PointSIFT

A SIFT-like Network Module for 3D Point Cloud Semantic Segmentation

2018/12/16 潘云逸

SIFT (Scale-invariant feature transform)

- SIFT 特征点位置和尺度的提取：
  - 将相同 size 的相邻高斯尺度之间的灰度图像进行减法运算，进而得到高斯差分图像。可以看得出其边缘特征比较明显。
  - 然后在高斯差分图像上检测特征点。
- SIFT 特征点方向的提取：
  - 选择好特征点之后，还需要提取出特征点方向信息。
  - 在特征点所在的 系数 * 高斯尺度（ σ ）为长宽的正方形区域内，求像素点之间梯度变化的方向。
  - 将方向划分到以 45° 为间隔的 8 个方向内进行统计， 最多的方向则为该特征点的主方向。
- SIFT 特征提取汇总：
  - 有了特征点的位置、尺度、方向三个信息，之后再依据尺度、方向对特征点进行描述，则特征点的特征向量将具有尺度不变性和旋转不变性。
- SIFT 特征描述：
  - 以特征点的主方向作为特征描述的 X 轴，在其坐标系的四个象限上分别划出 2*2 个小格子，每个小格子分别对格子中的灰度变化方向进行统计。
  - 每个小格子统计出来一个按照 8 个方向划分， 8 个方向的数量 归一化后的结果，成为一个 8 维的向量。一共 4 个象限即 4*4 个小格子， 4*4*8=128 ， 最终 SIFT 的特征点将用 128 维向量表示。

SIFT (scale-invariant feature transform)

- 特征点方向 - 旋转不变性

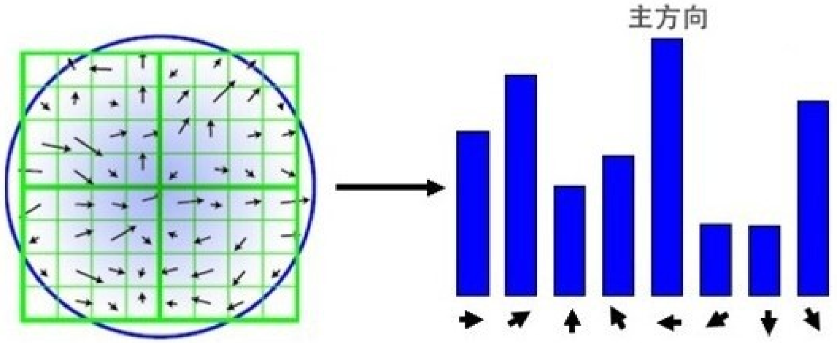


图 5.1 关键点方向直方图

特征点三个信息

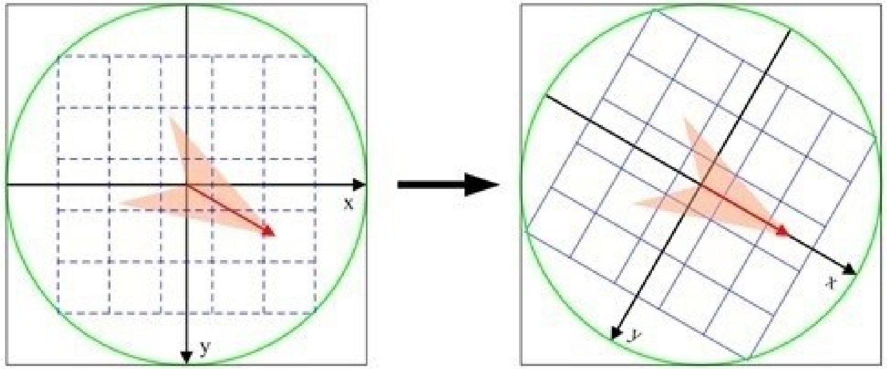
位置

尺度 - 尺度不变性

方向 - 旋转不变性

SIFT (scale-invariant feature transform)

