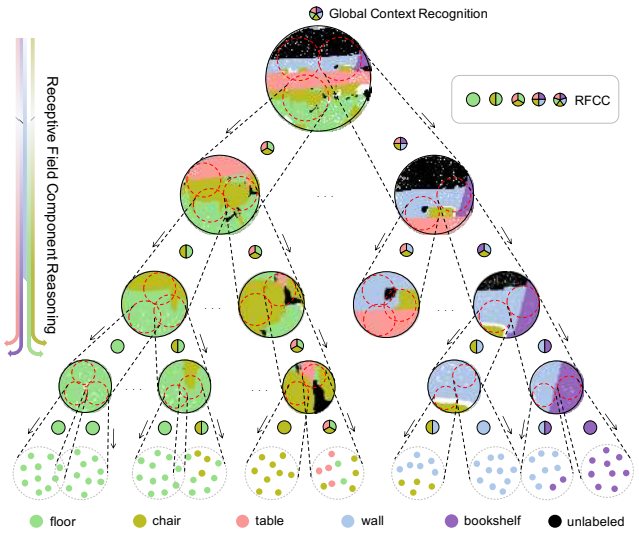

Figure 1: Illustration of Receptive Field Component Reasoning for a point cloud in ScanNet v2 from top to bottom. The Receptive Field Component Code (RFCC) indicates the category components in the receptive field. In the decoding stage, the segmentation problem is decomposed into a much easier global context recognition problem (predicting the global RFCCs, see the top of figure) and a series of receptive field component reasoning problems. During reasoning, the target RFCCs generated in the encoder are used as the groundtruth in the decoder to guide the network to gradually reason the RFCCs in a coarse-to-fine manner, and finally obtain the semantic labels.

1. Introduction

Semantic segmentation of point cloud in which we need to infer the point-level labels is a typical but still challenging task in 3D vision. Meanwhile, this technique can be widely used in many applications like robotics, autonomous driving, and virtual/augmented reality.

To handle point cloud segmentation, previous works usu-

*Corresponding Author

ally introduced well-designed encoder-decoder architecture to hierarchically extract global context features in the encoding stage, and distribute contextual features to points in the decoding stage to achieve point-wise labeling [5, 31, 38]. However, in the typical encoder-decoder framework, network is merely supervised by labels of points in the final layer [36, 31, 9], while ignoring a critical fact that, hidden units in other layers lack direct supervision to extract features with informative representation. In other words, multi-scale/omni-scale supervision is indeed necessary.

In 2D vision, CVAE [28] attempted to give a multi-scale prediction and supervision to extract useful features in segmentation task. CPM [35] and MSS-net [14] tried to add intermediate supervision periodically and layer-wise loss, respectively. PointRend [16] proposed to segment image in low-resolution, and iteratively up-sample the coarse prediction and fine-tune it to obtain final result, thus prediction at different scales can be supervised together.

However, so far, no one succeed in applying multi-scale, let alone omni-scale supervision to 3D semantic segmentation, due to the irregularity of point cloud. Unlike in image domain, it is hard to up-sample the hidden features to the original resolution through simple tiling or interpolation, because there is no fixed mapping relationship between sampled point cloud and original point cloud especially when the sampling is random [36, 9]. Additionally, the common up-sampling methods using nearest neighbors cannot trace the encoding relationship, thus introducing improper supervisions to the intermediate features (referring Sec 4.4 for discussion). More recently, SceneEncoder [37] provided a method to supervise the center-most layer to extract meaningful global features, but lots of other layers remain unhandled.

To solve this problem, we propose an omni-scale supervision method via gradual Receptive Field Component Reasoning. Instead of up-sampling the hidden features to the original resolution, we design a Receptive Field Component Code (RFCC) to effectively trace the encoding relationship and represent the categories within receptive field for each hidden unit. Based upon this, we generate the target RFCCs at different layers from semantic labels in the encoding stage to supervise the network at all scales. Specifically, in the decoding stage, the target RFCCs will supervise the network to predict the RFCCs at different scales, and the features (hints) from skip link can help further deduce RFCCs within more local and specific receptive fields. In this way, the decoding stage is transferred into a gradual reasoning procedure, as shown in Figure 1.

Inspired by SceneEncoder [37], for each sampled point in any layer of encoder, according to the existence of categories in its receptive field, a multi-hot binary code can be built, designated as target Receptive Field Component Code (RFCC). The target RFCCs at different layers are generated alongside the convolution and down-sampling, thus they can precisely record the existing categories in corresponding receptive fields without any extra annotations. In Figure 1, we show the target RFCCs at various layers for a point cloud in the decoding stage, where the network will first recognize the global context (inferring the categories of objects existing in the whole point cloud). Then, contextual features will be up-sampled iteratively to gradually reason the RFCCs in a coarse-to-fine manner. By comparing the target RFCCs and the predicted RFCCs, the omni-scale supervision can be realized. It is noteworthy that even the network reasons the RFCCs gradually, the training and inference of network is implemented in a end-to-end manner.

Additionally, to further unleash the potential of omni-scale supervision, more active features (features with large magnitude) are required to make unambiguous contribution to the RFCC prediction. Contrarily, in traditional networks [36, 31, 37], lots of units are inactive with tiny magnitude, such that having minor contribution to the final prediction. The principle underlying the above observations comes from entropy regularization [6, 18] over features, where greater number of active dimensionalities would bring low-density separation between positive features and negative features, generating more unambiguous features with certain signals. Consequently, in point cloud scenario, more certainty in features can help the training of the network to better reason the RFCCs at various scales and finally predict the semantic labels. Motivated by this, we proposed a Feature Densification method with a well-deigned potential function to push hidden features away from 0. Moreover, this potential is in effect equivalent to a entropy loss over features (detailed deduction is shown in Sec 3.4), leading to a simple but highly effective regularization for intermediate features.

