一种基于均值漂移的视频场景检测方法

张玉珍 王建宇 戴跃伟 魏带娣
(南京理工大学自动化学院, 南京 210094)

摘 要 提出了一种高效的视频场景检测方法。首先基于均值漂移, 在滑动镜头窗内对各镜头聚类, 并获得相应的聚类中心, 然后根据电影视频场景的发展模式, 计算两个镜头类之间的时序距离, 接着基于时空关系进行场景检测, 并且由相应的聚类中心获得场景关键帧, 最后对场景过分割进行后续处理。实验证实该方法能快速聚类, 并且有效地检测出场景和场景关键帧。

关键词 均值漂移 聚类 镜头类 场景检测 场景关键帧
中图法分类号: TP391 TN941.1 **文献标志码**: A **文章编号**: 1006-8961(2010)02-0314-07

A Video Scene Detection Method Based on Mean Shift

ZHANG Yu-zhen, WANG Jian-yu, DAI Yue-wei, WEI Dai-di
(School of Automation, Nanjing University of Science & Technology, Nanjing 210094)

Abstract An efficient algorithm for scene detection is proposed. Where firstly within a sliding shot window, the shots are clustered and each cluster center is achieved rapidly by employing MS clustering; then according to the development pattern of film video scene, the temporal distances between two shots are computed and scenes are detected based on the temporal and spatial relationship. In addition, the scene key frames can be achieved on the basis of corresponding cluster centers. Finally, a succeeding procession for over-segmented scenes is introduced. Experiments prove this algorithm can cluster shots rapidly and detect scenes and the key frames of the scence efficiently.

Keywords mean shift; clustering; shot cluster; scene detection; scene key frame

0 引 言

为了从海量的视频数据中快速获得感兴趣的片段, 基于语义的视频检索已成为现代研究的热点之一。视频数据是非结构化数据, 对视频数据的结构化是实现视频检索的前提。视频结构化一般先将视频流分割成粒度较小的镜头, 每个镜头由若干关键帧代替。然后将相似的镜头组合成镜头组(类), 接着基于镜头类, 将语义相关、时间相近的镜头类组成有一定语义的场景, 表达一个完整的故事单元。

现有场景分割方法大体分为两类: 基于先验模型的方法和基于视频制作原理的方法[1]。基于模型的方法需要根据特定应用或领域建立先验模型。这种方法的主要缺陷是在每一类应用之前都要建立一个领域模型, 需要相关领域知识。以场景转移图为代表的一系列算法属于基于视频制作原理的方法。这类方法在视频流结构化成镜头后, 用时间约束的聚类法把视觉相似和时间相邻的镜头聚类成镜头类, 然后在镜头类的基础上构建场景。这种方法是基于视频制作原理的, 且不需要精确的领域知识, 因此它在电影视频的场景分割中应用非常广泛。目

基金项目: 南京理工大学科技发展基金(XKF09023)
收稿日期: 2008-09-04; **改回日期**: 2008-11-29
第一作者简介: 张玉珍(1973—), 女, 讲师。南京理工大学博士研究生。研究方向为基于语义的视频检索、模式识别。E-mail: olindazh@163.com

前多数场景分割都是比较镜头相似度把相关镜头聚类[2]。一个长的视频文档可能包含几千个镜头, 如果对所有的镜头进行简单的层次聚类, 单是计算相互间的相似度就需要很大的计算量和时间, 方法的效率很低。考虑到组成场景的各镜头在时间上相近, 文献[3]提出基于时间受限的镜头聚类方法, 它只考虑位于一个固定时间窗口 T 内的镜头的相似性, 而位于 T 外的镜头的相似性则不予考虑 (即相似度为零), 因此聚类结果不够完全。为了克服时间受限镜头聚类算法的不足, Rui等人提出了时间自适应分组法[4], 即镜头的相似度随着它们之间时间距离的改变而变化, 距离越大, 相似度越小。但是根据电影编辑的原理, 为了突出效果, 很多镜头的长短不一, 因此, 单纯地使用时间阈值来判别相关性不尽合理, 使用镜头数目则更好一些。在文献[5-6]中, 都提出了基于滑动镜头窗的比较镜头相似度的镜头聚类, 其中镜头窗是以镜头为单位。

上面场景分割中的聚类都是基于相似度阈值聚类, 即求出视频中各镜头间的相似度, 当相似度超过给定阈值时, 合并为一类。采用这种聚类方法, 要事先确定阈值, 对于不同视频阈值是不一样的。而且若阈值取得较小, 聚类时可能有些类别没有检测出来, 阈值取得较大则易产生较多类别, 导致过度聚类。而且这样的聚类只能得到聚类成员, 没有可以代表该类的聚类中心。因此提出了一种基于均值漂移的镜头聚类方法。首先是基于滑动镜头窗进行均值漂移获得视觉相似、时间相近的镜头类和相应的聚类中心, 然后根据电影视频场景内容的发展模式, 在定义镜头类的时序距离的基础上, 基于时空约束关系将镜头类组合成场景, 并且由相应的聚类中心获得场景关键帧, 最后对于场景过分割的情形进行有效地处理。

