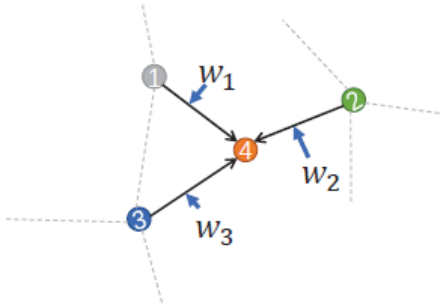
中国科学技术大学
University of Science and Technology of China

PointVector: A Vector Representation In Point Cloud Analysis

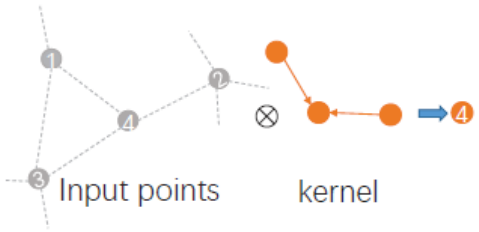
Xin Deng,WenYu Zhang,Qing Ding,XinMing Zhang

Paper ID 7469
WED-AM-117

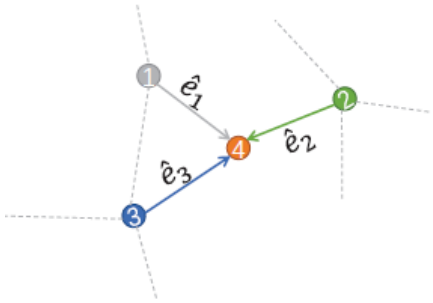
Quick Preview



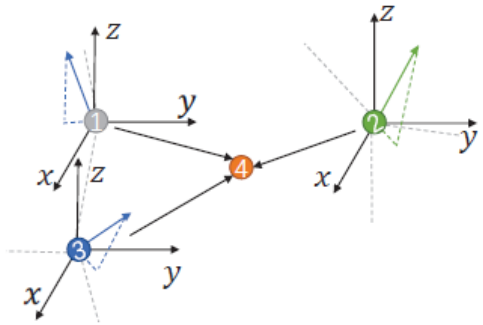
(a) Attention



(b) Templated-based method



(c) Dynamic Conv



(d) Vector

In contrast to other dynamic aggregation methods, our approach introduces a novel representation that guides the process of feature aggregation.

The primary focus of our work lies in enhancing the local features of the standard MLP.

Quick Preview

$$f_i * w = w f_i, i = 0 \cdots c$$

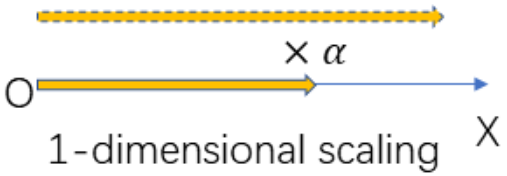
$$\begin{bmatrix} f_i & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} w & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} = \begin{bmatrix} w f_i & 0 & \cdots & 0 \end{bmatrix}$$

$$\begin{pmatrix} f_i & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} q & t & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} = \begin{pmatrix} f_i q & f_i t & \cdots & 0 \end{pmatrix}$$

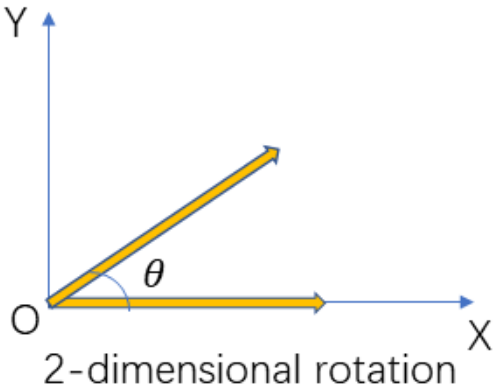
Naturally, it is reasonable to imagine that each feature component is transformed in a high-dimensional space.

Our motivation stems from the weighted summation mechanism employed in MLPs, which is commonly used in methods for weight assignment. We consider the input feature vector component as a high-dimensional vector containing only a single non-zero value.

