

一种有效的视频场景检测方法

程文刚 须德 郎丛妍

(北京交通大学计算机与信息技术学院, 计算机研究所, 北京 100044)

摘要 合理地组织视频数据对于基于内容的视频分析和应用有着重要的意义。现有的基于镜头的视频分析方法由于镜头信息粒度太小而不能反映视频语义上的联系, 因此有必要将视频内容按照高层语义单元——场景进行组织。提出了一种快速有效的视频场景检测方法, 根据电影编辑的原理, 对视频场景内容的发展模式进行了分类, 给出了场景构造的原则; 提出一种新的基于滑动镜头窗的组合方法, 将相似内容的镜头组织成为镜头类; 定义了镜头类相关性函数来衡量镜头类之间的相关性并完成场景的生成。实验结果证明了该方法的快速有效性。

关键词 镜头 镜头类 场景 滑动镜头窗 镜头类相关性函数

中图分类号: TP391 TN941.1 **文献标识码**: A **文章编号**: 1006-8961(2004)08-0984-07

An Efficient Method for Video Scene Detection

CHENG Wen-gang, XU De, LANG Cong-yan

(Institute of Computer, School of Computer & Information Technology, Beijing Jiaotong University, Beijing 100044)

Abstract It is important to organize the unstructured video data properly for content video analysis and application. However, existing shot-based video analysis can not get good performance for effective browsing and retrieval because shots are too numerous to handle and can not convey meaningful semantics, so it is necessary to organize video content with the scene structure, which is a more meaningful and high-level semantic video unit. In this paper, a simple and efficient method is presented for video scene detection. As the buildup of video scene should obey the film grammar, it concludes the types of the scenes that are widely used in filmmaking firstly. The rules to generating the video scene are put out then. Based on the rules, a scheme for video scene detection is proposed; The video stream is segmented into shots through shot boundary detection; Key frame extraction is performed based on the content variation of shots; A new clustering method based on the sliding shot window is used to group the shots into shot clusters; A correlation function between shot clusters is defined to analysis the correlations of them and to construct the final video scene structure. Experimental results verify it is an efficient method.

Keywords shot, shot cluster, scene, sliding shot window, correlation function of shot clusters

1 引言

随着多媒体技术、数字电视和网络的发展, 产生了大量的视频文档。视频以其内容丰富直观而得到越来越广泛的应用。但由于视频与文本数字等数据的特性不同, 基于内容的分析和存取成为处理视频数据的有效手段。视频数据是连续的数据流, 为了方便地检索和浏览, 需要对视频数据结构化, 按照内容组织存储。

镜头(shot)是摄像机的一个连续拍摄过程, 表示了一个时间或空间上连续的动作。虽然基于镜头的视频分析提供了一种处理视频流的有效方式, 但是由于镜头所表达的信息粒度太小, 不便于语义级的视频浏览和检索, 因为用户可能更关心故事单元而不是单个的动作。特别是, 一个长视频可能包含了太多的镜头, 如果把镜头作为视频内容组织的单元, 粒度过小的问题会显得尤为突出。场景(scene)是语义相关、时间相近镜头的组合, 表达了一个完整故事单元, 是比镜头更高一层的视频内容组织单位。因

此,在镜头边界检测技术日趋成熟的基础上,研究视频场景的检测(或生成)是很有意义的。

目前的视频场景检测方法大体分为两类:基于模型的方法和基于电影规则的方法。在第一类方法中,首先需要根据特定应用或领域建立一个先验模型(a priori model)。Swangberg 等人提出了这类方法的一个理论框架^[1],并且在新闻节目的分析^[2,3]和体育节目的分析^[4]中得到了实现和应用,检测的准确率比较高。但是这类方法在每一次应用之前都要建立一个领域模型,需要很好的领域知识,因而其应用范围受到限制。在第二类方法中,基于场景转移图(scene transition graph, STG)的方法^[5],提供了视频流中故事的发展线索;综合运用视频、音频等信息^[6,7],采用类似镜头边界检测的算法来检测场景的边界,但是这种方法并没有将镜头看作一个整体,开始帧和结束帧对检测的影响较大,同时由于视频的多功能性,使得视频和对应音频之间有多种关系,因此如何有效地综合视频和音频信息仍然是一个困难的问题,场景检测的效果也比较有限^[7];应用统计方法,如隐马尔科夫模型^[8]或贝叶斯工具^[9],也可以得到视频的高层描述结构;基于时间约束的镜头聚类算法及其改进算法^[5,10]也用来组织镜头,生成视频场景结构。