To evaluate the performance and versatility of our method in point cloud semantic segmentation task, we embed our method into four prevailing backbones (deformable KPConv, rigid KPConv [31], RandLA [9], and SceneEncoder [37]), and test on three challenging point cloud datasets (ScanNet v2 [2] for indoor cluttered rooms, S3DIS [1] for large indoor space, and Semantic3D [7] for large-scale outdoor space). In all the three datasets, we outperform the backbone methods and almost all the state-of-the-art point-based competitors. What's more, we also push the state-of-the-art of S3DIS [1] and Semantic3D [7] ahead.

2. Related Work

Point Cloud Semantic Segmentation. PointNet [25] proposed to directly concatenate global features to point-wise features before several Multi-Layer Perceptrons (MLPs) to finish the semantic segmentation. Later, PointNet++ [26], SubSparseConv [5] and KPConv [31] utilized an encoder-decoder architecture with skip links for better

fusion of local and global information. Joint tasks like instance segmentation and edge detection are also introduced to enhance the performance of semantic segmentation through additional supervision [24, 43, 10]. SceneEncoder [37] designed a meaningful global scene descriptor to guide the global feature extraction. These methods directly utilized semantic labels to supervise the output features or features in the center-most layer.

Compared with previous works, we propose an omni-scale supervision method for point cloud semantic segmentation via a gradual Receptive Field Component Reasoning.

Multi-scale Supervision. In 2D Vision, CVAE [28] proposed to give multi-scale prediction in the segmentation task. RMI [44] proposed to predict and supervise the neighborhood of each pixel rather than the pixel itself. PointRend [16] segmented the images in a coarse-to-fine fashion, *i.e*. give low-resolution prediction, and iteratively up-sample and fine-tune it to obtain the original-resolution prediction. CPM [35] and MSS-net [14] added intermediate supervision periodically and layer-wise loss, respectively.

Compared with these methods, we design a Receptive Field Component Code (RFCC) to represent receptive field component and dynamically generate target RFCCs to give omni-scale supervision to the network rather than simply up-sample the features to the original resolution or down-sample the ground truth. Thanks to the omni-scale supervision, the network can infer the RFCCs gradually and finally obtain RFCCs in the original resolution which is also the semantic labels.

Entropy Regularization. Entropy Regularization [6] minimized the prediction entropy in semi-supervised classification task to obtain unambiguous final features. This idea is introduced into the deep neural network for self-training by [18], and the final features with tiny magnitude will be pushed away from 0 to make deterministic contribution to the final prediction. In these methods, final features with positive values will be greater and negative features will be smaller due to the entropy loss.

Compared with their methods, our Feature Densification introduce the entropy regularization [6, 18] into the hidden features rather than just the final features to obtain more active hidden features which can directly contribute to the RFCC prediction.

3. Methods

In the following parts, we will first give an overview of our method in Sec 3.1. Then, we will introduce the Receptive Field Component Codes (RFCCs) and the target RFCCs that we generate at various layers in Sec 3.2. In Sec 3.3, how to use these target RFCCs to supervise the network, and make the gradual Receptive Field Component Reasoning, would be explained. At last, we will show the strategy of Feature Densification for more active features in Sec 3.4.

3.1. Overview

The framework of our gradual Receptive Field Component Reasoning (RFCR) is shown in Figure 2. In our method, we generate target Receptive Field Component Codes (RFCCs) at different layers alongside the convolution and sampling of features (Figure 2 (a)) in the encoding stage. In the decoding stage, the network will reason the RFCCs at different layers, and the corresponding target RFCCs will give omni-scale supervision on the predicted RFCCs (Figure 2 (b)). Consequently, the semantic segmentation task can be treated as a coarse-to-fine receptive field component reasoning procedure after recognizing the global context (predicting categories of objects existing in the point cloud). Additionally, we introduce Feature Densification through a centrifugal potential to obtain more active features for omni-scale RFCC prediction (Figure 2 (c)).

3.2. Receptive Field Component Code

For a point cloud, it is easy to define the label of a point in the original point cloud. Nevertheless, it is non-trivial to give a label to a point in any down-sampled point cloud which receives information from points inside its receptive field. In our method, we design a Receptive Field Component Code (RFCC) to represent all categories within the receptive field of sampled points in the encoder. The target RFCCs are generated alongside the convolution and sampling of features in the encoding stage. In other words, sharing sampling is used between the encoding stage (left part of top branch in Figure 2) and RFCC generation (Figure 2 (a)), thus the generated target RFCCs can precisely record the category components in the receptive fields, even though the sampling of point cloud is a random process.

