一、 Mean Shift 算法简介

Mean Shift 算法，又称为均值漂移算法，Mean Shift 的概念最早是由 Fukunage 在 1975 年提出的，在后来由 Yizong Cheng 对其进行扩充，主要提出了两点的改进：

- 定义了核函数；
- 增加了权重系数。

核函数的定义使得偏移值对偏移向量的贡献随之样本与被偏移点的距离的不同而不同。权重系数使得不同样本的权重不同。Mean Shift 算法在聚类，图像平滑、分割以及视频跟踪等方面有广泛的应用。

- 图像平滑：图像最大质量下的像素压缩；
- 图像分割：类似图像平滑的应用，但可以使平滑的图像进行分离以达到前后景分割的目的；
- 目标跟踪：例如针对监控视频中某个人物的动态跟踪；
- 常规聚类：如用户聚类等。

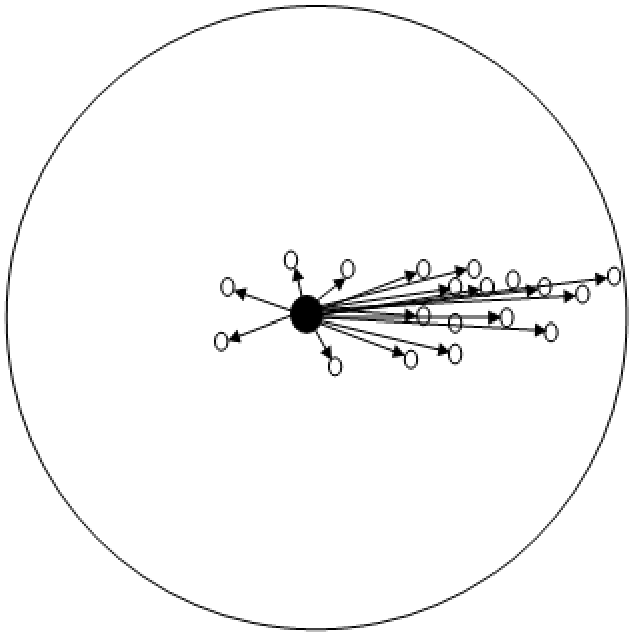
二、 Mean Shift 算法原理

- **Mean Shift 向量**

对于给定的d维空间R^d中的 n 个样本点$x_i, i = 1,2,\dots,n$ 则对于 x 点，其 Mean Shift 向量的基本形式为：

$$M_h(x) = \frac{1}{k}\sum_{x_i \in S_h}(x_i - x)$$

其中，S_h指的是一个半径为 h 的高维球区域，如图所示。

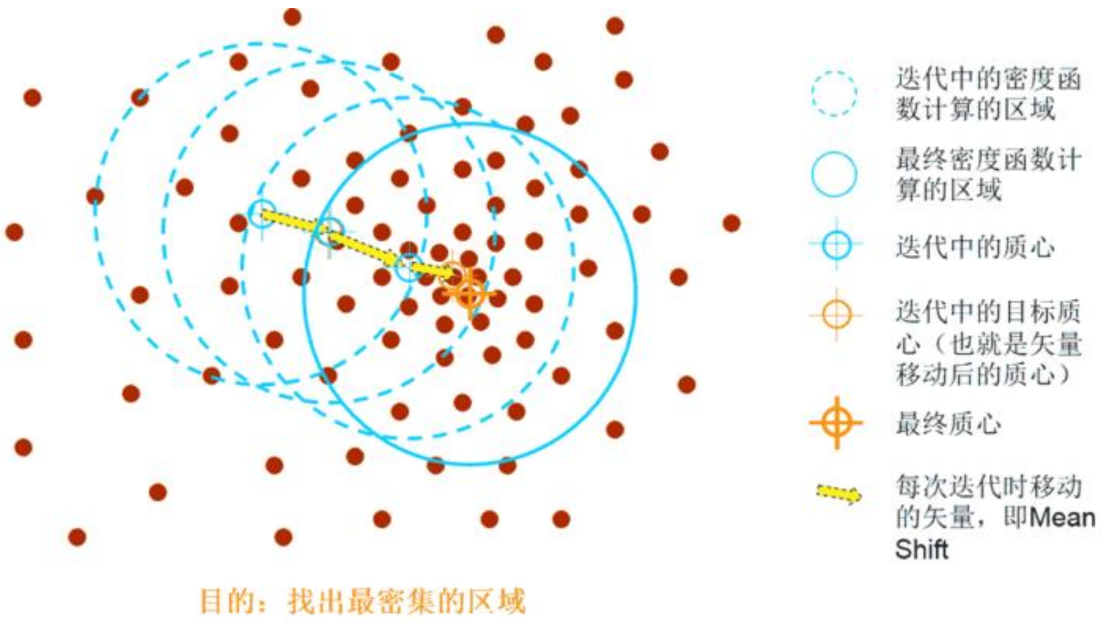


S_h的定义为

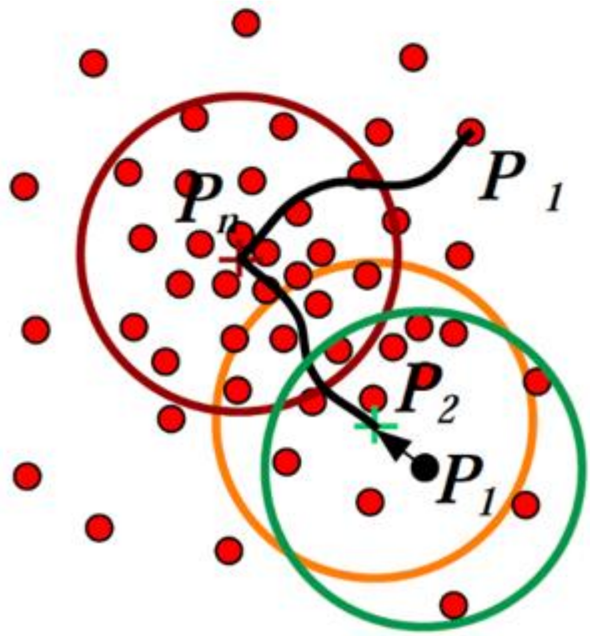
$$S_h(x) = (y|(y-x)(y-x)^T \le h^2)$$

里面所有点与圆心为起点形成的向量相加的结果就是 Mean shift 向量。下图黄色箭头就是 M_h（Mean Shift 向量）。

对于 Mean Shift 算法，是一个迭代的步骤，即先算出当前点的偏移均值，将该点移动到此偏移均值，然后以此为新的起始点，继续移动，直到满足最终的条件。



Mean-Shift 聚类就是对于集合中的每一个元素，对它执行下面的操作：把该元素移动到它邻域中所有元素的特征值的均值的位置，不断重复直到收敛。准确的说，不是真正移动元素，而是把该元素与它的收敛位置的元素标记为同一类。



如上的均值漂移向量的求解方法存在一个问题，即在 的区域内，每一个样本点 x 对样本 X 的共享是一样的。而实际中，每一个样本点 x 对样本 X 的贡献是不一样的，这样的共享可以通过核函数进行度量。

- **核函数**

在 Mean Shift 算法中引入核函数的目的是使得随着样本与被偏移点的距离不同，其偏移量对均值偏移向量的贡献也不同。核函数是机器学习中常用的一种方式。核函数的定义如下所示：

X 表示一个 d 维的欧式空间，x 是该空间中的一个点，$x = \{x_1, x_2, \dots, x_d\}$。其中，x 的模$\|x\|^2 = xx^T$，R 表示实数域，如果一个函数 K:X→R 存在一个剖面函数 k:[0,∞]→R，即$K(x) = k(\|x\|^2)$