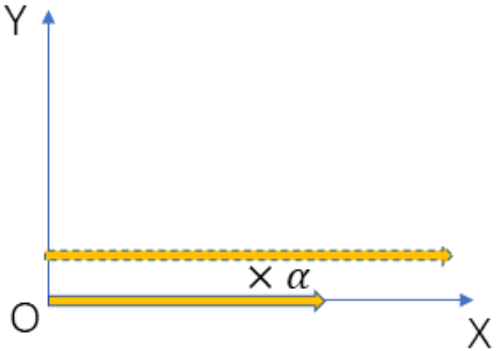
Quick Preview



1-dimensional scaling



2-dimensional rotation



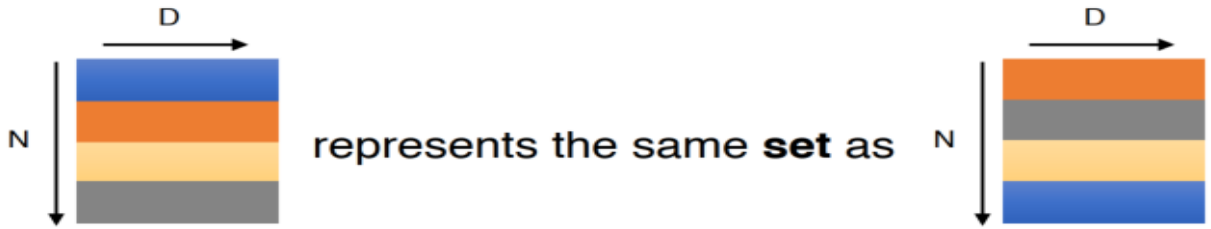
2-dimensional scaling

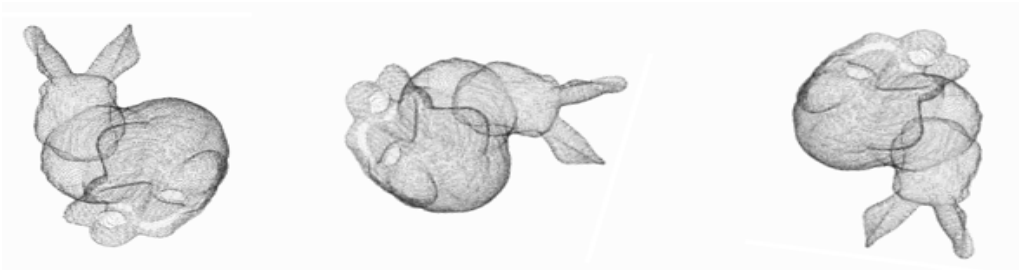
It can be observed that the 1D vector transformation solely involves scale transformation, whereas the 2D vector transformation introduces an additional rotation operation.

Preliminary

Given an unordered point set $\{x_1, x_2, ..., x_n\}$ with $x_i \in \mathbb{R}^d$, one can define a set function $f : \mathcal{X} \to \mathbb{R}$ that maps a set of points to a vector:

$$f(x_1, x_2, ..., x_n) = \gamma \left(\underset{i=1,...,n}{\text{MAX}} \{h(x_i)\} \right) \quad (1)$$





These observations highlight the requirements that must be met for effective point cloud feature extraction. Consequently, we adopt a similar component to PointNet, as it naturally fulfills these requirements.

QI C R, SU H, MO K, et al. Pointnet: Deep learning on point sets for 3d classification and segmentation. IEEE Conference on Computer Vision and Pattern Recognition, 2017:77 - 85.

Preliminary

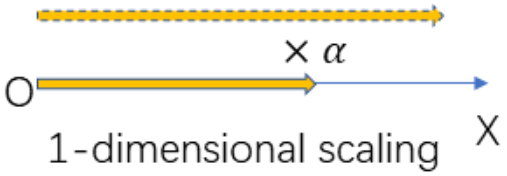
$$f_i * w = w f_i, i = 0 \cdots c$$

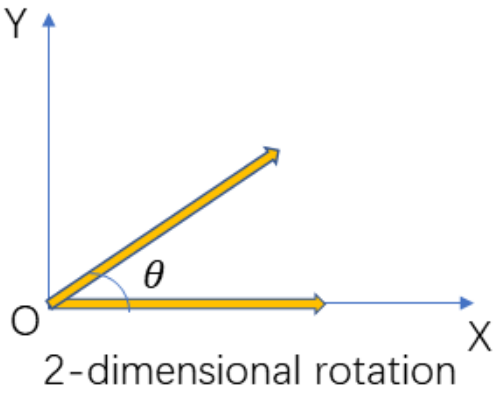
$$\begin{bmatrix} f_i & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} w & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{bmatrix} = \begin{bmatrix} w f_i & 0 & \cdots & 0 \end{bmatrix}$$