- 特征描述



6.2 坐标轴旋转

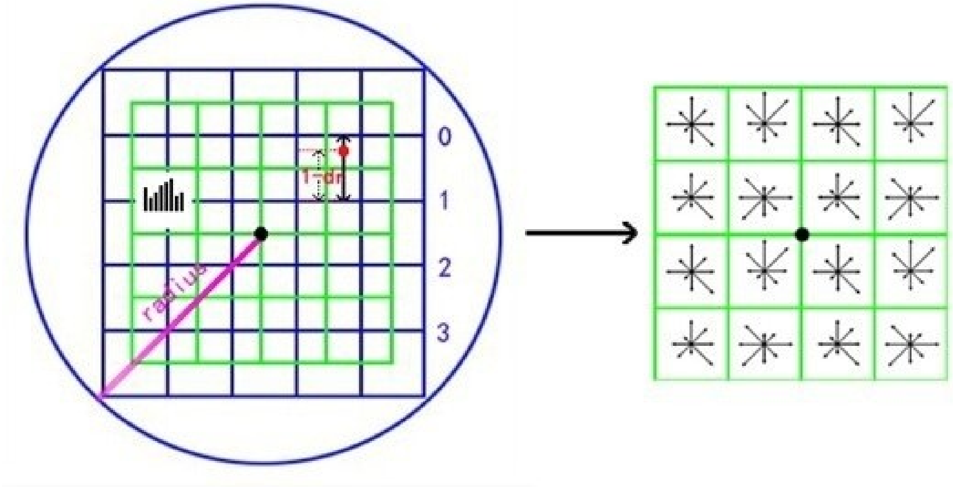


图 6.3 描述字梯度直方图

4*4*8 = 128 维向量

Motivation

- sparseness of point cloud in 3D space make most spatial operators inefficient
- the relationship between points is implicit and difficult to be represented due to the uno rdered and unstructured property of point cloud
- SIFT does a good job in 2D images
- PointNet architecture directly operates on point cloud
- PointNet lost the local feature information
- PointNet++use the Set Abstrction module to get the local feature information， but still lost some information （ local centroids using kNN to find neighbors ）

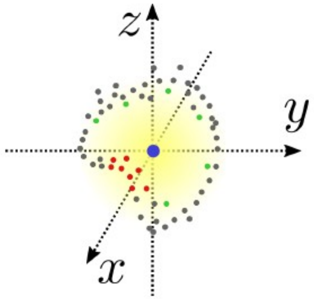


Figure 7. In this case, using K nearest neighbors, all chosen points are from one direction (red points). If we select points in different directions (green points), the representation ability will be better.

Figure 1

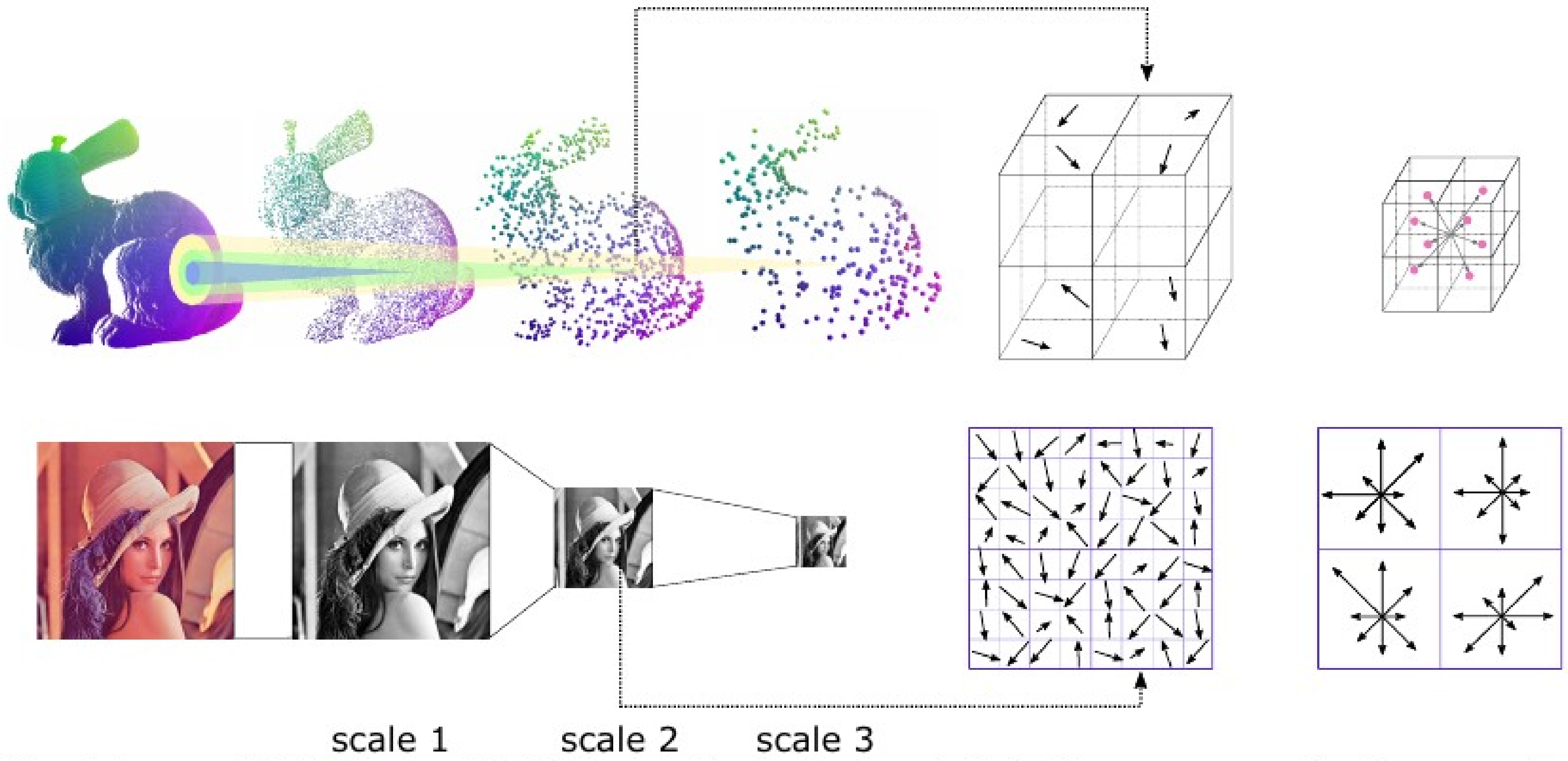


Figure 1. Structure of SIFT [15] and our PointSIFT module. The left side shows that both of them can capture multi-scale patterns and is adaptive to various scales. The right side shows that orientation is encoded in each key point/pixel.