并且满足：

- k 是非负的
- k 是非增的
- k 是分段连续的

那么，函数 K(x)就称为核函数。

引入核函数后的 Mean Shift 向量

假设在半径为 h 的范围S_h范围内，为了使得每一个样本点 x 对于样本 X 的共享不一样，向基本的 Mean Shift 向量形式中增加核函数，得到如下改进的 Mean Shift 向量形式：

$$M_h(x) = \frac{\sum_{i=1}^{n} G\left(\frac{x_i - x}{h_i}\right)(x_i - x)}{\sum_{i=1}^{n} G(\frac{x_i - x}{h_i})}$$

其中，$G(\frac{x_i-x}{h_i})$为核函数，通常可以取S_h为整个数据集范围。

计算M_h时需要考虑距离的影响，同时也可以认为在所有的样本点 X 中,重要性并不一样,因此对每个样本还引入一个权重系数。如此以来就可以把 Mean Shift 形式扩展为：

$$M_h(x)=\frac{\sum_{i=1}^{n} G\left(\frac{x_i-x}{h_i}\right)w(x_i)(x_i-x)}{\sum_{i=1}^{n} G\left(\frac{x_i-x}{h_i}\right)w(x_i)}$$

其中，$w(x_i)$是一个赋给采样点的权重。

三、 Mean Shift 算法流程

1. 在未被标记的数据点中随机选择一个点作为起始中心点 center；
2. 找出以 center 为中心半径为 radius 的区域中出现的所有数据点，认为这些点同属于一个聚类 C。同时在该聚类中记录数据点出现的次数加 1。
3. 以 center 为中心点，计算从 center 开始到集合 M 中每个元素的向量，将这些向量相加，得到向量 shift。
4. center = center + shift。即 center 沿着 shift 的方向移动，移动距离是||shift||。
5. 重复步骤 2、3、4，直到 shift 的很小（就是迭代到收敛），记住此时的 center。注意，这个迭代过程中遇到的点都应该归类到簇 C。

6. 如果收敛时当前簇 C 的 center 与其它已经存在的簇 C2 中心的距离小于阈值，那么把 C2 和 C 合并，数据点出现次数也对应合并。否则，把 C 作为新的聚类。
7. 重复 1、2、3、4、5 直到所有的点都被标记为已访问。
8. 分类：根据每个类，对每个点的访问频率，取访问频率最大的那个类，作为当前点集的所属类。

四、 Mean Shift 算法优缺点总结

优点

- 算法计算量不大，在目标区域已知的情况下完全可以做到实时跟踪；
- 采用核函数直方图模型，对边缘遮挡、目标旋转、变形和背景运动不敏感。

缺点

- 缺乏必要的模板更新；
- 跟踪过程中由于窗口宽度大小保持不变，当目标尺度有所变化时，跟踪就会失败；
- 当目标速度较快时，跟踪效果不好；
- 直方图特征在目标颜色特征描述方面略显匮乏，缺少空间信息；