Implementation. Our RFCC is designed to be a multi-hot label for every point in any layer of encoder. Specifically, in the semantic segmentation task where we need to classify each point into C categories, the RFCC will be a $1 \times C$ binary vector. Given the i-th point in the l-th layer of the encoder p_i^l, the target RFCC g_i^l represents the categories of objects existing in the receptive field of p_i^l, and each element $g_i^l[k]$ indicates the existence of category k. Based upon this definition, we can first assign the one-hot label of input point p_i to the RFCC g_i^1 in the input layer, because the receptive field of point p_i only contains p_i itself:

$$g_i^1 = \text{one-hot}(y_i), \tag{1}$$

where y_i is the label of point p_i in the original point cloud. As illustrated in Figure 2 (a), we can obtain g_i^l from the

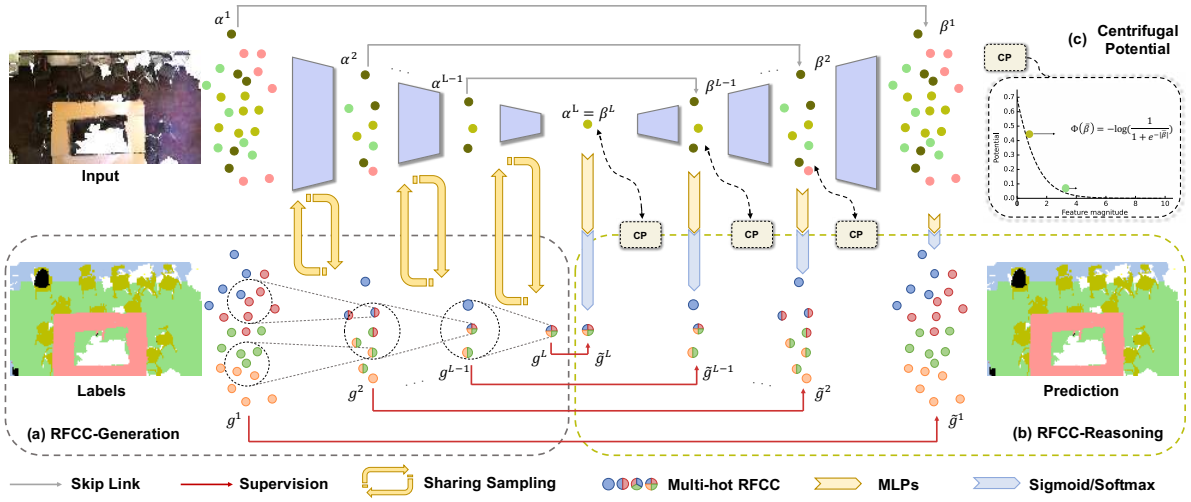

Figure 2: Framework of gradual Receptive Field Component Reasoning. (a) shows the target Receptive Field Component Codes (RFCCs) is generated alongside the common encoding procedure. (b) indicates the network will predict the RFCCs in a coarse-to-fine manner. (c) represents the centrifugal potential which pushes hidden features away from 0. In our network, the target RFCCs will supervise the RFCC predictions, and the learnt feature can reason RFCCs in more local and specific receptive fields as more and more local features (clues) are provided through skip links. The prediction activation function will be Softmax for the final layer and Sigmoid otherwise.

RFCCs in the previous layer g_i^{l-1} alongside the 3D Convs:

$$g_i^l[k] = \bigvee_{j \in \mathcal{N}(i)} \{g_j^{l-1}[k]\} \tag{2}$$

where $k \in [1, C]$ indicates the channel index, and j is the index of point in p_i^l's receptive field at the $(l-1)$-th layer. That is to say, p_i^l receives features from p_j^{l-1} in the 3D Convs thanks to the sharing sampling. $\vee$ represents the logical OR (disjunction) operation. It is noteworthy that the generation of RFCCs only occurs in the encoder, rather than the decoder. The generation of RFCCs is iterated until reaching the center-most layer L. Typically, the scene descriptor is only a naturally deduced global supervisor when the center-most layer contains only one point [37]. Besides, g_i^2 can also be treated as a simplified version of neighborhood multi-dimension distribution in RMI [44], which exploits the semantic relationship among neighboring points.

3.3. RFCC Reasoning

The decoder of network is to infer the category of each input point in the task of semantic segmentation. In our method, as shown in Figure 2 (b), we decompose this complex problem into a much easier global context recognition problem (predicting g_i^L) and a series of gradual receptive field component reasoning problem (reasoning g_i^{l-1} from g_i^l gradually with additional features α_i^l from skip link and finally obtain the semantic labels g_i^1).

As shown in Figure 2, β_i^l is the features of sampled point p_i^l in decoder. For each layer of decoder except the last one, we apply a shared Multi-Layer Perceptron (MLP) $\mathcal{M}^l$ and a sigmoid function σ to β_i^l to predict the RFCCs $\tilde{g}_i^l$:

$$\tilde{g}_i^l = \sigma(\mathcal{M}^l(\beta_i^l)). \tag{3}$$

Then, the target RFCC g_i^l generated in the encoding stage is directly used to guide $\tilde{g}_i^l$ prediction through layer-wise supervision $\mathcal{L}_R^l$:

$$\mathcal{L}_R^l = -\frac{1}{C|P^l|} \sum_{i=1}^{|P^l|} \sum_{k=1}^{C} \mathcal{L}_R^l(i,k), \tag{4}$$

where

$$\mathcal{L}_R^l(i,k) = g_i^l[k] log(\tilde{g}_i^l[k]) + (1 - g_i^l[k]) log(1 - \tilde{g}_i^l[k]), \tag{5}$$

P^l denotes the sampled point cloud in the l-th layer of encoder, and $|P^l|$ corresponds the number of points in P^l.