Problem Statement

- Point cloud semantic segmentation
  - Input:
    - a point set containing n points p1,p2,...,pn with d dimensional feature(its cordinate(x,y,z)in 3D space and RGB values etc.)
  - output:
    - assign semantic labels to each point in the point cloud

$$\Psi : P \rightarrow L^n$$

SA and FP module in PointNet++

- SA module: finding N' centroids with farthest point sampling to downsample
  - input: a point cloud of N points and d dimensional feature each point.
  - output: N' downsampled points each with d' dimensional feature.
- FP module: linear interpolation weighted by distances to upsample the point cloud
  - input:N points
  - output:N' points

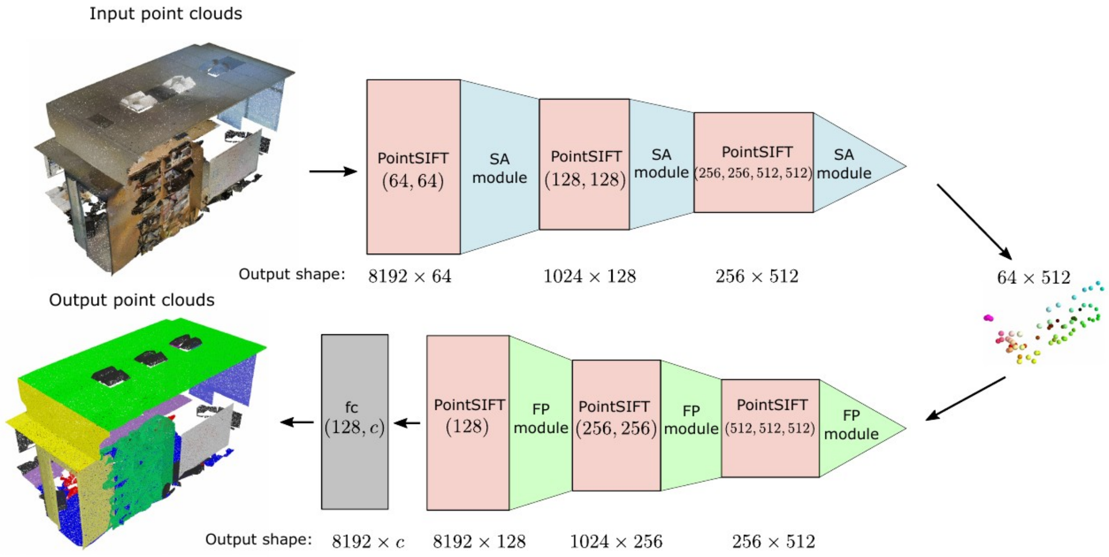


Figure 2. Illustration of our two-stage network architecture. The network consists of downsampling (set abstraction) and upsampling (feature propagation) procedures. PointSIFT modules (marked in red) are interleaved with downsampling (marked in blue) and upsampling (marked in green) layers. Both SA and FP module are introduced in [24]. The FP-shortcuts are not shown in the figure for better clarity. PointSIFT$(\cdot)$ specifies feature dimensionalities of each orientation-encoding(OE) units, for example, PointSIFT$(64, 64)$ stands for two stacked OE units both having 64 output feature channels. The number beneath layers is the shape of output point set of corresponding layers, for example, 8192×64 means 8192 points with 64 feature channel each point.

Method

1.Structure:

a encode-decode(downsample-upsample)framework similar to general semantic segmentation network for point cloud segmentation

2.Component of PointSIFT:

1.orientation-encoding
2.scale-awareness

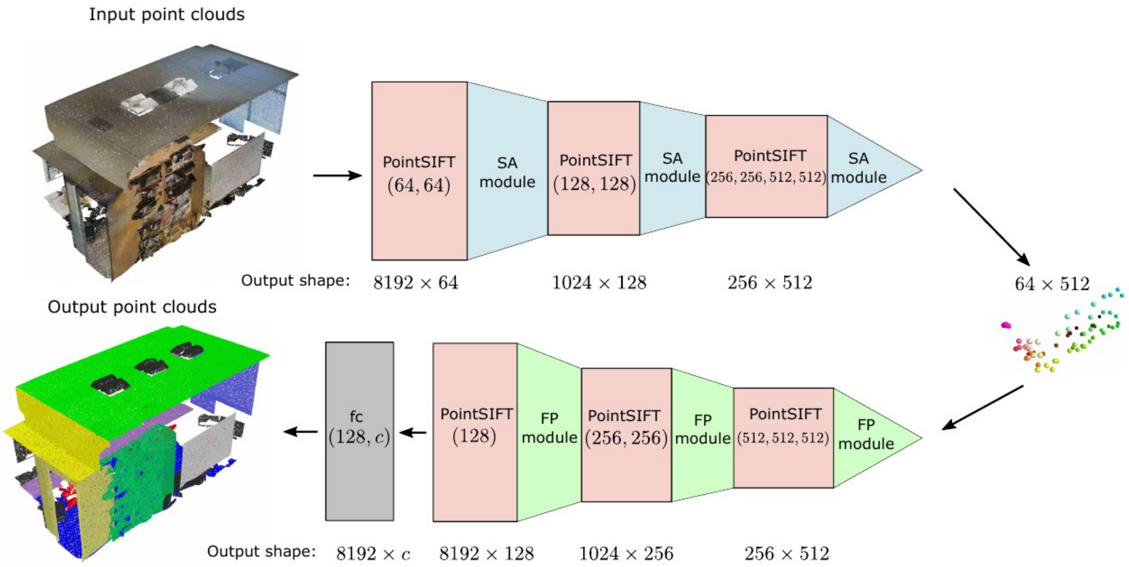
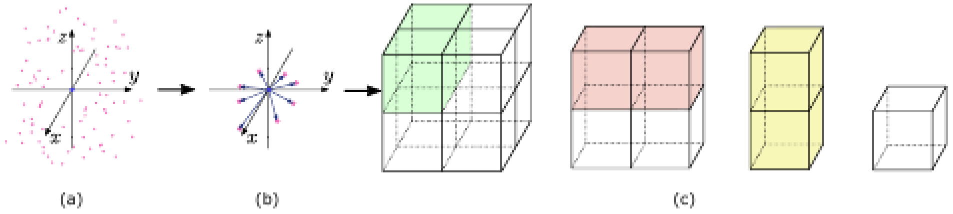


Figure 2. Illustration of our two-stage network architecture. The network consists of downsampling (set abstraction) and upsampling (feature propagation) procedures. PointSIFT modules (marked in red) are interleaved with downsampling (marked in blue) and upsampling (marked in green) layers. Both SA and FP module are introduced in [24]. The FP-shortcuts are not shown in the figure for better clarity. PointSIFT$(\cdot)$ specifies feature dimensionalities of each orientation-encoding(OE) units, for example, PointSIFT$(64, 64)$ stands for two stacked OE units both having 64 output feature channels. The number beneath layers is the shape of output point set of corresponding layers, for example, 8192×64 means 8192 points with 64 feature channel each point.

Orientation-encoding Unit

- convolves the features of nearest points in 8 orientations
  - input: d-dimension feature vector
    - 1.S8N(Stacked 8-neighborhood)
      - Search which finds nearest neighbors in each of the eight octants partitioned by ordering of three c oordinates.
      - output: 2*2*2 cube for local pattern description centering at p0.
    - 2.OEC(Orientation-encoding convolution)
      - which is a three-stage operator that convolves the 2 × 2 × 2 cube along X, Y , and Z axis successively.
      - output: a d-dimension feature by reshaping $V_{xyz} \in \mathbb{R}_{1\times1\times1\times d}$.
  - output:d-dimension feature vector



$$V_x = g(Conv(W_x, V)) \in \mathbb{R}_{1\times2\times2\times d}$$
$$V_{xy} = g(Conv(W_y, V_x)) \in \mathbb{R}_{1\times1\times2\times d}$$
$$V_{xyz} = g(Conv(W_z, V_{xy})) \in \mathbb{R}_{1\times1\times1\times d}$$

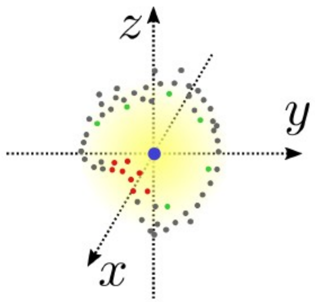


Figure 7. In this case, using K nearest neighbors, all chosen points are from one direction (red points). If we select points in different directions (green points), the representation ability will be better.

Scale-awareness

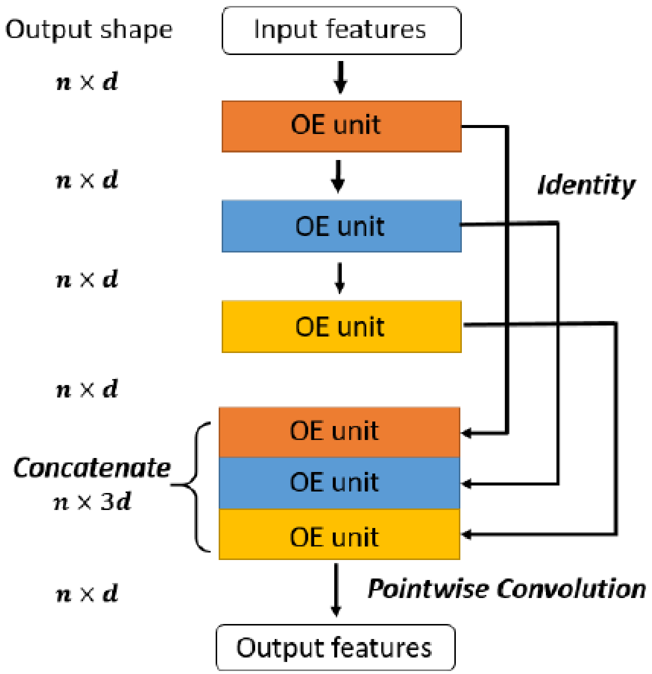


Figure 4. PointSIFT Module. Input features first pass through a series of Orientation-encoding(OE) layers, then outputs of OE units are concatenated and transformed by another point-wise convolution to obtain multi-scale feature.