目前的场景检测算法都比较复杂,需要很大的计算量和运算时间。为此提出一种新的自动检测视频场景的方法,该方法包括两部分:镜头类的形成和镜头类之间相关性的分析。实验证明了它的快速有

效性。

2 场景检测方法

视频场景由一组语义相关的镜头组成,描述了一个独立的高层语义单元。同一场景内的镜头在时间上相邻,可能发生在同一个地点共享同一背景,也可能是同一个内容的不同侧面,它们组合起来表达了一个完整的语义。视频场景的构成要符合电影的编辑原则,在分析这种原则的基础上提出了场景检测算法:通过镜头边界的检测,将视频流分割为镜头,根据镜头内容的变化来动态提取关键帧;将时间相近,内容相似的镜头组合为镜头类;分析镜头类之间的相关性来完成视频场景的检测。

2.1 视频场景的组成原则

N-type 和 M-type 为两种视频场景的构成模式^[6]。通过对电影制作和编辑中常用场景特征的分析,基于以上两种视频场景的构成模式定义,把视频场景中内容的发展模式分为 3 类:顺序进展、交错进展和混合进展。

(1) 顺序进展

组成场景的各镜头在色彩、光照或声音上保持了连贯性,所有这些镜头在视觉和听觉上相似。图 1 是这种模式的一个例子(每个镜头用一个关键帧来代替,下同)。



图 1 场景顺序进展模式示例(取自 Shakespeare In Love)

(2) 交错进展

组成场景的镜头在视觉上可能有很大差别,但它们表达着同一个主题,交替显示。图 2 和图 3 是这种模式的例子。图 2 是一个对话的片段,镜头在对话的双方之间来回移动,这在电影视频以及访谈节目中是很普遍的。图 3 是杀手暗杀一个人的片段,镜头在杀手和目标之间交错。类似的例子还有很多,例如



图 2 场景交错进展模式示例(取自 Out of Africa)

人看窗外的东西,窗外的东西可能不停地在变化,而人不变,镜头在窗外和窗内不停地切换。

(3) 混合进展

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。图 4 描述了这种进展方式。

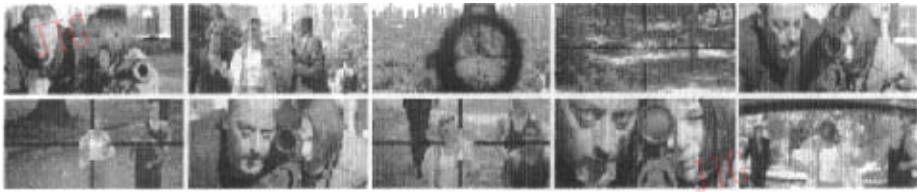


图 3 场景交错进展模式示例(取自 Leon)



图 4 场景混合进展模式示例(取自 Leon)

定义镜头类(shot cluster)为一组时间相近、内容相似镜头的集合。可以看出,在顺序进展模式中,组成场景的镜头都来自同一个镜头类;在交错进展模式中,镜头可能来自两个或多个镜头类,取自不同镜头类中的镜头交替显示;在混合模式中,镜头也是来自几个镜头类,与交错进展模式不同的是,它的交错频率可能没有前种方式高,但是镜头类之间必定有交错。可以看出镜头类是组成场景的基本内容单元,表达了单一的内容,而语义相关的镜头类则组成了场景;镜头类中的镜头具有时间相近和内容相似的特点,可以通过比较时间相近镜头的内容相似度来得到;而语义是否相关则可以通过分析镜头类之间的交错来判断,因此,视频场景检测的方法是首先形成镜头类,然后分析镜头类之间的关系来判断场景的边界。

2.2 镜头边界的检测和关键帧的提取

视频分割的目的是通过检测各种编辑效果(如 cut, fade in/out, dissolve 等)在原始视频流中的位置,来得到单个的镜头,即镜头边界检测。“双阈值”方法是一种简单有效的方法,应用这种方法进行渐变检测时,如果相邻两帧的帧间差小于低阈值 T_1 ,而累计帧间差还没有大于高阈值 T_2 时,渐变的起始边界便被重置。但在实际的渐变发生过程中,相邻两帧的帧间差有可能低于设定的低阈值 T_1 ,按照原算法就会造成漏检。因此,改进了“双阈值”方法:设