According to Eq. (3), the center-most features β_i^L which contain global information will learn to recognize the global context, *i.e.*, predict $\tilde{g}_i^L$ with largest receptive field. Meanwhile, g_i^L will be used to regularize this prediction to help β_i^L learn a better representation. Then, for the following layer of decoder, β^L which learns informative representation to predict $\tilde{g}_i^L$ will be up-sampled and concatenated with α_i^{L-1} from the skip link. After that, the concatenated features will be used to extract more distinguishable β_i^{L-1} via 3D Convs, and the extracted features β_i^{L-1} will be used to reason the RFCCs $\tilde{g}_i^{L-1}$ of more local and specific receptive field. This procedure is iterated until $l = 2$. The whole

RFCC reasoning loss can be simply expressed by

$$\mathcal{L}_R = \frac{1}{L-1}\sum_{l=2}^{L}\mathcal{L}_R^l. \tag{6}$$

In the last layer, we can simply utilize the MLPs and softmax to predict the $\tilde{g}_i^1$, and cross entropy loss is used to supervise the output features in the original scale.

3.4. Feature Densification

Due to the large amounts of supervision introduced by the gradual Receptive Field Component Reasoning, more active features with unambiguous signals are required. However, there are many inactive hidden units with tiny magnitude in the traditional network (detailed experiment is shown in Sec 4.4). Therefore, we introduce a centrifugal potential to bring low-density separation between positive features and negative features (*i.e.* push features away from 0) as shown in Figure 2 (c):

$$\Phi(\bar{\beta}) = -\log\frac{1}{1+e^{-|\bar{\beta}|}}, \tag{7}$$

where $\bar{\beta} = a(\beta)$ and a can be an identity function or a simple perceptron. We can see the negative gradient of potential function over feature is:

$$-\frac{\partial\Phi(\bar{\beta})}{\partial\bar{\beta}} = \text{sign}(\bar{\beta})\frac{e^{-|\bar{\beta}|}}{1+e^{-|\bar{\beta}|}} \tag{8}$$

which have the same sign as the feature. This indicates positive features will become greater and negative features will be smaller given this potential. Additionally, features with smaller absolute value will receive larger gradient according to this formula.

Meanwhile, this centrifugal potential can be implemented by a simple entropy loss:

$$\begin{aligned}\mathcal{L}_F^l(i,k) &= \Phi(\bar{\beta}_{i,k}^l)\\ &= -\log\frac{1}{(1+e^{-|\bar{\beta}_{i,k}^l|})}\\ &= \begin{cases}-\log(\sigma(\bar{\beta}_{i,k}^l)) & \bar{\beta}_{i,k}^l \geq 0\\ -\log(1-\sigma(\bar{\beta}_{i,k}^l)) & \bar{\beta}_{i,k}^l < 0\end{cases},\end{aligned} \tag{9}$$

where $\bar{\beta}_{i,k}^l$ is the k-th channel of $\bar{\beta}_i^l$.

If we take the following notation:

$$\begin{aligned}&\tilde{t}_{i,k}^l = \sigma(\bar{\beta}_{i,k}^l)\\ &t_{i,k}^l = 1 \text{ if } \bar{\beta}_{i,k}^l \geq 0,\ 0 \text{ if } \bar{\beta}_{i,k}^l < 0,\end{aligned} \tag{10}$$

we can reformulate Eq. (9) into

$$\mathcal{L}_F^l(i,k) = -[t_{i,k}^l\log(\tilde{t}_{i,k}^l) + (1-t_{i,k}^l)\log(1-\tilde{t}_{i,k}^l)]. \tag{11}$$

So, our centrifugal potential can be treated as entropy regularization [18] over hidden features which can decrease ambiguity of features in the intermediate layers. On the other side, our omni-scale supervision can directly benefit from more active features with certain signal introduced by the Feature Densification. That is because more unambiguous features can participate into the RFCC predictions and help learning better representation of hidden layer, improving the semantic segmentation performance.

The total loss for Feature Densification can be summarized by

$$\mathcal{L}_F = \frac{1}{L-1}\sum_{l=2}^{L}\frac{1}{|P^l|K^l}\sum_{i=1}^{|P^l|}\sum_{k=1}^{K^l}\mathcal{L}_F^l(i,k), \tag{12}$$

and K^l represents the number of features' channel in $\bar{\beta}_i^l$.

In a nutshell, all the supervision can be concluded by

$$\mathcal{L} = \mathcal{L}_S + \lambda_1\mathcal{L}_R + \lambda_2\mathcal{L}_F. \tag{13}$$

where λ_1 and λ_2 are two adjustable hyper-parameters while $\mathcal{L}_S$ represents the common cross entropy loss for semantic segmentation. In our experiment, we simply set λ_1 and λ_2 to 1, and we find it can perform well in most cases.

4. Experiments

To show the effectiveness of our method and prove our claims, we embed our method into four prevailing methods (deformable KPConv, rigid KPConv [31], RandLA [9] and SceneEncoder [37]), and conduct experiments on three popular point cloud segmentation datasets (ScanNet v2 [2] for cluttered indoor scenes, S3DIS [1] for large-scale indoor rooms and Semantic3D [7] for large outdoor spaces). First, we introduce these three datasets in Sec 4.1. Next, implementation details and hyper-parameters used in our experiments are described in Sec 4.2. Then, we give the metric used to evaluate the performance as well as the quantitative and qualitative results in Sec 4.3. Finally, we conduct more ablation studies to prove our claims in Sec 4.4.

4.1. Datasets

ScanNet v2. In the task of ScanNet v2 [2], we need to classify all the points into 20 different semantic categories. This dataset provides $1,513$ scanned scenes with point-level annotations, $1,201$ scanned scenes for training, and 312 scanned scenes for validation. Another 100 scanned scenes are published without any annotations for testing. We need to make prediction on the test set and submit our final result to ScanNet server for testing.