1　均值漂移算法

均值漂移 (MS)算法[7]是一种有效的统计迭代算法[8]。由于 MS算法完全依靠特征空间中的样本点进行分析, 不需要任何先验知识, 收敛速度快, 近年来被广泛应用于图像分割、关键帧提取[9]、目标跟踪等计算机视觉领域。MS算法的基本思想是, 通过反复迭代搜索特征空间中样本点最密集的区域, 搜索点沿着样本点密度增加的方向“漂移”到局部密度极大点。

1.1　非参数估计

MS算法是从密度函数梯度的非参数估计中推导获得, 而非参数估计则是从样本集出发对密度函数进行估计。其中最常用的是核密度估计, 它根据核函数 $K(\boldsymbol{x})$对样本集进行计算得到密度函数。

定义 1　$\boldsymbol{R}^d$代表一个 d 维的欧氏空间, $\boldsymbol{x}$ 是该空间中一个点, 用列向量表示。$\boldsymbol{x}$ 的模为 $\|\boldsymbol{x}\|^2=\boldsymbol{x}^{\mathrm{T}}\boldsymbol{x}$, $\mathbf{R}$ 表示实数域。若一个函数 K: $\boldsymbol{R}^d\rightarrow\mathbf{R}$ 存在一个轮廓函数 k: $[0,\infty]\rightarrow\mathbf{R}$, 即

$$K(\boldsymbol{x})=CkK(\|\boldsymbol{x}\|^2) \tag{1}$$

式中, $Ck>0$ 为标准化常数, 并且满足: ① k 是非负的; ② k 是非增的; ③ k 是分段连续的, 并且 $\int k(r)\,\mathrm{d}r<\infty$, 则函数 $K(\boldsymbol{x})$就被称为核函数[10]。

给定 $\boldsymbol{R}^d$空间中的 n 个采样点 $\{\boldsymbol{x}_i,\ 1\leqslant i\leqslant n\}$, 利用核函数 $K(\boldsymbol{x})$及正定的 $d\times d$ 带宽矩阵 $\boldsymbol{H}_i$, 密度函数的核密度估计公式为[11]

$$f(\boldsymbol{x})=\sum_{i=1}^{n}w(\boldsymbol{x}_i)|\boldsymbol{H}_i|^{-\frac{1}{2}}K(\boldsymbol{H}_i^{-\frac{1}{2}}(\boldsymbol{x}-\boldsymbol{x}_i))=\sum_{i=1}^{n}Ckw_i|\boldsymbol{H}_i|^{-\frac{1}{2}}k(\|\boldsymbol{x}-\boldsymbol{x}_i\|^2_{\boldsymbol{H}_i}) \tag{2}$$

式中, $w(\boldsymbol{x}_i)\geqslant 0$ 是采样点 $\boldsymbol{x}_i$ 的权重, 满足 $\sum w(\boldsymbol{x}_i)=1$, 简记为 w_i。核函数 $K(\boldsymbol{x})$决定了采样点 $\boldsymbol{x}_i$与核中心点 $\boldsymbol{x}$ 之间的相似性度量, 带宽矩阵 $\boldsymbol{H}_i$决定了核函数的影响范围。$\|\boldsymbol{x}-\boldsymbol{x}_i\|^2_{\boldsymbol{H}_i}=(\boldsymbol{x}-\boldsymbol{x}_i)^{\mathrm{T}}\boldsymbol{H}_i^{-1}(\boldsymbol{x}-\boldsymbol{x}_i)$称作马哈拉诺比斯 (Mahalanobis)距离。直观地说, 密度估计 $f(\boldsymbol{x})$是每个采样点处的核函数加权求和的结果。

1.2　均值漂移公式

密度函数梯度估计等于密度函数估计的梯度, 即

$$\nabla f(\boldsymbol{x})=\sum_{i=1}^{n}2Ckw_i|\boldsymbol{H}_i|^{-\frac{1}{2}}k'(\|\boldsymbol{x}-\boldsymbol{x}_i\|^2_{\boldsymbol{H}_i})^{-1}(\boldsymbol{x}-\boldsymbol{x}_i)=2Ck\left(\sum_{i=1}^{n}w_i|\boldsymbol{H}_i|^{-\frac{1}{2}}\boldsymbol{H}_i^{-1}g(\|\boldsymbol{x}-\boldsymbol{x}_i\|^2_{\boldsymbol{H}_i})\right)(m\boldsymbol{H}_i(\boldsymbol{x})-\boldsymbol{x}) \tag{3}$$

