# 能量函数



where L is a given set of labels (planes) and d(.) is an indicator function. Let P represent a set of data points, the multiple labeling task is to assign a point p P, a label lp  L such that the labeling L minimizes the energy E(L), where lp is the set of labels appearing in L and N is an assumed neighborhood for data points. Three energy terms are considered in the energy formula.

The optimize target is to find a set of labels for each points.

D: measures the discrepancy between data points and labels. It is the sum of the distances of points to their assigned labels. the key issue is to find a function to describe points with the same label, and define a proper distance function.

S: The smooth cost term measures the label inconsistency between neighboring points. It is the sum of

weight wpq of each pair of neighboring points p and q that are

assigned to different labels.

C: The label cost term measures the number of labels appearing in L.

# workflow

## algorithm 1

用binary Graph cut

用于downtown，高楼

(1) isolate removing

(2) voxelization and set up neighbourhood

(3) feature extraction for each voxel

(3.1) unique features

(3.1.1) eigenvalue / eigenvector

计算协方差矩阵



Eigen::Vector4f xyz\_centroid\_;

EIGEN\_ALIGN16 Eigen::Matrix3f covariance\_matrix\_;

computeMeanAndCovarianceMatrix (\*org\_pts, indices, covariance\_matrix\_, xyz\_centroid\_);

Eigen::EigenSolver<Eigen::Matrix3f> es(covariance\_matrix\_);

Eigen::Matrix3f D = es.pseudoEigenvalueMatrix();

Eigen::Matrix3f V = es.pseudoEigenvectors();

调整法线方向

按单位向量n0(0,0,1)进行调整



(3.2) neighbour-relevant feature

(4) energy function

分层聚类，聚类中心作为种子点

初始聚类过滤，去除树木和杆状物

## algorithm 2

voxelization

从高到低，选取一个voxel(非墙面)