S3DIS. S3DIS [1] provides point clouds of 271 rooms with comprehensive annotations in 6 large-scale indoor areas from 3 different buildings. There are 273 million points in total, and all these points are categorized into 13 classes. Following [25, 31], we take Area 5 as the test set and rooms in the remaining areas for training.

Semantic3D. Semantic3D [7] is a large-scale outdoor point cloud dataset with online benchmark. It contains more than 4 billion points from diverse urban scenes, and all the points are classified into 8 categories. The whole dataset includes 15 point clouds for training and another 15 point clouds for testing. For easy evaluation, Semantic3D provides the task of Semantic3D reduced-8, where 15 large-scale point clouds are used for training and 4 down-sampled point clouds are used for testing.

4.2. Implementation

All the experiments can be conducted on a single GTX 1080Ti with 3700X CPU and 64 GB RAM. We apply our method to a common backbone deformable KPConv [31] and evaluate the performance on all three datasets. To show the versatility of our method, we also embed our method into three other backbones (one for each dataset).

ScanNet. We separately choose deformable KPConv [31] and SceneEncoder [37] as our backbones and apply our method. When we take deformable KPConv as our backbone, we randomly sample spheres with radius equal to 2 meters from scenes in the training set during training procedure, and the batch size is set to 10. When we take SceneEncoder as our backbone and train our model, we randomly sample 8 3m×1.5m×1.5m cubes from training scenes for every batch like SceneEncoder [37]. After training, we separately predict the results of the test set using these two trained models and submit them to the online benchmark server for testing [2].

S3DIS. We insert our methods into deformable KPConv [31] and RandLA [9] respectively and treat them as our backbones. When we take deformable KPConv as our backbone, we randomly sample spheres with 2m radius from original point clouds, and the batch size is set to 5. We randomly sample $40,960$ points from entire rooms for each training sample and set the batch size to be 6 when taking RandLA [9] as the backbone. Rooms in Area-1,2,3,4,6 are used for training. After training, we test the model on the whole S3DIS Area-5 set.

Semantic3D. Deformable KPConv and rigid KPConv proposed in [31] are taken as our backbones to evaluate our method on Semantic3D reduced-8 task [7]. Because Semantic3D is a large-scale outdoor space dataset, point cloud is randomly sampled into a sphere with 3m radius for deformable KPConv backbone and 4m radius for rigid KPConv backbone. Every time, 10 samples are fed into the network for training and testing. We need to submit the final predictions to the Semantic3D server for testing [7].

Method	mIoU(%)
PointNet++ (NIPS'17) [26]	33.9
PointCNN (NIPS'18) [21]	45.8
3DMV (ECCV'18) [3]	48.4
PointConv (CVPR'19) [36]	55.6
TextureNet (CVPR'19) [11]	56.6
HPEIN (ICCV'19) [13]	61.8
SPH3D-GCN (TPAMI'20) [20]	61.0
FusionAwareConv (CVPR'20) [41]	63.0
FPConv (CVPR'20) [22]	63.9
DCM-Net (CVPR'20) [27]	65.8
PointASNL (CVPR'20) [38]	66.6
FusionNet (ECCV'20) [40]	68.8
SceneEncoder (IJCAI'20) [37]	62.8
SceneEncoder + Ours	65.9
KPConv *deform* (ICCV'19) [31]	68.4
KPConv *deform* + Ours	**70.2**

Table 1: Results of indoor scene semantic segmentation segmentation on ScanNet v2.

4.3. Metric and Results

Metric. For better evaluation of segmentation performance, we take mean Intersection over Union (mIoU) among categories as our metric like many previous works [4, 25, 31].

The results of semantic segmentation on ScanNet v2 [2] are reported in Table 1, where we achieve 70.2% mIoU and rank first in this benchmark among all point-based methods. Here, we take deformable KPConv as our baseline and 1.8% improvement is achieved in mIoU. To show the generalization ability of our method, we also apply our method to SceneEncoder [37]. As shown in Table 1, 3.1% improvement in mIoU is achieved. Additionally, we provide the qualitative results of our baseline (deformable KPConv) and our method in Figure 3. The red dashed circles indicate the obvious qualitative improvements.

We report the segmentation results on S3DIS Area-5 [1] in Table 2. In this dataset, we also take deformable KPConv as our backbone and achieve 68.73% mIoU in S3DIS Area-5 task which pushes the state-of-the-art performance ahead. Deformable KPConv is also treated as our baseline for its good performance. Meanwhile, we also apply our method to RandLA and the improvement over these backbones is also obvious (i.e., 2.67% mIoU). Figure 4 gives the visualization results of our method and the qualitative improvement over the baseline (deformable KPConv).

In Table 3, we show the results of our method and other prevailing methods on Semantic3D [7]. In this task, we achieve 77.8% in mIoU, outperforming all the state-of-the-art competitors. When taking deformable KPConv as our backbone, our method improves it by 4.7%. Then we take rigid KPConv as our backbone, and our method can also bring 3.0% improvement in mIoU. We present the visual re-

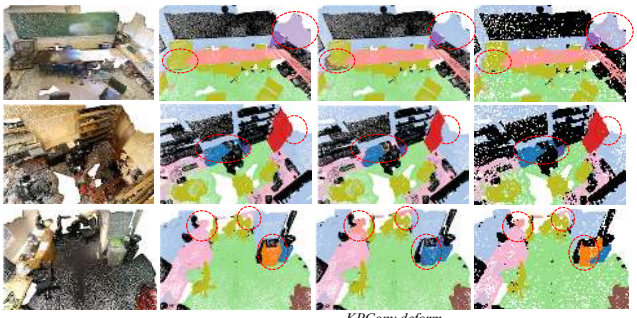

Figure 3: Visualization results on the validation dataset of ScanNet v2. The images from the left to right are input point clouds, semantic labels, predictions given by our baseline and our method, respectively.