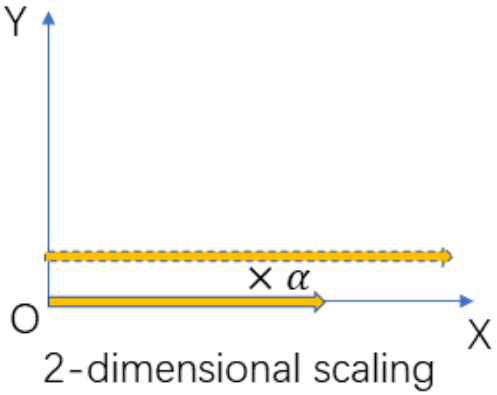
$$\begin{pmatrix} f_i & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} q & t & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 \end{pmatrix} = \begin{pmatrix} f_i q & f_i t & \cdots & 0 \end{pmatrix}$$

It is intuitive to perceive each channel value in an n-dimensional feature vector as a coordinate value along a specific axis, akin to a basis vector pointing in the direction of that axis.

Method



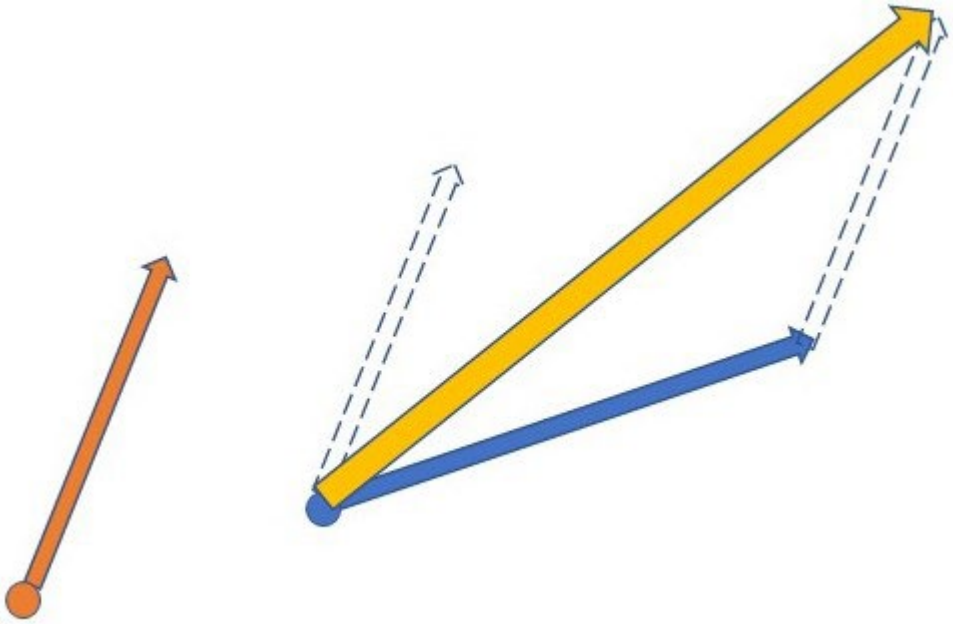




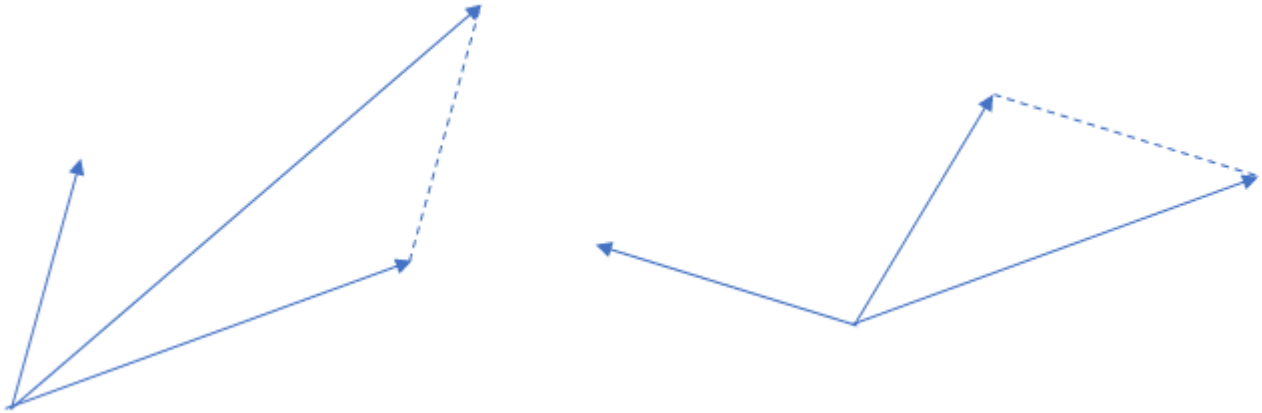
By interpreting the weighted summation from the vector perspective mentioned earlier, it is reasonable to deduce