式中, $g(\boldsymbol{x})=-k'(\boldsymbol{x})$。式 (3)的第 2个等式中的第 2个括号记为 $M\boldsymbol{H}_i(\boldsymbol{x})=m\boldsymbol{H}_i(\boldsymbol{x})-\boldsymbol{x}$称为均值漂移向量, $m\boldsymbol{H}_i(\boldsymbol{x})$ 为均值漂移迭代公式。即

$$m\boldsymbol{H}_i(\boldsymbol{x})=\frac{\sum_{i=1}^{n}w_i|\boldsymbol{H}_i|^{-\frac{1}{2}}\boldsymbol{H}_i^{-1}g(\|\boldsymbol{x}-\boldsymbol{x}_i\|^2_{\boldsymbol{H}_i})\boldsymbol{x}_i}{\sum_{i=1}^{n}w_i|\boldsymbol{H}_i|^{-\frac{1}{2}}\boldsymbol{H}_i^{-1}g(\|\boldsymbol{x}-\boldsymbol{x}_i\|^2_{\boldsymbol{H}_i})} \tag{4}$$

它表示采样点的加权平均值, 类似于“重心”的概念。一般 $m\boldsymbol{H}_i(\boldsymbol{x})$处的密度大于原核中心点 $\boldsymbol{x}$ 处的密度, 因此均值漂移向量 $M\boldsymbol{H}_i(\boldsymbol{x})$总是指向密度大的方向, 即密度梯度增加的方向。均值漂移算法的收敛点为局部密度极大值点。

1.3 均值漂移算法

若设采样点的权重相等, 即 $w(\boldsymbol{x}_i)=1/n$, 带宽矩阵与单位矩阵成正比 $\boldsymbol{H}_i=h^2\boldsymbol{I}$, 则均值漂移迭代公式为

$$\boldsymbol{x}_{i+1}=mh(\boldsymbol{x})=\frac{\sum_{i=1}^{n}g(\|(\boldsymbol{x}-\boldsymbol{x}_i)h^{-1}\|^2)\boldsymbol{x}_i}{\sum_{i=1}^{n}g(\|(\boldsymbol{x}-\boldsymbol{x}_i)h^{-1}\|^2)} \tag{5}$$

若使用高斯核, 则

$$\boldsymbol{x}_{i+1}=mh(\boldsymbol{x})=\frac{\sum_{i=1}^{n}\exp(\|(\boldsymbol{x}-\boldsymbol{x}_i)h^{-1}\|^2)\boldsymbol{x}_i}{\sum_{i=1}^{n}\exp(\|(\boldsymbol{x}-\boldsymbol{x}_i)h^{-1}\|^2)} \tag{6}$$

给定核函数 $K(\boldsymbol{x})$ (文中实验选用高斯核)和容许误差 ε, 则 MS算法的步骤如下:

1) 对特征空间 S 中任意一点 X_O, 设置搜索区域圆 O, 其半径为带宽 h, 如图 1所示;

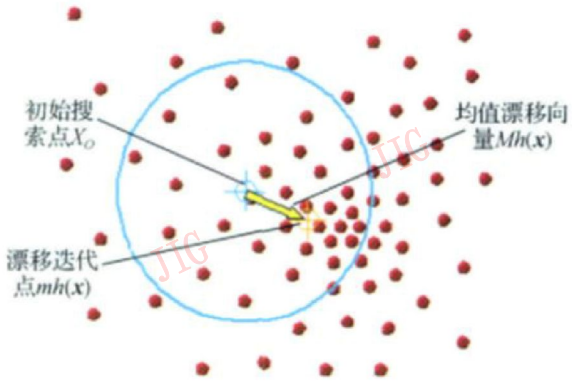


图 1 均值漂移一次迭代示意图

Fig 1 Demonstration of mean shift once

2) 根据式 (6)计算圆 O 中采样点的均值 $mh(\boldsymbol{x})$, $mh(\boldsymbol{x})$处的密度大于圆心 X_O 处的密度, 并标记此轮迭代过程中属于该类的采样点;

3) 计算圆心 X_O 与均值 $mh(\boldsymbol{x})$之差, 它表示均值漂移向量 $Mh(\boldsymbol{x})$, 该向量总是指向密度增加的方向;

4) 如果均值漂移向量的模小于容许误差即 $\|Mh(\boldsymbol{x})\|<\varepsilon$ 则对从初始点 X_O 出发的漂移聚类迭代完毕, 而且这一类的聚类中心是点 $mh(\boldsymbol{x})$; 否则执行步骤 5);