Method	mIoU(%)
PointNet (CVPR'17) [25]	41.09
RSNet (CVPR'18) [12]	51.93
PointCNN (NIPS'18) [21]	57.26
ASIS (CVPR'19) [34]	54.48
ELGS (NIPS'19) [33]	60.06
PAT (CVPR'19) [39]	60.07
SPH3D-GCN (TPAMI'20) [20]	59.5
PointASNL (CVPR'20) [38]	62.6
FPConv (CVPR'20) [22]	62.8
Point2Node (AAAI'20) [8]	62.96
SegGCN (CVPR'20) [19]	63.6
DCM-Net (CVPR'20) [27]	64.0
FusionNet (ECCV'20) [40]	67.2
RandLA (CVPR'20) [9]	62.42
RandLA [9] + Ours	65.09
KPConv *deform* (ICCV'19) [31]	67.1
KPConv *deform* + Ours	**68.73**

Table 2: Results of indoor scene semantic segmentation on S3DIS Area-5.

sults of our method and the baseline (deformable KPConv) on the validation set of Semantic3D in Figure 5. The dark blue dashed circles indicate the qualitative improvements.

4.4. Ablation Study

In this section, we conduct more experiments to evaluate the effectiveness of the proposed gradual Receptive Field Component Reasoning (RFCR) method from different aspects. Without loss of generality, our ablation studies are mainly conducted on the task of Semantic3D reduced-8 and deformable KPConv [31] is chosen as backbone.

Gradual Receptive Field Component Reasoning. To conduct ablation studies on different parts of gradual Receptive Field Component Reasoning in the semantic segmentation, we firstly only give the omni-supervision in the decoding procedure to guide the network reason Receptive Field Component Codes (RFCCs) gradually without the loss for Feature Densification (FD). Then, we add the centrifugal potential to obtain more active features for RFCC prediction, and the results are reported in Table 4. The results indicate the Receptive Field Component Reasoning can improve the segmentation performance by 2.9% alone, and FD

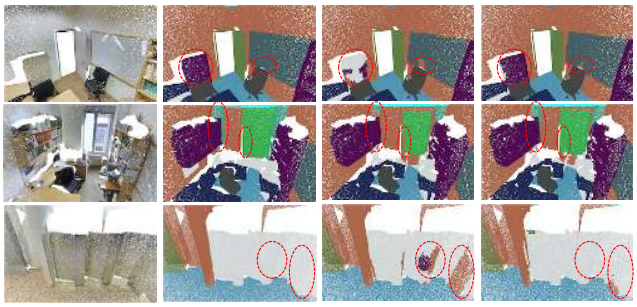

Figure 4: Visualization results on the test dataset of the S3DIS Area-5. The left-most images are input point clouds and the following images are segmentation ground truth, predictions of baseline and our method separately.

Method	mIoU(%)
SegCloud (3DV'17) [29]	61.3
RF_MSSF (3DV'18) [30]	62.7
SPG (CVPR'18) [17]	73.2
ShellNet (ICCV'19) [42]	69.4
GACNet (CVPR'19) [32]	70.8
FGCN (CVPR'20) [15]	62.4
PointGCR (WACV'20) [23]	69.5
RandLA (CVPR'20) [9]	77.4
KPConv *rigid* (ICCV'19) [31]	74.6
KPConv *rigid* + Ours	77.6
KPConv *deform* (ICCV'19) [31]	73.1
KPConv *deform* + Ours	**77.8**

Table 3: Results of outdoor space semantic segmentation on Semantic3D (reduced-8).

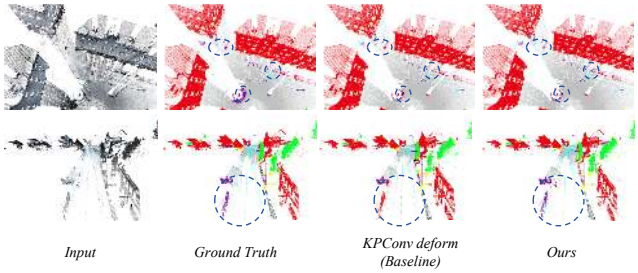

Figure 5: Visualizations on validation set of Semantic3D. Inputs, semantic labels, results of our baseline and our method are presented separately from the left to the right.

Method	mIoU
KPConv *deform*	73.1
+ RFCR	76.0
+ FD	77.8

Table 4: Ablation study on impact of different parts of gradual Receptive Field Component Reasoning.

Method	mIoU
KPConv *deform*	73.1
KPConv *deform* + OvU + FD	76.2
KPConv *deform* + RFCR[one-hot] + FD	76.4
KPConv *deform* + RFCR + FD	77.8

Table 5: Ablation study on omni-scale supervision strategy.

can further bring 1.8% improvement. We also conduct ablation studies on the effects of supervisions at different scales and provide the details in supplementary materials.

Omni-supervision via Up-sampling. Multi-scale supervision is usually used in 2D segmentation via up-sampling the low-resolution prediction. Even we cannot up-sample the point cloud through simple tiling or interpolation, we attempt to up-sample the intermediate predictions iteratively using the nearest neighbors. Then, semantic labels are used to supervise all the up-sampled predictions. Same as our method, all scales are supervised and Feature Densification is also used to provide more unambiguous features for intermediate prediction. We report the result of Omni-supervision via Up-sampling (OvU) in Table 5 and compare it with our method. It shows inferior performance (76.2%) because the up-sampling method using nearest neighbors cannot trace the proper encoding relationship.