定一个数值 N_{low} ,如果渐变检测中累计出现这样的帧数不大于 N_{low} ,仍然对其继续进行渐变检测。

同一镜头中的各帧图像内容比较相似,如果简单地使用镜头中的所有帧来表示镜头,信息冗余太大且不方便计算,因此在视频分析中经常使用关键帧来表示镜头。关键帧是一个镜头的帧序列中能够反映镜头主要内容的帧,当镜头的内容变化比较大时,需要多个关键帧来表示它。

由于颜色特征有较强的辨识能力,它被广泛地用于表示图像内容。使用颜色直方图来描述帧的视觉内容。定义两帧 f_i 和 f_j 之间的视觉相似度为

$$SimFF(f_i, f_j) = \sum_{l=1}^{bins} \min(Hf_i(l), Hf_j(l)) \quad (1)$$

式中, $bins$ 表示直方图的 bin(盒子)的数目; Hf_i 和 Hf_j 分别是两帧 f_i 和 f_j 的归一化直方图。由于 HSV 颜色空间与人的视觉感知系统有较好的一致性,因此选用 HSV 颜色空间来表示帧的颜色分量。按照人的视觉分辨能力,把 H 分成 8 份, S 和 V 各 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的量化。

为了衡量镜头中其余的帧 f_i 与已有的关键帧集合 $KF = \{F_1, F_2, \dots, F_{N_k}\}$ (N_k 为集合中关键帧的数目)的相似度,定义

$$SimFK(f_i, KF) = \max(SimFF(f_i, F_n)) \quad (F_n \in KF) \quad (2)$$

选取镜头中的首帧作为关键帧标记为 F_1 , 用

式(2)中定义的相似度,顺序计算后继帧 f_i 与已有关键帧 $F_n(n=1, \dots, N_k)$ 的相似度,如果相似度小于阈值 T_k ,则 f_i 成为新的关键帧, $N_k=N_k+1$,直至镜头的最后一帧。这样就能保证选取的关键帧能够适应视频内容的变化,既减少了信息的冗余,又反映了镜头的内容。

2.3 基于滑动镜头窗的镜头类形成方法

将时间相近,内容相似的镜头组合在一起形成镜头类,最直接常用的方法就是层次聚类(hierarchical clustering)。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,算法的效率很低。考虑到组成场景的各镜头在时间上相近,在文献[5]中,设定时间阈值 T (根据帧的编号差确定),如果两个镜头之间的时间距离超过 T ,那么两个镜头之间的相似度为 0;在文献[10]中,提出了一种随着时间增长相似度降低的时间自适应的镜头聚类算法。较之无约束的层次聚类,减少了部分的计算量,但是除了没有改变比较次数之外,它们还存在着另外两个缺陷。首先,根据电影编辑的原理,为了突出效果,很多镜头的长短不一。使用长镜头,说明视频要表达一种抒情、安静的感觉;而使用短镜头,说明这个镜头表示一种动态、快

乐的感觉^[11]。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。其次,在一个包含若干个镜头的短片段(如一个场景)中,两个镜头之间的相似度不应该随时间间隔的变化而变化,相距较远的镜头也可能具有较大的相似度。

根据以上的分析,提出一种基于滑动镜头窗的镜头聚类方法来形成镜头类。滑动镜头窗是一个滑动的窗口,以镜头为单位,其意义在于:设滑动镜头窗的大小为 L (L 是镜头的个数),计算相似度时,当前进行相似度分析的镜头(称为当前镜头, CurShot)只跟其窗口内的 L 个镜头(称为目标镜头, DestShot)或目标镜头所在的镜头类(如果目标镜头已经与其他镜头合并为一类或者单独作为一类)进行比较。

通常,滑动窗口包含以当前镜头为中心的 L 个镜头。但是在实际进行镜头序列的相似度分析时,是按照时序即镜头出现的次序来分析的,因此设置窗口的位置是从当前镜头开始的后面 L 个镜头。如图 5 所示(镜头 $i-3$ 单独为一类,镜头 $i-2$ 和镜头 $i-1$ 聚为一类,当前镜头为镜头 i , $L=3$):当镜头 i 之前的 3 个镜头分别作为当前镜头时,已经与镜头 i 计算过相似度,所以只需进行当前镜头 i 与后面窗口宽度(L)个镜头作比较即可。