5) 将均值 $mh(\boldsymbol{x})$赋给圆心 O, 执行步骤 2)。

MS算法原理简单、迭代效率高, 收敛快, 而且能自动确定类别, 不需用户事先指定类别数。数据集中的每一点都可作为初始点, 分别执行均值漂移算法, 收敛到同一点的算做一类。如图 2所示, 是对 2维空间 750个采样点在 Matlab环境里的均值漂移聚类, 聚类结果是 3类, 分别对应红、兰和绿色, 其中点 O 是相应类的聚类中心, 聚类时间是 0.204 732 s。

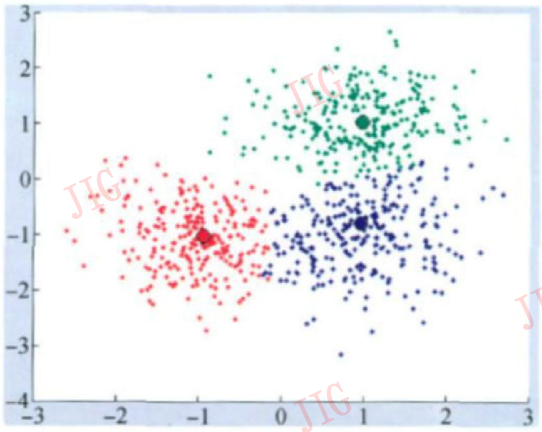


图 2 均值漂移示例图

Fig 2 Example of MS

2 基于滑动镜头窗的均值漂移镜头聚类和视频场景检测

2.1 基于滑动镜头窗的均值漂移镜头聚类

2.1.1 滑动镜头窗

视频中的一个场景包含的镜头数目远远少于整个视频 (长达几千个镜头)。为减少镜头距离的计算次数, 提高聚类速度, 镜头聚类时是基于滑动镜头窗的, 即只需比较当前窗口内镜头间的距离 (相似度), 不必一次计算整个视频所有镜头间的距离。而且基于滑动镜头窗使得时序距离很远的镜头不进行相似度比较, 保证了只有同一个场景的相似镜头才被聚类为一个镜头类, 提高了场景检测的精度。

滑动镜头窗是一滑动的窗口, 大小为 L (L 是镜头的个数)。由于包含在一个场景的镜头数目通常是小于 50, 因此 L 的值设定为 50, 以保证同一个场景的所有镜头在一个镜头窗内。当然, 一个镜头窗中可能包含两个或两个以上场景的镜头。镜头窗以 Lm 的增量移动。如图 3所示, Lm 是一个变量, 它的

计算依据是在电影视频中, 一个场景的相似镜头时序距离通常小于 3, 如果从某个镜头开始以后连续 3个镜头与前面的镜头都不相似 (即不属于相同镜头类), 则认为该镜头是下一个场景的起始。*Lm* 的计算式为 (7)和式 (8)。

$$Lm(i)=\begin{cases} i & C(i+k)\neq C(j);\ i=1,2,\cdots,L;\\ & j=1,2,\cdots,i;\ k=1,2,3 \\ 0 & \text{其他}\end{cases} \tag{7}$$

$$Lm=\begin{cases} Lm(i) & (Lm(i)=i)\ \text{且}\ (Lm(j)\neq 0)\\ 0 & Lm(i)=0\end{cases} \tag{8}$$

式 (7)中, $C(j)$表示镜头 j所属的种类。

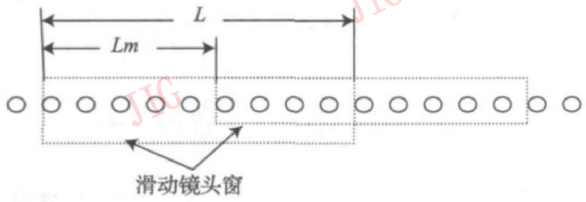


图 3　滑动镜头窗示意图

Fig. 3　Sketch map of slideing shot window

2.1.2 镜头间的距离

由于颜色特征有较强的辨识能力, 被广泛地用于表示图像内容, 本文使用颜色直方图来描述帧的视觉内容。又因为 HSV 颜色空间与人的视觉感知系统有较好的一致性, 所以选用 HSV 颜色空间模型。H 分成 8份, S和 V 各分成 3份, 并且按照色彩的不同范围和主观颜色感知进行不等间隔量化。设镜头 i的关键帧集合 $\boldsymbol{KF}_i=(Kf_{i1}, Kf_{i2}, \cdots, Kf_{in})$, 则镜头 i的视觉特征表示为 HSV 空间的归一化的颜色直方图 $Hist(Kf_{im})$, $m=1,2,\cdots,n$。这样, 通过提取镜头的视觉特征, 镜头 i表示为 $shoti=(\boldsymbol{KF}_i, Hist(Kf_{im}))$。