One-hot RFCC. In previous works like PointRend [16], they give one-hot predictions at low resolutions, and these predictions will be up-sampled to be supervised by the one-hot labels at original resolution. So, it is intuitive to take an one-hot RFCC for the major category in the receptive field to supervise the prediction. However, the category information of some points will be ignored in this way. Compared with this method, we take a multi-hot label for every sampled point at all the scales, and no labels will be ignored in the supervision of down-sampled points. In order to show the benefit of multi-hot labels, we replace the multi-hot labels with one-hot labels which represent the majority of categories in the receptive fields, and all other settings remain the same. We report the results in Table 5. We can see one-hot RFCC which ignores the minor category cannot fully represent the information in the receptive field, thus having sub-optimal performance (76.4%) in the segmentation which is 1.4% lower than multi-hot RFCC.

Feature Densification. As stated in Sec 3.4, active features will be densified by centrifugal potential given the loss in Eq. (12). The distribution of features' magnitude after training can be visualized by the bar chart shown in Figure 6. As indicated in this figure, features are pushed away from 0 and more unambiguous features are available for the Receptive Field Component Reasoning, thus improving the segmentation performance (Table 4).

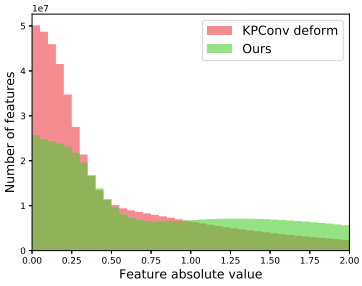

Figure 6: Visualization of features' magnitude in the decoding layers. The green chart bars represent the distribution of features' absolute value after adding Feature Densification while the red chart bars represent the distribution of features' absolute value in the original network.

5. Conclusion

In this paper, we propose a gradual Receptive Field Component Reasoning method for omni-supervised point cloud segmentation which decomposes the hard segmentation problem into a global context recognition task and a series of gradual Receptive Field Component Code reasoning steps. Additionally, we propose a complementary Feature Densification method to provide more active features for RFCC prediction. We evaluate our method with four prevailing backbones on three popular benchmarks and outperform almost all the state-of-the-art point-based competitors. Furthermore, our method brings new state-of-the-art performance for Semantic3D and S3DIS benchmarks. Even our method brings large improvements to many backbones for point cloud segmentation, it is more suitable for networks with encoder-decoder architecture.

6. Acknowledgments

This work is sponsored by National Natural Science Foundation of China (61972157, 61772524, 61876161, 61902129), Natural Science Foundation of Shanghai (20ZR1417700), CAAI-Huawei MindSpore Open Fund, National Key Research and Development Program of China (2019YFC1521104, 2020AAA0108301), Zhejiang Lab (No. 2019KD0AC02, 2020NB0AB01).

References

[1] Iro Armeni, Ozan Sener, Amir R Zamir, Helen Jiang, Ioannis Brilakis, Martin Fischer, and Silvio Savarese. 3d semantic parsing of large-scale indoor spaces. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1534–1543, 2016.

[2] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5828–5839, 2017.

[3] Angela Dai and Matthias Nießner. 3dmv: Joint 3d-multi-view prediction for 3d semantic scene segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 452–468, 2018.

[4] Jingyu Gong, Jiachen Xu, Xin Tan, Jie Zhou, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Boundary-aware geometric encoding for semantic segmentation of point clouds. In *AAAI*, 2021.

[5] Benjamin Graham, Martin Engelcke, and Laurens van der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9224–9232, 2018.

[6] Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. In *Advances in neural information processing systems(NeurIPS)*, pages 529–536, 2005.

[7] Timo Hackel, Nikolay Savinov, Lubor Ladicky, Jan D Wegner, Konrad Schindler, and Marc Pollefeys. Semantic3d. net: A new large-scale point cloud classification benchmark. *arXiv preprint arXiv:1704.03847*, 2017.

[8] Wenkai Han, Chenglu Wen, Cheng Wang, Xin Li, and Qing Li. Point2node: Correlation learning of dynamic-node for point cloud feature modeling. In *Thirty-fourth AAAI conference on artificial intelligence*, 2020.

[9] Qingyong Hu, Bo Yang, Linhai Xie, Stefano Rosa, Yulan Guo, Zhihua Wang, Niki Trigoni, and Andrew Markham. Randla-net: Efficient semantic segmentation of large-scale point clouds. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[10] Zeyu Hu, Mingmin Zhen, Xuyang Bai, Hongbo Fu, and Chiew-lan Tai. Jsenet: Joint semantic segmentation and edge detection network for 3d point clouds. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[11] Jingwei Huang, Haotian Zhang, Li Yi, Thomas Funkhouser, Matthias Nießner, and Leonidas J Guibas. Texturenet: Consistent local parametrizations for learning from high-resolution signals on meshes. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4440–4449, 2019.

[12] Qiangui Huang, Weiyue Wang, and Ulrich Neumann. Recurrent slice networks for 3d segmentation of point clouds. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2626–2635, 2018.

[13] Li Jiang, Hengshuang Zhao, Shu Liu, Xiaoyong Shen, Chi-Wing Fu, and Jiaya Jia. Hierarchical point-edge interaction network for point cloud semantic segmentation. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 10433–10441, 2019.