1. 一个 OE 单元仅仅捕获了 8 个近邻的小尺度信息。但是如多堆叠多个 OE 单元来搭建深度网络。
2. 最后一个可以观察到如图 1 所示的大规模 3D 区域，并且不同的堆叠单元代表不同的尺度。
3. 理想情况下，在第 i 个堆叠单元中，神经元可以观察到 8^i 个点。因此，我们有机会选择合适的比例。
4. 我们的策略是通过快捷方式连接不同堆叠单元的输出。 因此，我们将比例选择留给神经网络去优化。
5. 为了优化性能，网络应选择非常适合定位形状的比例。

Effectiveness of PointSIFT Module

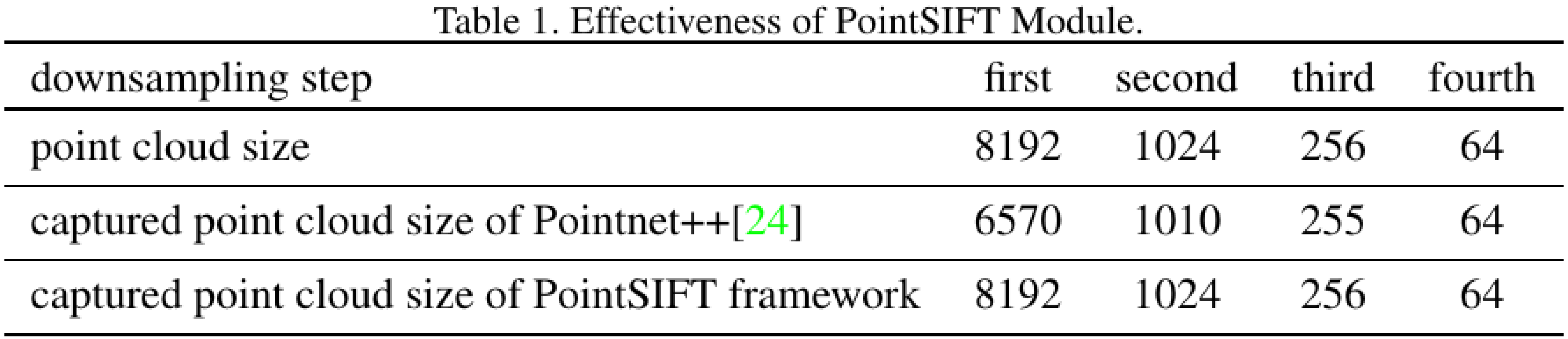
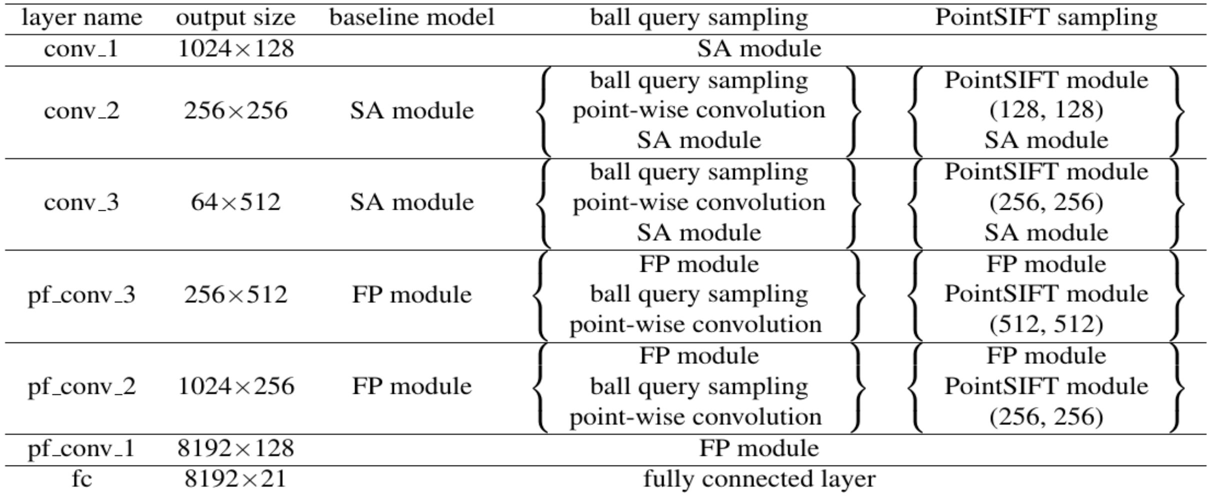


Table 1. Effectiveness of PointSIFT Module.

downsampling step	first	second	third	fourth
point cloud size	8192	1024	256	64
captured point cloud size of Pointnet++[24]	6570	1010	255	64
captured point cloud size of PointSIFT framework	8192	1024	256	64