that higher-dimensional vector transformations can lead to more intricate and sophisticated representations.

Higher-dimensional vectors naturally accommodate a wider range of transformations, and provide a more effective means of representing intricate neighbor relationships.

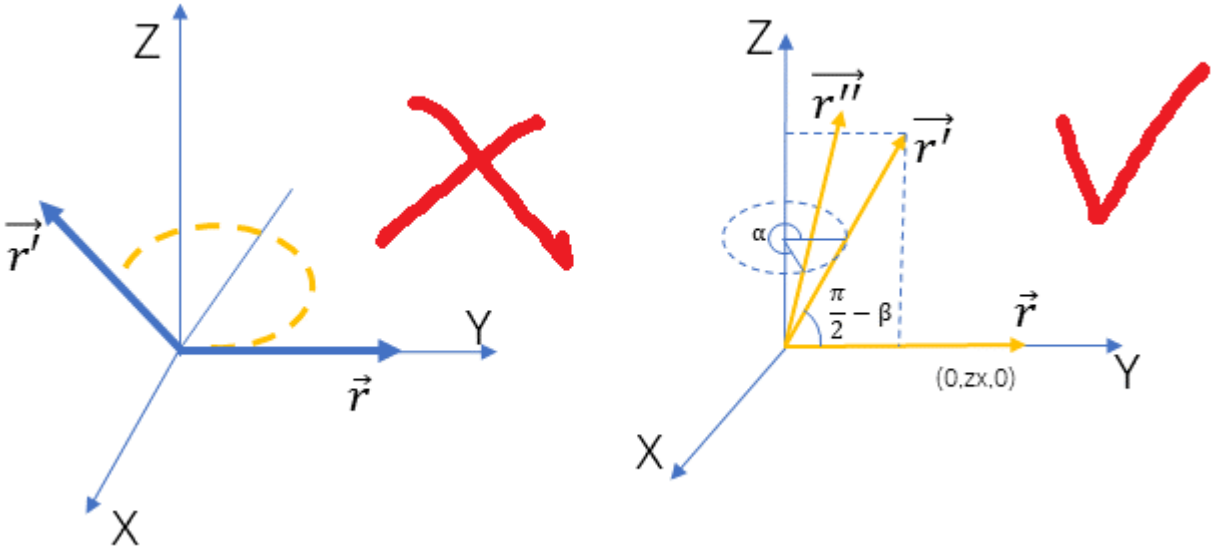
Method



The challenge of reconciling the relationships between neighboring features using scalar weights is transformed into a problem of vector summation. This approach enables the incorporation of homogeneous promotion, reverse inhibition, and vector orientation, facilitating a more expressive representation of their relationships.