[14] Lipeng Ke, Ming-Ching Chang, Honggang Qi, and Siwei Lyu. Multi-scale structure-aware network for human pose estimation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 713–728, 2018.

[15] Saqib Ali Khan, Yilei Shi, Muhammad Shahzad, and Xiao Xiang Zhu. Fgcn: Deep feature-based graph convolutional network for semantic segmentation of urban 3d point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[16] Alexander Kirillov, Yuxin Wu, Kaiming He, and Ross Girshick. Pointrend: Image segmentation as rendering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[17] Loic Landrieu and Martin Simonovsky. Large-scale point cloud semantic segmentation with superpoint graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4558–4567, 2018.

[18] Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, 2013.

[19] Huan Lei, Naveed Akhtar, and Ajmal Mian. Seggcn: Efficient 3d point cloud segmentation with fuzzy spherical kernel. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11611–11620, 2020.

[20] Huan Lei, Naveed Akhtar, and Ajmal Mian. Spherical kernel for efficient graph convolution on 3d point clouds. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

[21] Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 820–830, 2018.

[22] Yiqun Lin, Zizheng Yan, Haibin Huang, Dong Du, Ligang Liu, Shuguang Cui, and Xiaoguang Han. Fpconv: Learning local flattening for point convolution. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4293–4302, 2020.

[23] Yanni Ma, Yulan Guo, Hao Liu, Yinjie Lei, and Gongjian Wen. Global context reasoning for semantic segmentation of 3d point clouds. In *The IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2931–2940, 2020.

[24] Quang-Hieu Pham, Thanh Nguyen, Binh-Son Hua, Gemma Roig, and Sai-Kit Yeung. Jsis3d: joint semantic-instance segmentation of 3d point clouds with multi-task pointwise networks and multi-value conditional random fields. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8827–8836, 2019.

[25] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 652–660, 2017.

[26] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems(NeurIPS)*, pages 5099–5108, 2017.

[27] Jonas Schult, Francis Engelmann, Theodora Kontogianni, and Bastian Leibe. Dualconvmesh-net: Joint geodesic and euclidean convolutions on 3d meshes. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[28] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *Advances in neural information processing systems (NeurIPS)*, pages 3483–3491, 2015.

[29] Lyne Tchapmi, Christopher Choy, Iro Armeni, JunYoung Gwak, and Silvio Savarese. Segcloud: Semantic segmentation of 3d point clouds. In *2017 international conference on 3D vision (3DV)*, pages 537–547. IEEE, 2017.

[30] Hugues Thomas, François Goulette, Jean-Emmanuel Deschaud, Beatriz Marcotegui, and Yann LeGall. Semantic classification of 3d point clouds with multiscale spherical neighborhoods. In *2018 International Conference on 3D Vision (3DV)*, pages 390–398. IEEE, 2018.

[31] Hugues Thomas, Charles R Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, François Goulette, and Leonidas J Guibas. Kpconv: Flexible and deformable convolution for point clouds. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 6411–6420, 2019.

[32] Lei Wang, Yuchun Huang, Yaolin Hou, Shenman Zhang, and Jie Shan. Graph attention convolution for point cloud semantic segmentation. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10296–10305, 2019.

[33] Xu Wang, Jingming He, and Lin Ma. Exploiting local and global structure for point cloud semantic segmentation with contextual point representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 4571–4581, 2019.

[34] Xinlong Wang, Shu Liu, Xiaoyong Shen, Chunhua Shen, and Jiaya Jia. Associatively segmenting instances and semantics in point clouds. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4096–4105, 2019.

[35] Shih-En Wei, Varun Ramakrishna, Takeo Kanade, and Yaser Sheikh. Convolutional pose machines. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4724–4732, 2016.

[36] Wenxuan Wu, Zhongang Qi, and Li Fuxin. Pointconv: Deep convolutional networks on 3d point clouds. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9621–9630, 2019.

[37] Jiachen Xu, Jingyu Gong, Jie Zhou, Xin Tan, Yuan Xie, and Lizhuang Ma. Sceneencoder: Scene-aware semantic segmentation of point clouds with a learnable scene descriptor. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI)*, 2020.

[38] Xu Yan, Chaoda Zheng, Zhen Li, Sheng Wang, and Shuguang Cui. Pointasnl: Robust point clouds processing using nonlocal neural networks with adaptive sampling. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[39] Jiancheng Yang, Qiang Zhang, Bingbing Ni, Linguo Li, Jinxian Liu, Mengdie Zhou, and Qi Tian. Modeling point clouds with self-attention and gumbel subset sampling. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3323–3332, 2019.

[40] Feihu Zhang, Jin Fang, Benjamin Wah, and Philip Torr. Deep fusionnet for point cloud semantic segmentation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[41] Jiazhao Zhang, Chenyang Zhu, Lintao Zheng, and Kai Xu. Fusion-aware point convolution for online semantic 3d scene segmentation. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4534–4543, 2020.

[42] Zhiyuan Zhang, Binh-Son Hua, and Sai-Kit Yeung. Shellnet: Efficient point cloud convolutional neural networks using concentric shells statistics. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 1607–1616, 2019.

[43] Lin Zhao and Wenbing Tao. Jsnet: Joint instance and semantic segmentation of 3d point clouds. In *AAAI*, pages 12951–12958, 2020.

[44] Shuai Zhao, Yang Wang, Zheng Yang, and Deng Cai. Region mutual information loss for semantic segmentation. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 11117–11127, 2019.