Effectiveness of OE Unit

- S8N Vs ball query(PointNet++)



layer name	output size	baseline model	ball query sampling	PointSIFT sampling
conv_1	1024×128	SA module		
conv_2	256×256	SA module	ball query sampling point-wise convolution SA module	PointSIFT module (128, 128) SA module
conv_3	64×512	SA module	ball query sampling point-wise convolution SA module	PointSIFT module (256, 256) SA module
pf_conv_3	256×512	FP module	FP module ball query sampling point-wise convolution	FP module PointSIFT module (512, 512)
pf_conv_2	1024×256	FP module	FP module ball query sampling point-wise convolution	FP module PointSIFT module (256, 256)
pf_conv_1	8192×128	FP module		
fc	8192×21	fully connected layer		

Table 3. Architectures for comparison of different sampling methods. After ball query sampling, point-wise convolution takes 32 × 1 kernels for extracting features.

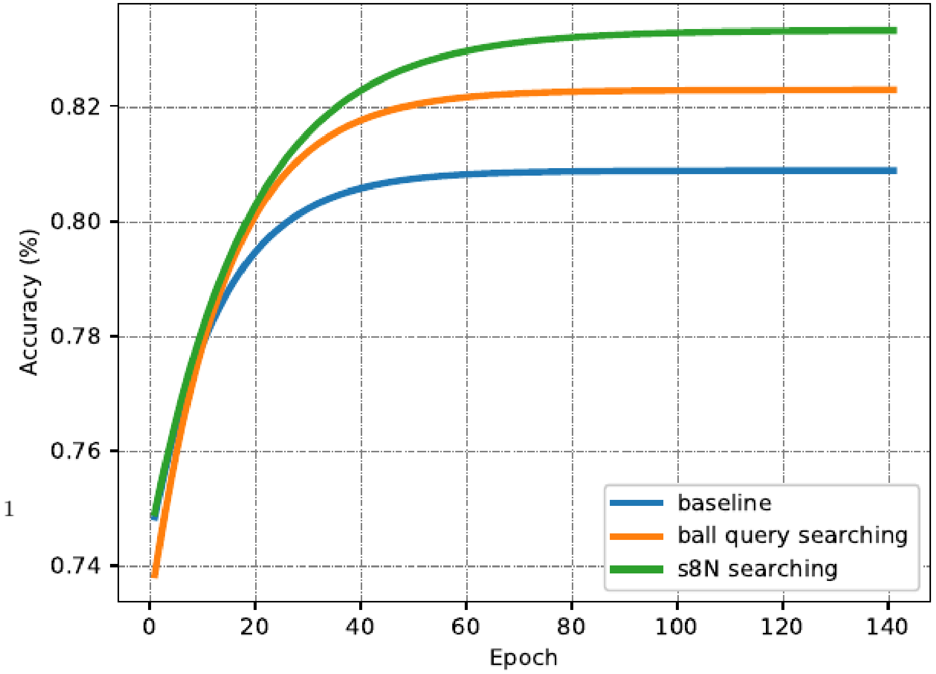


Figure 6. Accuracy for different searching methods. We use SavitzkyGolay filter for smoothing all the lines.

Effectiveness of Scale Awareness

- 生成 10000 个简单的不同尺度的 shape，在生成的 shape 数据上训练网络
- 测试不同层中 PointSIFT modules 的激活程度对于特定 shape 是否对应于 shape 的 scale。
- 在层次结构中具有最高激活程度的 PointSIFT 模块的位置与输入形状相对于 max 和 min scale 的比例对应。

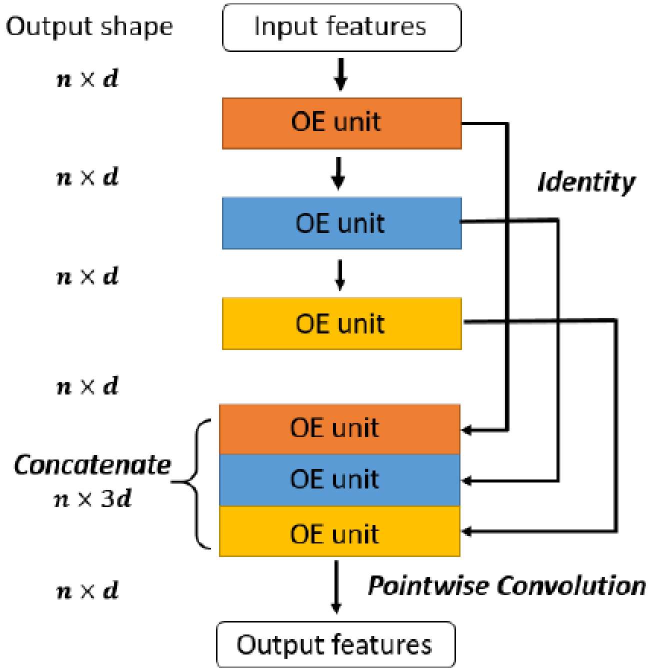


Figure 4. PointSIFT Module. Input features first pass through a series of Orientation-encoding(OE) layers, then outputs of OE units are concatenated and transformed by another point-wise convolution to obtain multi-scale feature.

