深蓝学院
shenlanxueyuan.com

三维点云作业第七讲

主讲人 王永浩

ISS算法实现

- Intrinsic Shape Signatures (ISS): A Shape Descriptor for 3D Object Recognition
    - Keypoints are those have large 3D point variations in their neighborhood
    - Simple, Principle Component Analysis (PCA)
    - The smallest eigenvalue of the covariance matrix should be large.

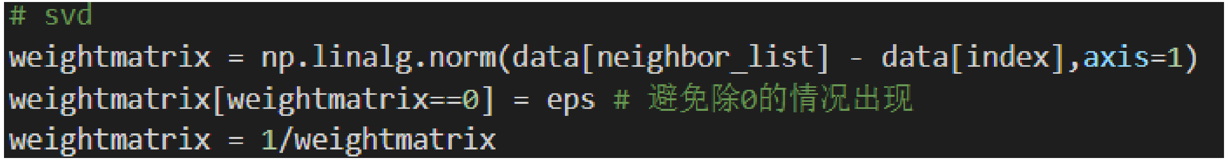
ISS算法的整体实现流程并不算复杂，需要大家细心理解算法步骤。

ISS算法实现

- 计算给定点的权重

$$w_j = \frac{1}{|\{p_k : \|p_k - p_j\|_2 < r\}|}$$

与距离成反比



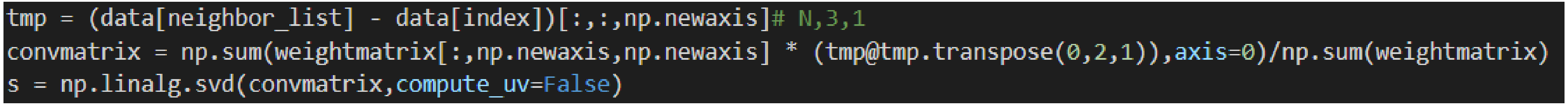
```
# svd
weightmatrix = np.linalg.norm(data[neighbor_list] - data[index],axis=1)
weightmatrix[weightmatrix==0] = eps # 避免除0的情况出现
weightmatrix = 1/weightmatrix
```

ISS算法实现

- 计算加权协方差矩阵并对其Eigenvalue排序

Weighted covariance matrix:

$$Cov(p_i) = \frac{\sum_{\|p_j-p_i\|_2<r} w_j(p_j-p_i)(p_j-p_i)^T}{\sum_{\|p_j-p_i\|_2<r} w_j}$$



```
tmp = (data[neighbor_list] - data[index])[:,:,np.newaxis]# N,3,1
convmatrix = np.sum(weightmatrix[:,np.newaxis,np.newaxis] * (tmp@tmp.transpose(0,2,1)),axis=0)/np.sum(weightmatrix)
s = np.linalg.svd(convmatrix,compute_uv=False)
```

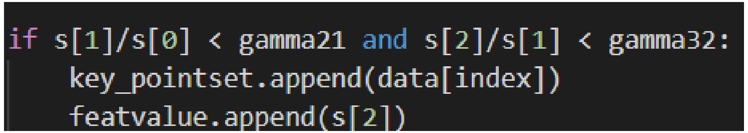
ISS算法实现

- 根据特征值筛选关键点

p_i is a keypoint if

$$\frac{\lambda_i^2}{\lambda_i^1} < \gamma_{21} \ \ and \ \ \frac{\lambda_i^3}{\lambda_i^2} < \gamma_{32}$$

- A flat surface can be $\lambda_i^1 = \lambda_i^2 > \lambda_i^3$
- A line can be $\lambda_i^1 > \lambda_i^2 = \lambda_i^3$
- So we have to ensure $\lambda_i^1 > \lambda_i^2 > \lambda_i^3$



```
if s[1]/s[0] < gamma21 and s[2]/s[1] < gamma32:
    key_pointset.append(data[index])
    featvalue.append(s[2])
```