定义两帧 I, J之间的视觉相似度为

$$S(I,J)=\sum_{k=1}^{L-1}\min(Hist_I(k), Hist_J(k)) \tag{9}$$

式中, $Hist_I(k)$为图像 I的归一化直方图, L为量化台阶数。则镜头 i, j间的距离为

$$x_i-x_j=Dist(shoti, shotj)=1-\max_{I\in KF_i, J\in KF_j}[S(I,J)] \tag{10}$$

式中, KF_i和 KF_j分别是对应镜头 i, j的关键帧集合; I和 J是相应关键帧集合中的任意关键帧。

2.1.3 基于滑动镜头窗的均值漂移镜头聚类

基于滑动镜头窗的均值漂移镜头聚类算法

输入: 镜头序列 $\boldsymbol{Shots}=\{Shot1, Shot2, \cdots, ShoM\}$和视觉特征空间的带宽 hc。

输出: 镜头类序列 $\boldsymbol{ShotClusters}=\{SC_j\}$和镜头类代表 $\{Y_j\}$, $j=1,2,\cdots N$; $1\leqslant N\leqslant M$。

1) 在当前滑动镜头窗的镜头序列 ***Shots*** 中对各镜头提取特征信息, 其中第 i个镜头对应特征空间的第 i个点, 表示为 $x_i=(KF_i, Hist(Kf_{im}))$。

2) 对每个特征点执行均值漂移, 直到收敛得到聚类中心 Zd, 存储所有聚类中心 $\{Zd\}$、相应的镜头类序列 $\{SC_d\}$, $d=1,2,\cdots,W$; $1\leqslant W\leqslant M$。

3) 对聚类中心合并。将距离不超过带宽 hc的所有聚类中心 Zd合并, 相应的类成员和类也合并, 形成新的聚类中心 $\{G_j\}$及镜头类序列 $\{SC_j\}$, $j=1,2,\cdots,N$ ($1\leqslant N\leqslant W\leqslant M$)。聚类中心 $\{G_j\}$是相应类序列 $\{SC_j\}$的密度极大值点, 若 $\{SC_j\}$中镜头数 $\geqslant 2$, 则在 $\{SC_j\}$中根据式 (9)和式 (11)求出最接近 $\{G_j\}$的关键帧 K_j, 用它代表类别 $\{SC_j\}$。这样, 镜头序列经过均值漂移, 形成了镜头类序列 $\{SC_j\}$和相应的类别代表 $\{K_j\}$。

4) 根据式 (7)和式 (8)计算滑动镜头窗的移动增量 Lm, 以 Lm为增量移动镜头窗。再转步骤 1), 对新的滑动镜头窗内镜头进行均值漂移聚类, 直至所有镜头聚类完毕。

在镜头类序列 $\{SC_j\}$中求出最接近聚类中心 $\{G_j\}$的关键帧 K_j的式为

$$S(G_j,K_j)=\max_{\substack{J_p\in KF_p\\ p\in SC_j}}(S(G_j,J_p)) \tag{11}$$

式中, p是镜头类序列 $\{SC_j\}$中的镜头, $\boldsymbol{KF}_p$是镜头 p的关键帧集合, J_p是 $\boldsymbol{KF}_p$中任一帧, 式 (11)中最大值成立时的关键帧 J_p就是所求的关键帧 K_j, 即 G_j和 K_j之间的相似度最大。

2.2　视频场景的检测

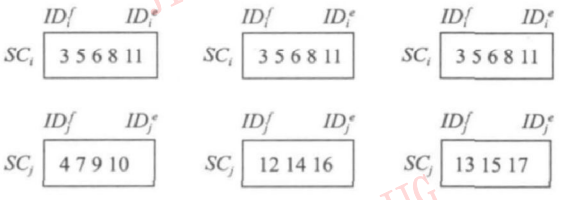
视频场景是由一组语义相关的镜头类组成。根据电影制作和编辑中常用场景特征, 视频场景中内容的发展模式可分为 3类[5,6]: 第 1类是顺序进展模式, 组成场景的各镜头在色彩、光照保持了连贯性, 这些镜头之间有更高的视觉相似度; 第 2类是交错进展模式, 组成场景的镜头在视觉上可能相似也可能有很大差别, 但它们表达着同一个主题, 交替显示, 如在对话片段中, 镜头常在对话的双方之间来回移动; 第 3类是混合进展模式, 同时包含上述 2种模式, 内容相似的镜头以及交替显示的镜头组合起来表达一个完整的故事情节。顺序进展的场景由一个或几个视觉上相似的镜头类构成; 交错进展和混合进展的场景由视觉上相似且时序上交叠的镜头类构