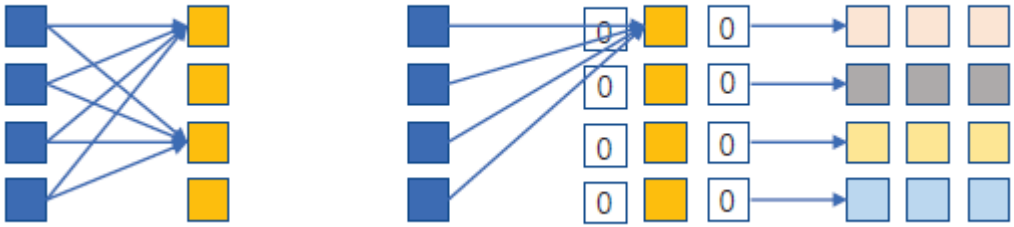
Method

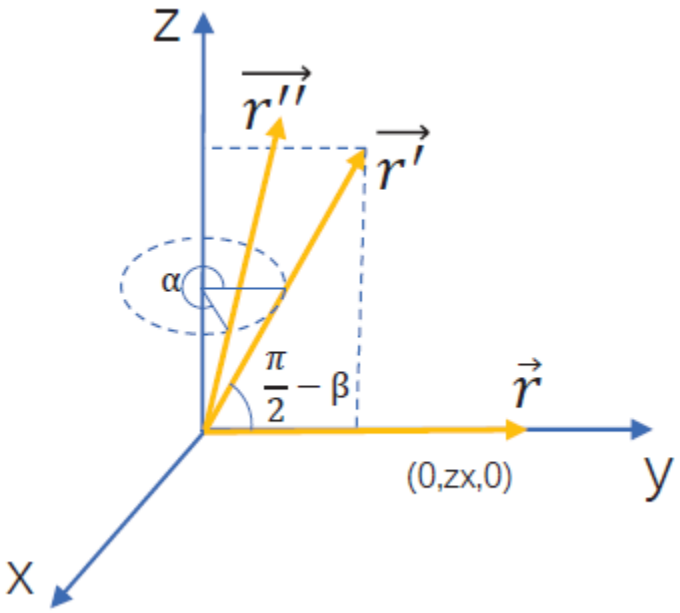


The vector transformation consists of a scaling transformation and a rotation transformation.

The transformation of 3D rotation involves rotating around a specific axis, necessitating the determination of both the rotation axis and the rotation angle. In accordance with Euler's theorem, this rotation can be decomposed into three successive rotations around orthogonal axes.

Method





Since we assume that the feature components lie on the coordinate axes, a rotation parameter can be omitted.

Method

$$
\begin{aligned}
\overrightarrow{r''} &= Rot_z\, Rot_x \vec{r} \\
&= \begin{bmatrix} cos(\alpha) & -sin(\alpha) & 0 \\ sin(\alpha) & cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & sin(\beta) & -cos(\beta) \\ 0 & cos(\beta) & sin(\beta) \end{bmatrix} \begin{bmatrix} 0 \\ zx \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} cos(\alpha) & -sin(\alpha)sin(\beta) & sin(\alpha)cos(\beta) \\ sin(\alpha) & cos(\alpha)sin(\beta) & -cos(\alpha)cos(\beta) \\ 0 & cos(\beta) & sin(\beta) \end{bmatrix} \begin{bmatrix} 0 \\ zx \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} -zx \cdot sin(\alpha)sin(\beta) \\ zx \cdot cos(\alpha)sin(\beta) \\ zx \cdot cos(\beta) \end{bmatrix},
\end{aligned} \tag{5}
$$

$$
\begin{aligned}
zx_j &= Linear(fp_j) \\
[\alpha_j, \beta_j] &= Relu(BN(Linear([fp_j]))),
\end{aligned} \tag{6}
$$

Since the vector transformation alone is completely linear, we choose to add relu to increase the nonlinear factor when predicting the rotation angle.