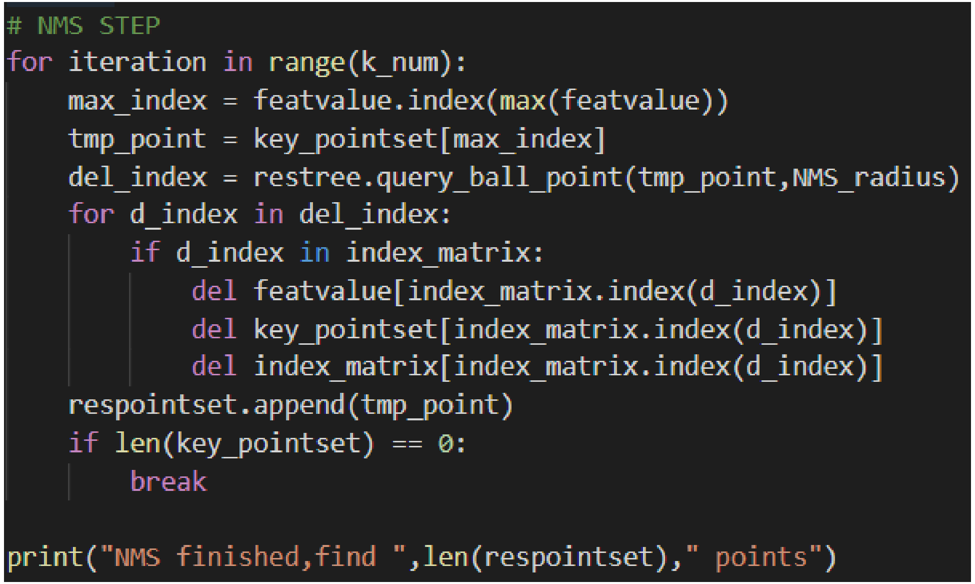
ISS算法实现

- 对λ_i^3做NMS，其过程如下：

1) 记构建用于存放最终结果点集为 A，记初始存放所有关键点的点集为 B。
2) 取 B 中所有点的λ^3(前面 ISS 步骤中的第三个特征值)最大的点p放入 A，并且求p的 R 近邻（参数 NMS_radius）,在 B 中除去点 p 和其 R 近邻。
3) 重复第 2 步，直到 B 空或达到预先设定的关键点数目

ISS算法实现

- 对λ_i^3做NMS，其过程如下：

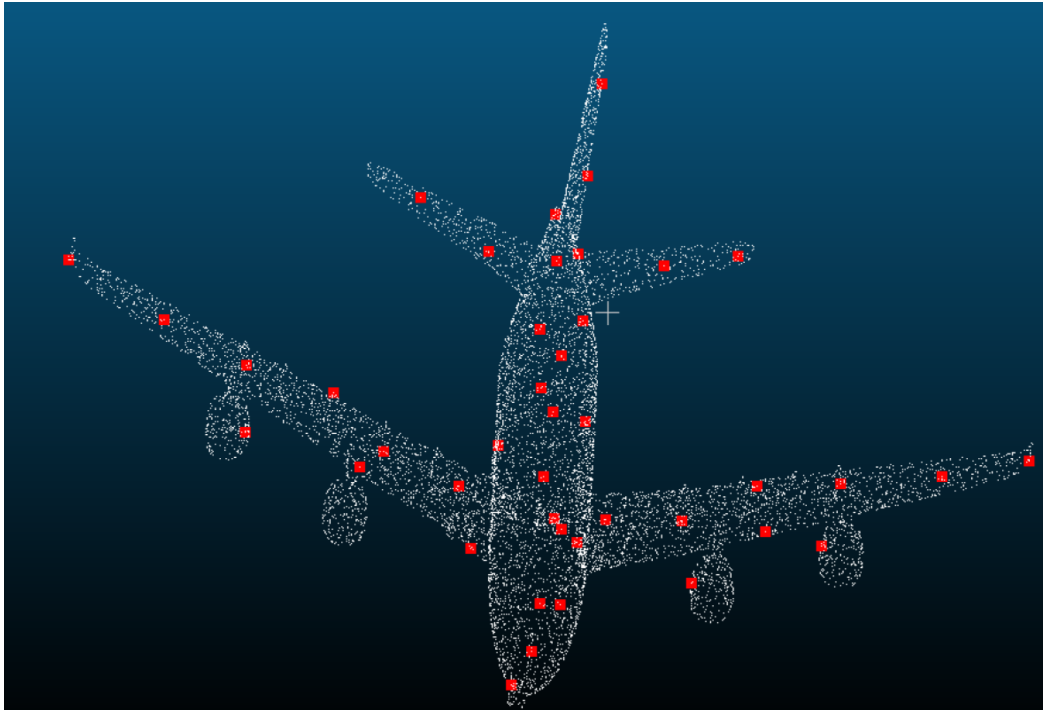


```
# NMS STEP
for iteration in range(k_num):
    max_index = featvalue.index(max(featvalue))
    tmp_point = key_pointset[max_index]
    del_index = restree.query_ball_point(tmp_point,NMS_radius)
    for d_index in del_index:
        if d_index in index_matrix:
            del featvalue[index_matrix.index(d_index)]
            del key_pointset[index_matrix.index(d_index)]
            del index_matrix[index_matrix.index(d_index)]
    respointset.append(tmp_point)
    if len(key_pointset) == 0:
        break

print("NMS finished,find ",len(respointset)," points")
```

ISS算法实现

- 实现效果：



在线问答

Q&A

感谢各位聆听！

Thanks for Listening