Result

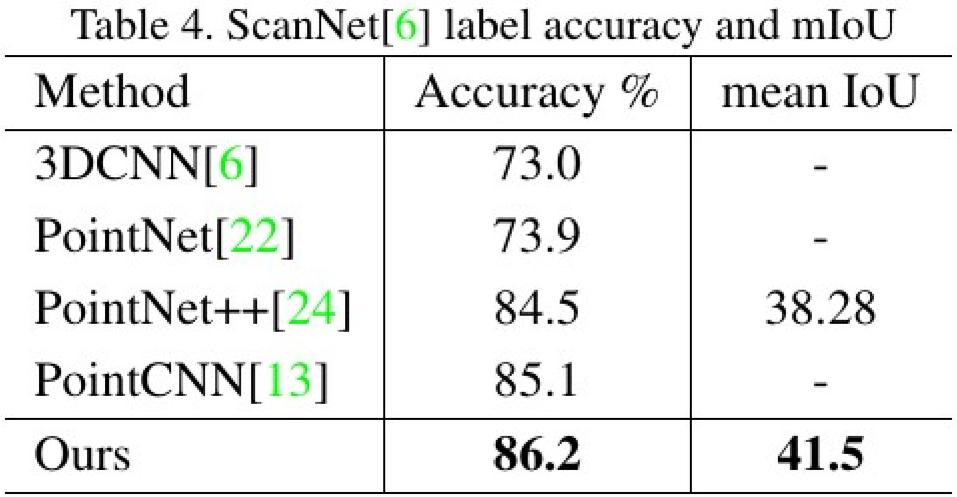


Table 4. ScanNet[6] label accuracy and mIoU

Method	Accuracy %	mean IoU
3DCNN[6]	73.0	-
PointNet[22]	73.9	-
PointNet++[24]	84.5	38.28
PointCNN[13]	85.1	-
Ours	**86.2**	**41.5**

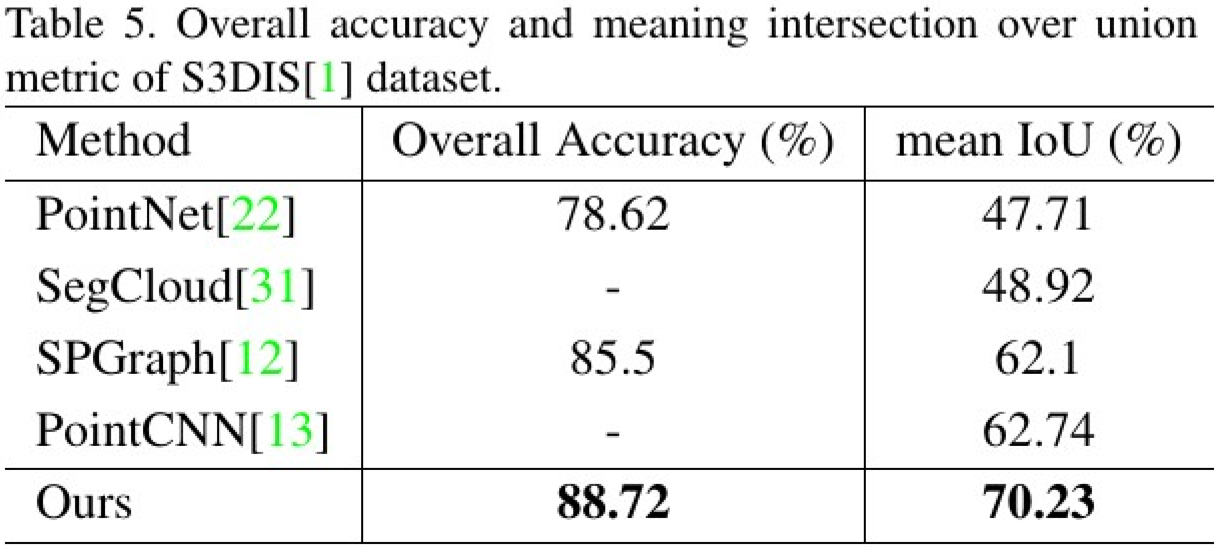
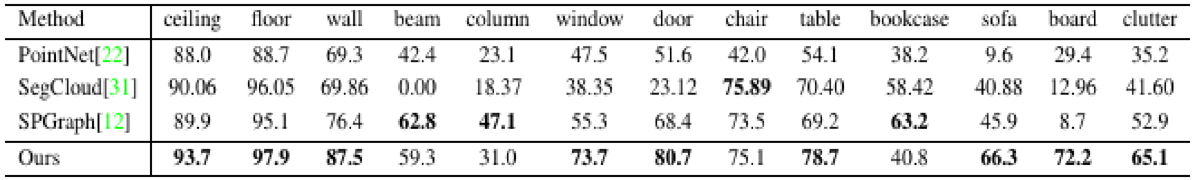


Table 5. Overall accuracy and meaning intersection over union metric of S3DIS[1] dataset.

Method	Overall Accuracy (%)	mean IoU (%)
PointNet[22]	78.62	47.71
SegCloud[31]	-	48.92
SPGraph[12]	85.5	62.1
PointCNN[13]	-	62.74
Ours	**88.72**	**70.23**

Result

Table 2. IoU for all categories of S3DIS[1] dataset.