Method

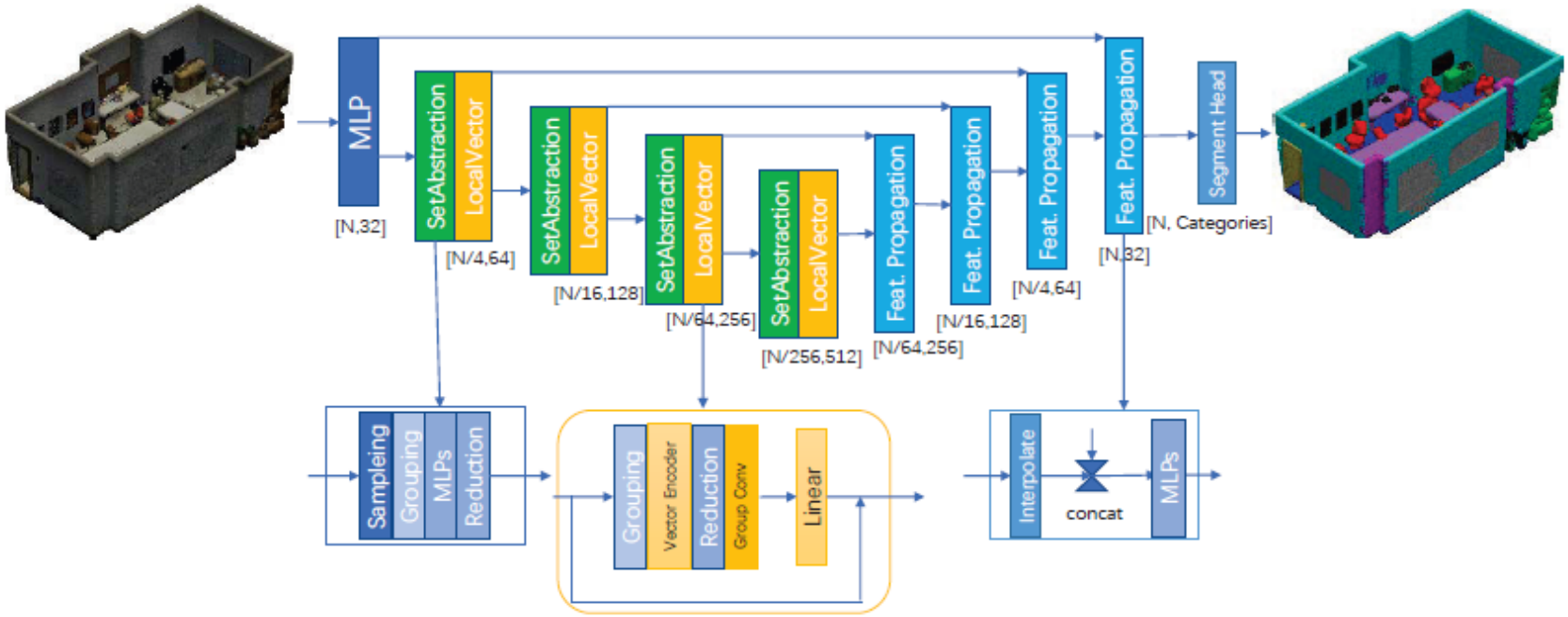


Figure 5. Overall Architecture. We reuse the SA module and Feature Propagation module of PointNet++ and propose the VPSA module to improve the feature extraction of sampled point clouds.

- PointVector-S: C=32, S=0, V=[1,1,1,1]
- PointVector-L: C=32, S=[1,1,1,1], V=[2,4,2,2]
- PointVector-XL: C=64, S=[1,1,1,1], V=[3,6,3,3]

We denote C as the channel of embedding MLP in the beginning, S as the numbers of the SA module, V as the numbers of the VPSA module.

Results

Method	OA %	mAcc %	mIOU %	Params M	FLOPs G	Throughput (ins./sec.)
PointNet [25]	78.5	66.2	47.6	3.6	35.5	162
PointCNN [17]	88.1	75.6	65.4	0.6	-	-
DGCNN [39]	84.1	-	56.1	1.3	-	8
DeepGCN [16]	85.9	-	60.0	3.6	-	3
KPConv [36]	-	79.1	70.6	15.0	-	30
RandLA-Net [12]	88.0	82.0	70.0	1.3	5.8	159
Point Transformer [53]	90.2	81.9	73.5	7.8	5.6	34
CBL [34]	89.6	79.4	73.1	18.6	-	-
RepSurf [30]	90.9	82.6	74.3	0.976	-	-
PointNet++ [26]	81.0	67.1	54.5	1.0	7.2	186
PointNeXt-L [28]	89.8	82.2	73.9	7.1	15.2	115
PointNeXt-XL [28]	90.3	83.0	74.9	41.6	84.8	46
PointVector-L	**91.4**	**85.5**	**77.4**	4.2	10.7	98
PointVector-XL(Ours)	**91.9**	**86.1**	**78.4**	24.1	58.5	40

Table 1. **Semantic segmentation on S3DIS with 6-fold cross-validation.** Methods are in chronological order. The highest and second scores are marked in bold.

The utilization of vector-guided feature aggregation results in a substantial performance enhancement, accompanied by a notable reduction in both parameter count and computational requirements. However, due to optimization challenges, it exhibits a slight lag in terms of speed.