成。可见一个镜头类描述了视频场景的一个侧面或故事线索,而多个镜头类则代表了多个故事线索,这些镜头类在时间维上相互交叠、衔接,共同表达了同一个主题。因此本文通过沿视频流追踪镜头在多个镜头类之间的“跳动”顺序,来判断场景的边界,将表达同一语义的镜头类组织到同一个场景。

定义 (两个镜头类的时序距离)设 ID_i^f, ID_i^e 分别是镜头类 $\{SC_i\}$的最小的镜头序号和最大的镜头序号,定义两个镜头类的时序距离为

$$TD(SC_i, SC_j) = \begin{cases} 0 & (ID_j^e - ID_i^f > L) \text{ 或 } (ID_i^e - ID_j^f > L) \\ ID_j^f - ID_i^e & (ID_j^f > ID_i^f) \text{ 且 } (ID_j^e - ID_i^f \leqslant L) \\ ID_i^f - ID_j^e & (ID_j^f < ID_i^f) \text{ 且 } (ID_i^e - ID_j^f \leqslant L) \end{cases} \quad (12)$$

定义两个镜头类的时序距离时先分为两种情况,即 $|ID_j^e - ID_i^f| > L$ (对应式(12)中第1行情况)和 $|ID_j^e - ID_i^f| \leqslant L$ (对应式(12)中第2行和第3行情况)。在 $|ID_j^e - ID_i^f| > L$ 时则镜头类 SC_i, SC_j 一定属于不同的场景,因为若属于同一场景,则该场景的镜头总数的绝对值 $|ID_j^e - ID_i^f|$ 已超过第2.1.1节中所说的一个场景的镜头数目的极限 L,定义此时的 $TD(SC_i, SC_j) = 0$。由上推理可知在 $|ID_j^e - ID_i^f| \leqslant L$ 时,镜头类 SC_i, SC_j 可能属于同一场景,此时又分为式(12)中第2行和第3行两种情况。若 $TD(SC_i, SC_j) < 0$,则两个镜头类是交错关系;若 $TD(SC_i, SC_j) = 1$,则两个镜头类是相邻关系;若 $TD(SC_i, SC_j) > 1$,则是分离关系。如图4所示。



(a) 交错关系 (b)相邻关系 (c)分离关系

图4 两个镜头类的时序关系

Fig 4 Temporal relationship between two shot clusters

对于可能属于同一场景的镜头类,就需要进行场景检测。场检测的思路是首先合并所有有交错关系的镜头类,然后根据式(9),式(13)和式(14)判断任意两个相邻关系的镜头类的相似度是否大于给定的阈值 TS,若满足条件,则合并成一个镜头类,否则不合并。最终形成的每个镜头类对应一个场景。式(13)是计算两个镜头间的视觉相似度,式(14)在式(13)的基础上计算两个镜头类 SC_m, SC_n 间的视觉相似度。

$$SimSS(shot_i, shot_j) = \max_{I \in KF_i} \max_{J \in KF_j} (S(I, J)) \quad (13)$$

$$SimCC(SC_m, SC_n) = \frac{1}{|SC_m|} \sum_{k=1}^{|SC_m|} \max_{shot_j \in SC_n} (SimSS(shot_k, shot_j)) \quad (14)$$

式(14)中基于这样一个假设即镜头类 SC_m 的镜头数 $|SC_m|$ < 镜头类 SC_n 的镜头数 $|SC_n|$。

场景检测算法

输入:镜头类序列和相应的类别代表

输出:场景序列及场景关键帧

1)初始化,输入镜头类序列 $\{SC_1, SC_2, \cdots, SC_t\}$ 和相应的类别代表 $\{K_1, K_2, \cdots, K_t\}$;

2)对于任意的两个镜头类,如果是交错关系,则合并为一个新的镜头类;

3)重复步骤2),直到不再有交错关系的镜头类;

4)若两个相邻关系的镜头类满足条件 $SimCC(SC_m, SC_n) \geqslant TS$,则合并;

5)重复步骤2),直到不再有满足条件的相邻关系的镜头类可以合并;

6)所有合并产生的新的镜头类和未合并的镜头类都被看作是场景。而组成场景的镜头类 $\{SC_i\}$ 对应的镜头类代表 $\{K_i\}$则是相应场景的关键帧,其中 $1 \leqslant i \leqslant t$。

