# Kmeans聚类对深度图进行图像分割测试报告

## Changelog

|  |  |  |  |
| --- | --- | --- | --- |
| 版本号 | 变更人 | 变更说明 | 变更时间 |
| V1.0 | 张琛 | 初稿 | 2015/05/12 |
| V1.1 | 张琛 | +**补充测试**：edge\_up 减少杂边，及其对应 edge\_whole, dtrans, kmeans(k=2) & 二值化效果/时间性能对比 | 2015/05/13 |

## 目标

本文使用kmeans聚类算法，对kinect获取的深度图进行图像分割，以试图提取出人等前景物体。

## 测试步骤

### 实验环境：

CPU: Intel i5 3470

OS: Windows7

IDE: Matlab R2013b

### 具体步骤

本次实验测试分为三类：

1. 直接对原始深度图（变量用***dmat***表示）进行聚类，可细分为：

a) 特征向量为三维 {x, y, z}，x, y 分别为像素坐标，z为像素对应深度值。因为x,y 单位为像素， z为毫米，可能x,y维对聚类影响微小；

b) 特征向量为一维 {z}；

2. 以距离变换之后的dist-map值作一维特征向量，进行聚类；

其计算过程简要表示：

a) 对原始深度图做canny，得大部分（脚部除外）身体轮廓***edge\_up***；

b) 用***kernel***=(1,1,1,-1,-1,-1)’进行卷积，得到 filter-map, 再canny，得到与地板分离后的脚部轮廓 ***edge\_ft***，此步骤包括手动选取threshold、图像闭操作等技巧性调整；

c) 人体轮廓 ***edge\_whole = edge\_up + edge\_ft***；

d) 对轮廓图做 distance-transform，变量用***dtrans***表示。

3. 使用2.b得到分离地板的mask，分别对***dmat***, ***dtrans***处理，去掉地面（填充零）；再对两者进行kmeans聚类。

## 测试结果

kmeans算法主要有两个参数：特征向量集n\*p矩阵，类别数目k。下面的测试中n为图像像素点个数，n=width\*height。

表 1 测试数据集

|  |  |  |  |
| --- | --- | --- | --- |
| 数据序号 | 文件名 | 分辨率 | 备注 |
| 1 | sun\_han\_short.oni | 640\*480 | 室内，两人，狭长杂乱，有地板， |
| ~~2~~ | ~~xxxxxxxxxxxxxx~~ | ~~320\*240~~ | ~~xxxxxxxxxxxxxxxxxx~~ |

### 数据1测试结果

原始的RGB以及深度数据如图1所示。



(a) (b)

图 1 原始数据RGB图(a)以及深度图(b)

1. 对***dmat***聚类

考虑深度图中存在无效区域，两个人，背景，设定k=5，聚类结果如图2所示。可以看出，因为x,y值域较小，对聚类产生的结果几乎为零；人体与周围深度相近的环境难以分离。

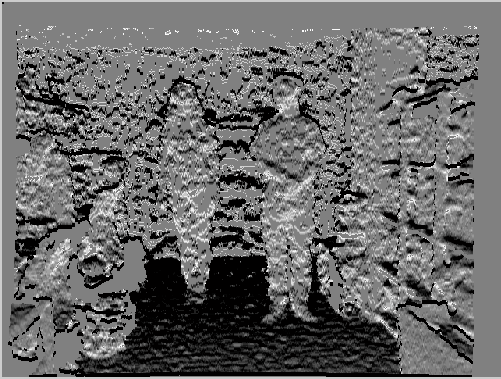
 

(a) (b)

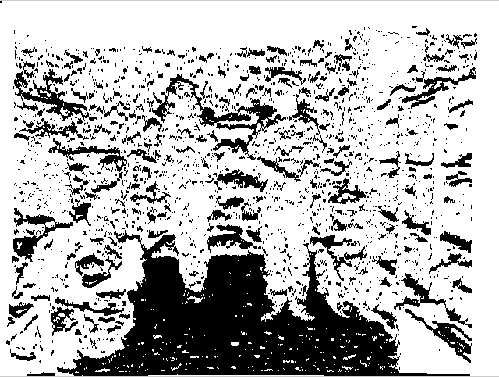
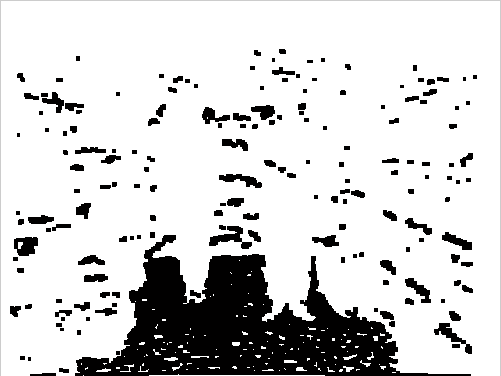
图 2 label图，不同灰度表示不同类别label，灰度是随机的。 (a)为特征向量为{x,y,z}结果，耗时1.05s；(b)为特征向量为{z}结果，耗时0.366s