Method	ceiling	floor	wall	beam	column	window	door	chair	table	bookcase	sofa	board	clutter
PointNet[22]	88.0	88.7	69.3	42.4	23.1	47.5	51.6	42.0	54.1	38.2	9.6	29.4	35.2
SegCloud[31]	90.06	96.05	69.86	0.00	18.37	38.35	23.12	**75.89**	70.40	58.42	40.88	12.96	41.60
SPGraph[12]	89.9	95.1	76.4	**62.8**	**47.1**	55.3	68.4	73.5	69.2	**63.2**	45.9	8.7	52.9
Ours	**93.7**	**97.9**	**87.5**	59.3	31.0	**73.7**	**80.7**	75.1	**78.7**	40.8	**66.3**	**72.2**	**65.1**

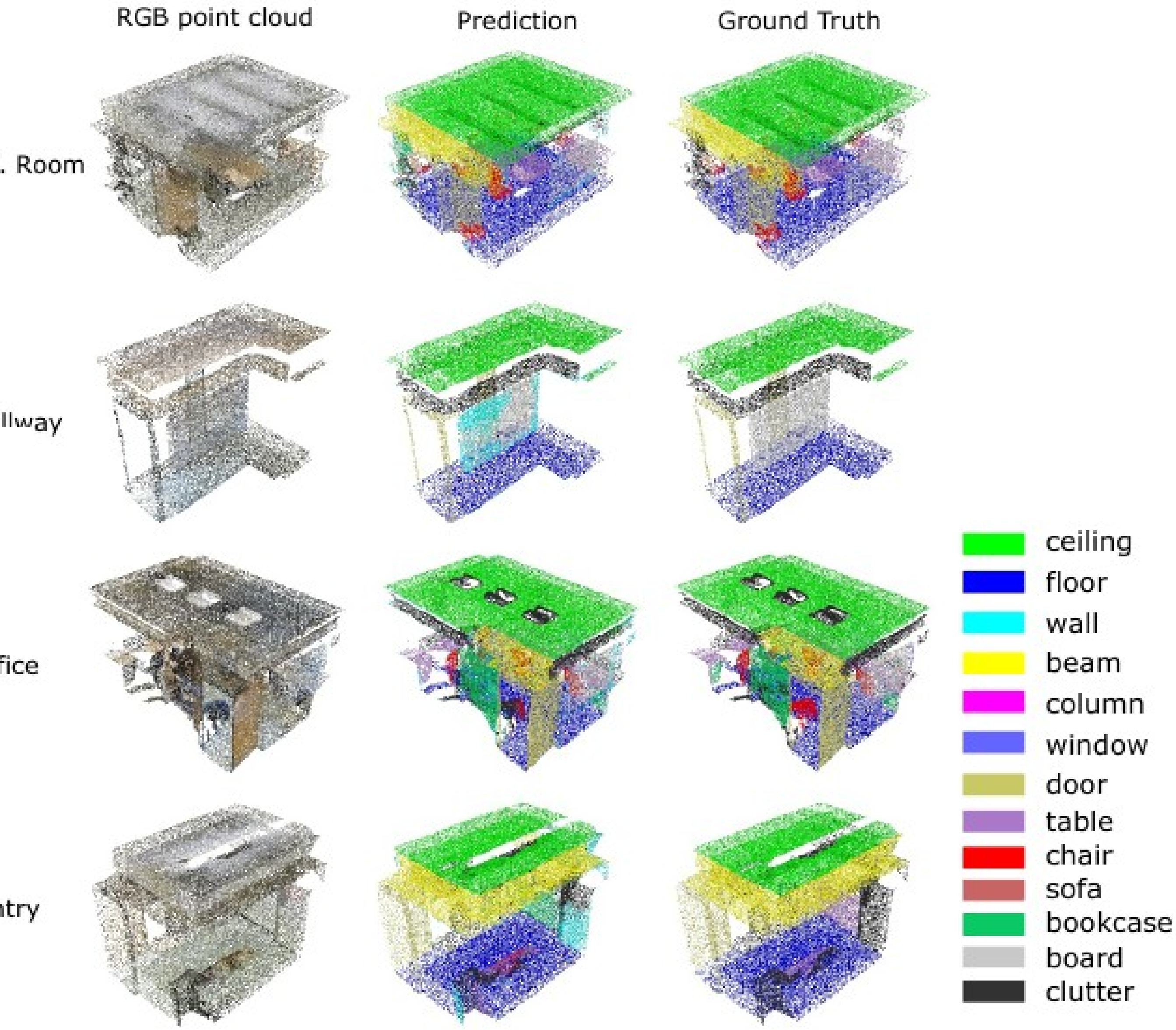


Figure 5. Visualization of results on S3DIS dataset[1]

Conclusion

1. An effective end-to-end architecture for point cloud semantic segmentation is proposed based on PointSIFT modules.

2. PointSIFT has a significant role in point cloud feature extraction

3. Orientation encoding unit capture the information of different orientations

4. multi-scale representation of PointSIFT modules enables the processing of objects with various scale.

Reference

- http://aishack.in/tutorials/sift-scale-invariant-feature-transform-introduction/