2.3 场景检测后续处理

镜头类相似度阈值 TS 的取值越大,则检测得到的场景就越多,易造成过度分割;若取值越小,则检测得到的场景就越少,将导致真实的场景检测不出来。相对而言,过度分割所造成的危害较小一些,因为被过度分割的场景可通过某种方法恢复出来,而没有检测出的场景却很难再恢复出[12]。因此,在合并场景时实验中采取让 TS 取比较大的数值,然后再对过度分割出的场景进行合并的方法。通过对电影中真实场景的研究,发现一个真实的场景所包含的镜头多数情况下应该不少于3个,所以对于任何所包含的镜头个数小于3的场景都认为是误检,应进行合并[2]。场景合并过程如下,首先找出镜头个数小于3的场景 C,和 C 最前面和最后面所包含镜头个数不小于3的场景 CF 和 CB,根据式(9)、式(13)和式(15)计算出场景 C 中镜头与 CF 和 CB 中镜头各自的最大相似度 $SimSe(C, CF)$ 和 $SimSe(C, CB)$,如果

$SimSe(C, CF) \geqslant SimSe(C, CB)$, 则场景 C 合并到 CF 中, 否则把 C 合并到 CB 中, 这样重复下去, 直至所有场景所包含的镜头个数都不小于 3为止。

$$SimSe(SceneA, SceneB) = \max_{shoti \in SceneA,\ shotj \in SceneB} (SimSS(shoti, shotj)) \qquad (15)$$

3 实验结果及分析

为检验本文方法, 选取电视剧《金婚》(记为 JH)、《突围行动》(TW)、《奋斗》(FD)、及《别了温哥华》(BW)里部分视频内容变化较平缓的视频片段作为试验对象, 如表 1所示。实验结果如表 2所示。相对以往的基于阈值比较的聚类, 虽然均值漂移聚类的方法也需要设置带宽参数 hc, 但是带宽是范围取值, 而阈值是点取值, 相对而言, hc的选取比阈值易于控制, 而且均值漂移聚类能够得到各类中心即镜头类代表 $\{K_j\}$, 可用作为场景关键帧。从表 2中可以看出, 本文方法的效果很好, 其中查全率均在 90% 以上, 而且其查全率和准确率都比文献 [5]、[6]中的高, 这主要是因为在文献 [5]、[6]中都是基于难以控制的阈值进行镜头聚类, 而且都没有镜头过分割后续处理。

表 1　测试视频信息

Tab. 1　The data of test videos

视频片段	时间	镜头数	镜头类别数	实际场景数
JH	17′46″	146	78	15
TW	10′05″	117	49	10
FD	11′39″	131	38	6
BW	15′48″	149	82	13

表 2　场景检测算法的检测结果

Tab. 2　Results of scene detection algorithms

%

视频片段	本文方法		文献 [5]方法		文献 [6]方法	
	查全率	准确率	查全率	准确率	查全率	准确率
JH	93.3	82.3	86.7	81.3	80.0	75.0
TW	90.0	81.8	80.0	80.0	80.0	72.7
FD	100.0	75.0	100.0	66.7	83.3	71.4
BW	92.3	80.0	84.6	78.6	84.6	73.3
平均	93.9	79.8	87.8	76.7	82.0	73.1

图 5给出对《突围行动》里部分视频片段 (长度为 56 s)进行场景检测的示例显示。这部分视频有 16个镜头 (每个镜头由一个关键帧表示), 均值漂移后聚类为 5类, 检测结果为一个场景, 镜头类为: 类 1包含镜头 1, 6和 8; 类 2包含镜头 2, 4, 9, 11和 14;