2. 对***dtrans***聚类

图3为***dtrans***获得的中间步骤，以及在***dtrans***上的kmeans聚类结果

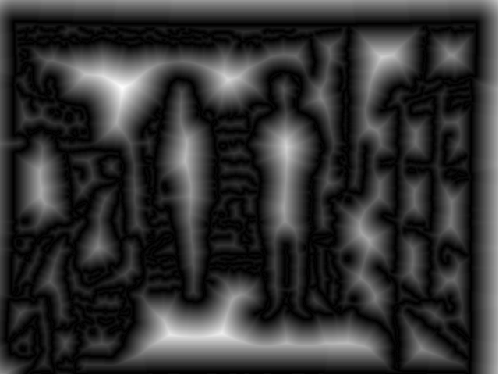
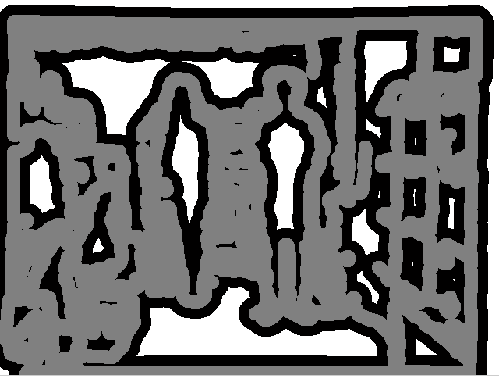
(a) edge\_up (b)filter-map

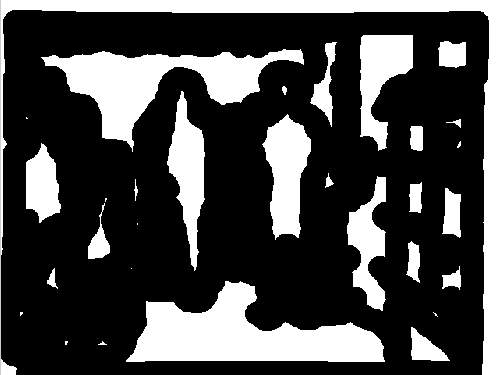
(c) otsu’s thresholding 二值化 (d) 图像闭操作

(e) edge\_ft (f) edge\_whole

(g) ***dtrans*** (h) k=3聚类



(i) k=2聚类

图 3 (a)~(f)为由原始深度图到人体整体轮廓过程。(g)为***dtrans***结果，(h)为kmeans，k=3时label图，(i) 为kmeans，k=2时label图，相当于thresholding二值化

3. 使用图3(d)二值图做mask，对***dmat***处理，去掉地面（填充零），再进行kmeans聚类

由上图3看出，对***dtrans***进行kmeans聚类，k>2时，无明确物理意义；k=2时，相当于直接二值化，因此这里不再对dtrans进行聚类测试。

图4为mask对***dmat***进行掩模处理，并对结果图进行kmeans聚类（k=5）的结果。

(a) mask对***dmat***处理后 (b) kmeans对(a)的聚类label结果

图 4 mask对***dmat***进行掩模处理，并对结果图进行kmeans聚类（k=5）

经过mask处理后，kmeans可以较好的将人体与地面分离；但是人体与场景中相同深度的其他物体仍被分为同一类。

## 结论分析

1. 对深度图聚类比对distance map聚类物理意义更加明确，分类效果更明显；

2. 使用一维深度值作为kmeans聚类的特征空间是足够的；

3. 使用图3(d)做mask对深度图进行预处理的操作，实际上等价于对kmeans特征空间增加一维二值特征。但是由于其值域与深度值值域差距太大，这里采用分步（mask预处理）的方式，以避免增加维度需要考虑的normalization问题。

**存在问题：**

1. 聚类的结果label是随机的，不具备标记哪些像素区域属于人体的能力，仅具有图像分割能力；如果要在图4(b)上选取人体区域，仍然需要比较准确的种子点。

2. kmeans的结果为局部最优解，结果很大程度依赖中心点初值选取，不同次的运行得到的分割结果并不相同，图5为一次不好的分割结果，人体手部交叉于身前，与其他部位被标记为不同类别。



图 5 kmeans不好的分割结果

3. 由于kmeans需要手动设定k值（e.g., k=5），无法支持动态的人体数量的检测；同时，由于初值选取的问题，有些时候kmeans算法会因为迭代过程中某一类点数为零，导致程序出错[[1]](#footnote-1)，如图6。

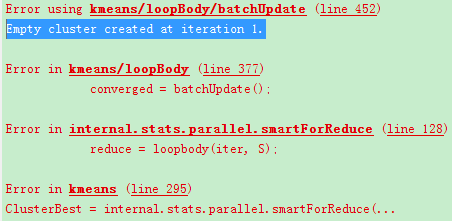


图 6 kmeans迭代出错时Matlab Traceback

4. 时间性能问题。表2记录了上述各步骤运行时间，红色为比较耗时的步骤，因为本次实验在matlab中实现，未在c++环境中运行，可能实际移植到c++环境中性能会有较大提升。