Results

Method	OA %	mAcc %	mIoU %	ceiling	floor	wall	beam	column	window	door	table	chair	sofa	bookcase	board	clutter
PointNet [25]	-	49.0	41.1	88.8	97.3	69.8	0.1	3.9	46.3	10.8	59.0	52.6	5.9	40.3	26.4	33.2
PointCNN [17]	85.9	63.9	57.3	92.3	98.2	79.4	0.0	17.6	22.8	62.1	74.4	80.6	31.7	66.7	62.1	56.7
DGCNN [39]	83.6	-	47.9	-	-	-	-	-	-	-	-	-	-	-	-	-
DeepGCN [16]	-	-	52.5	-	-	-	-	-	-	-	-	-	-	-	-	-
KPConv [36]	-	72.8	67.1	92.8	97.3	82.4	0.0	23.9	58.0	69.0	81.5	91.0	75.4	75.3	66.7	58.9
PVCNN [22]	87.1	-	59.0	-	-	-	-	-	-	-	-	-	-	-	-	-
PAConv [44]	-	73.0	66.6	94.6	**98.6**	82.4	0.0	26.4	58.0	60.0	89.7	80.4	74.3	69.8	73.5	57.7
ASSANet-L [27]	-	-	66.8	-	-	-	-	-	-	-	-	-	-	-	-	-
Point Transformer [52]	90.8	76.5	70.4	94.0	98.5	**86.3**	0.0	38.0	**63.4**	74.3	89.1	82.4	74.3	**80.2**	76.0	59.3
PatchFormer [51]	-	-	68.1	-	-	-	-	-	-	-	-	-	-	-	-	-
CBL [34]	90.6	75.2	69.4	93.9	98.4	84.2	0.0	37.0	57.7	71.9	**91.7**	81.8	77.8	75.6	69.1	62.9
RepSurf-U [30]	90.2	76.0	68.9	-	-	-	-	-	-	-	-	-	-	-	-	-
StratifiedFormer* [14]	**91.5**	**78.1**	**72.0**	**96.2**	**98.7**	**85.6**	0.0	**46.1**	60.0	**76.8**	**92.6**	84.5	77.8	75.2	78.1	**64.0**
PointNet++ [26]	83.0	-	53.5	-	-	-	-	-	-	-	-	-	-	-	-	-
PointNeXt-L [28]	90.1	76.1	69.5	94.0	98.5	83.5	0.0	30.3	57.3	74.2	82.1	91.2	74.5	75.5	76.7	58.9
PointNeXt-XL [28]	90.7	77.5	70.8	94.2	98.5	84.4	0.0	37.7	59.3	74.0	83.1	91.6	77.4	77.2	78.8	60.6
PointVector-L(Ours)	90.8	77.3	71.2	94.8	98.2	84.1	0.0	31.7	60.0	**77.7**	83.7	**91.9**	**81.8**	**78.9**	**79.9**	**63.3**
PointVector-XL(Ours)	**91.0**	**78.1**	**72.3**	**95.1**	**98.6**	85.1	0.0	**41.4**	**60.8**	76.7	84.4	**92.1**	**82.0**	77.2	**85.1**	61.4

Table 2. **Semantic segmentation on S3DIS Area5.** * denotes StratifiedFormer use 80k points as input points. The highest and second scores are marked in bold.

Although our setup differs significantly from StratifiedFormer, we still maintain a slight advantage over it.

Validation results for other benchmark datasets and ablation experiments are detailed in the paper.

Conclusion

中国科学技术大学
University of Science and Technology of China

1. We propose a new perspective on the weighted summation operation.

2. Indeed, we propose the adoption of intermediate vectors to represent the aggregation of neighboring features, guided by the direction indicated by the vector.

3. For the vector transformation operation, we propose a construction method for the rotation matrix utilizing independent rotation angles, in accordance with Euler's theorem.

Limitation

The speed of our approach is constrained by the grouped convolution implementation.

we have not delved into deconstructing the representation of rotation in four-dimensional space, which becomes more intricate, particularly within the plane.

Additionally, summing aftercomponent alignment aligns with our assumptions better than scalar projection. As shown below, each channel should be aligned and then summed directly.