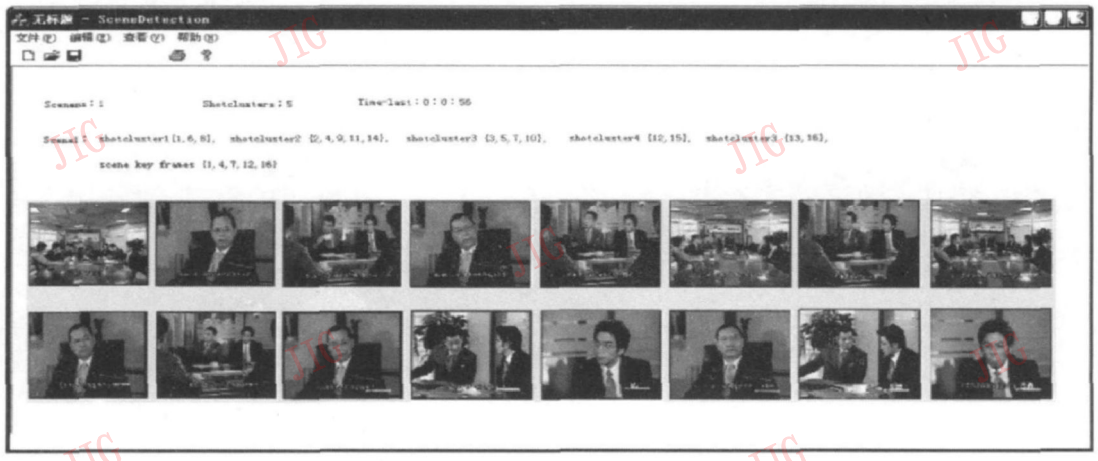


图 5　《突围行动》里的部分视频片段检测示例

Fig. 5　Example of scene detection for partial video in TW

类 3包含镜头 3, 5, 7和 10; 类 4包含镜头 12和 15; 类 5包含镜头 13和 16; 这几个类之间都是交错关系, 属于同一场景, 而场景关键帧为帧 1, 4, 7, 12和 16。

本文方法主要是根据视觉特征进行均值漂移聚类, 然后场景检测。所以如实验所示, 该方法适合视频内容变化较平缓的电影视频场景检测。对于镜头切换频繁, 视频内容变化较剧烈的动作片, 因为某些不相似的镜头在语义上也可能属于同一个场景, 所以单纯通过视觉特征, 场景检测效果相对要差些。

对于这样的视频,仍然可以采用提出的基于均值漂移的场景检测思路,只需在特征空间中融合声音、字幕、运动等其他特征。

4 结 论

提出了一种新颖的基于均值漂移聚类的电影视频场景检测方法。先是基于均值漂移,在滑动镜头窗内对镜头聚类,并且获得相应的镜头类关键帧。然后根据镜头类间的时序距离的关系,将镜头类组合成场景,并且由相应的镜头类关键帧获取场景关键帧。最后为了弥补阈值难以控制的缺点,对于场景过分割的情形进行有效地处理。实验证实该方法聚类快,效率高,为后期基于语义的视频检索打下了很好的视频结构化基础。

参考文献 (References)

[1] Zhang H J. Content-based video analysis, retrieval and browsing [DB/OL]. http // research microsoft com/asia/download/disquisition/0202e asp, 2002.

[2] Wang Xue-jun, Ding Hong-tao, Chen He-xin. A shot clustering based approach for scene segmentation[J]. Journal of Image and Graphics, 2007, 12(12): 2121-2131. [王学军, 丁红涛, 陈贺新. 一种基于镜头聚类的视频场景分割方法[J]. 中国图象图形学报, 2007, 12(12): 2127-2131.]

[3] Yeung M, Yeo B-L, Liu B. Segmentation of video by clustering and graph analysis [J]. Computer Vision and Image Understanding, 1998, 71(1): 94-109.

[4] Rui Y, Huang T, Mehrotra S. Constructing table-of cont-ent for videos[J]. Multimedia Systems, 1999, 7(5): 359-368.

[5] Zhao Ya-qin, Zhou Xian-zhang, He xin. Automatically generating hierarchical summary for film video[J]. Journal of Image and Graphics, 2007, 12(8): 1412-1417. [赵亚琴, 周献中, 何新. 一种层次的电影视频摘要生成方法[J]. 中国图象图形学报, 2007, 12(8): 1412-1417.]

[6] Cheng Wen-gang, Xu De, Lang Cong-yan. An efficient method for video scene detection[J]. Journal of Image and Graphics, 2004, 9(8): 984-990. [程文刚, 须德, 郎从妍. 一种有效的视频场景检测方法[J]. 中国图象图形学报, 2004, 9(8): 984-990.]

[7] Zhou Fang-fang, Fan Xiao-ping, Ye Zhen. Mean shift research and application[J]. Control and Decision, 2007, 22(8): 841-847. [周芳芳, 樊晓平, 叶榛. 均值漂移算法的研究与应用[J]. 控制与决策, 2007, 22(8): 841-847.]

[8] Fukunaga K, Hostetler L D. The estimation of the gradient of a density function with applications in pattern recognition[J]. IEEE Transactions on Information Theory, 1975, 21(1): 32-40.

[9] Chen Zhuo-yi. Key-frame extraction using nonparametric clustering based on density estimation[J]. Computer Science, 2007, 34(4): 119-162. [陈卓夷. 基于非参数密度估计聚类的关键帧提取方法[J]. 计算机科学, 2007, 34(4): 119-162.]

[10] Cheng Y Z. Mean shift, mode seeking, and clustering[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1995, 17(8): 790-799.

[11] Comaniciu D, Meer P. Mean shift: A robust approach toward feature space analysis [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(5): 603-619.

[12] Yeung M, Yeo B-L, Liu B. Segmentation of video by clustering and graph analysis [J]. Computer Vision and Image Understanding, 1998, 71(1): 94-109.