表 2 Matlab中各步骤运行时间

|  |  |  |
| --- | --- | --- |
| 步骤 | 640\*480耗时(s) | 320\*240耗时(s) |
| edge\_up | 0.082 | 0.035 |
| filter-map | 0.001 | 0.001 |
| thresholding | 0.002 | 0.001 |
| imclose | 0.015 | 0.013 |
| edge\_ft | 0.071 | 0.025 |
| edge\_whole | 0.001 | 0.000 |
| dtrans | 0.011 | 0.003 |
| masking | 0.001 | 0.000 |
| kmeans-1D | 0.325 | 0.151 |
| **TOTAL** | 0.509 | 0.229 |

# 补充测试

Matlab中canny算子接口如下：

|  |
| --- |
| BW = edge(I,'canny',thresh,sigma) |

不指定thresh、sigma（高斯标准差）的默认情况下，thresh（即th\_high）由算法自动求取(th\_low=0.4\*th\_high)，sigma=sqrt(2)。其中，thresh越大，得到的边缘越少；sigma越大，边缘棱角越不明显。

上文中对原始深度图做canny得到图3(a)所示边缘时，采用默认canny参数，其值：

*[th\_low, th\_high]= [0.0125, 0.0313]*

*sigma=sqrt(2)*

结果如图7所示：



图 7 图3(a)拷贝

其优点为，人体四周书架等物体轮廓（不算噪声？）标记较为准确；缺点为，背景噪声比较明显。

若要去掉周围复杂物体轮廓，手动调整[th\_low, th\_high]、sigma值：

a) 保持thresh=0.03125不变，增大sigma，得到的轮廓线条更少，棱角更圆滑，如图8



(a) (b)

图 8 canny, thresh=0.03125. (a)sigma=3; (b)sigma=5

b) 取sigma=3保持不变，增大thresh（th\_high），轮廓边缘明显减少，如人右侧、背后书架轮廓基本消失。

(a) (b)

图 9 canny, sigma=3. (a)thresh=0.05; (b)thresh=0.1

c) 固定 thresh=0.1; sigma=3，即对图9(b)，继续求解edge\_whole, ***dtrans***：

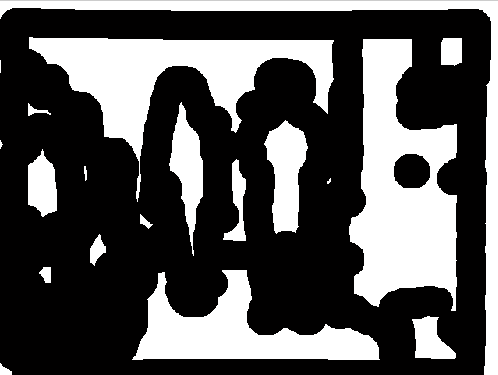
(a) (b)

(c) (d)

图 10 (a), (b)为设定thresh=0.1; sigma=3时的edge\_whole, ***dtrans***，注意到右侧书架轮廓消失； (c), (d)为默认参数时的对应图，即图3(f), 3(g)

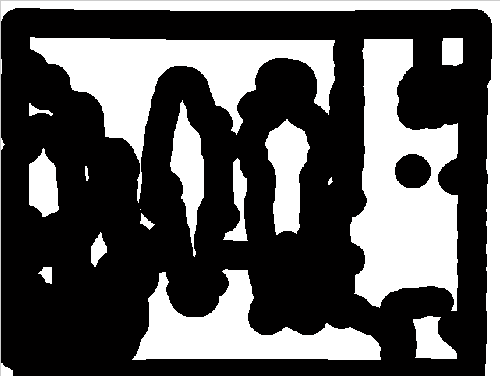
d) 对图10(b)得到的***dtrans***进行kmeans聚类，设定k=2，结果如下

(a) (b)

图 11 kmeans取k=2两次运行的不同结果，说明label是随机的，并不保证按数据值升序或降序进行label

e) 直接对图10(b)得到的***dtrans***二值化，可以确保原图中亮色区域二值化为白色，暗色区域保持为黑色；同时，其效率远高于kmeans聚类，如表3所示

(a) (b)

图 12 对图10(b)的***dtrans***二值化。(a)为采用otsu’s方法，自动求解阈值，此处th=0.498；(b)为手动设定阈值th=0.23，效果与图11(a)几乎完全相同

表 3 kmeans(k=2) vs. thresholding时间性能（秒）

|  |  |
| --- | --- |
| 方法 | 分辨率640\*480 |
| kmeans(k=2) | 0.190s |
| thresholding(手动阈值) | 0.013s |

1. <http://stackoverflow.com/questions/18009664/kmeans-matlab-empty-cluster-created-at-iteration-1-error> [↑](#footnote-ref-1)