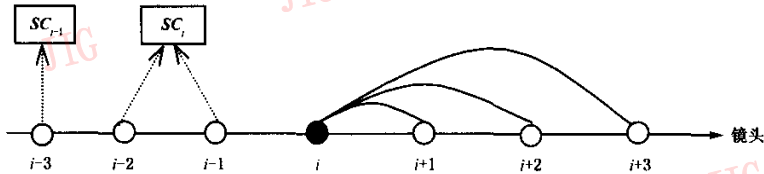


图 5 滑动镜头窗示意图

为了衡量任意两个镜头之间的相似度,定义:

$$SimSS(Shot_i, Shot_j) = \max_{k \in KF_i, l \in KF_j} (SimFF(k, l)) \quad (3)$$

式中, KF_i 和 KF_j 分别是对应 $Shot_i$ 和 $Shot_j$ 的关键帧集合; k 和 l 是相应关键帧集合中的任意关键帧。

定义镜头 $Shot_i$ 与镜头类 SC_p 之间的相似度为

$$SimSC(Shot_i, SC_p) = \max_{Shot_j \in SC_p} (SimSS(Shot_i, Shot_j)) \quad (4)$$

式中, $Shot_j$ 表示镜头类 SC_p 中的任意镜头。

定义两个镜头类 SC_p 与 SC_q 之间的相似度为

$$SimCC(SC_p, SC_q) = \max_{Shot_i \in SC_p, Shot_j \in SC_q} (SimSS(Shot_i, Shot_j)) \quad (5)$$

式中, $Shot_i$ 表示镜头类 SC_p 中的任意镜头; $Shot_j$ 表示镜头类 SC_q 中的任意镜头。

式(3)、(4)、(5)使用 \max 函数主要是为了最大限度地相似内容镜头组合在一起。

根据以上几个相似度的定义,基于滑动镜头窗的镜头类形成算法如下:

算法 1 基于滑动镜头窗的镜头类形成算法

输入: 镜头序列 $Shots = \{Shot_1, Shot_2, \dots, Shot_M\}$

输出: 镜头类序列 $ShotClusters = \{SC_1, SC_2, \dots, SC_N\} (1 \leq N \leq M)$

步骤:

(1) 初始化,输入镜头序列 $Shots$ 。当前镜头是第 1 个镜头 $CurShot = Shot_1, k = 1$ 。

(2) 如果镜头序列 $Shots$ 为空,算法停止,否则

转第 3 步;

(3)从 Shots 中取出当前镜头 $CurShot = Shot_k$ 。计算 $CurShot$ (或者 $CurShot$ 所在的镜头类) 与其滑动镜头窗内的其他镜头 (或者目标镜头所在镜头类) 的相似度, 如果超过阈值 T_k , 则将其合并为一类。否则, $CurShot$ 单独作为一类, 转第 4 步;

(4)当前镜头成为镜头序列中的下一个镜头: $Shots = Shots - \{Shot_k\}$, $k = k + 1$, 转第 2 步。

设视频序列中共有 Num 个镜头, 按文献[5]、[10]中的算法, 都需要比较 $Num \times (Num - 1) / 2$ 次, 而基于滑动镜头窗的镜头类形成算法最好情况只需要 $Num \times L$ 次比较, 最坏情况下才可能达到上面的复杂度, 但是由于视频序列中的情节是随时间的变化不断发展的, 所以最坏情况一般不会出现。镜头组合的算法大大减少了比较次数, 减少了计算量, 而且更为合理。镜头经过组合, 形成了镜头类。

2.4 镜头类的相关性分析

根据上节中的分析, 可以认为场景是由一个或几个镜头类组成的。如果一个场景包含几个镜头类, 则取自这几个镜头类的镜头应该是交错出现的, 即镜头类之间存在相关性。

为了描述这种相关性, 定义一个相关函数来度量任意两个镜头类 SC_x 和 SC_y 的相关系数为 $Cor(SC_x, SC_y) =$

$$\begin{cases} \frac{1}{2} \times \left(\frac{InMid(SC_x, SC_y)}{Count(SC_x) - 1} + \frac{InMid(SC_y, SC_x)}{Count(SC_y) - 1} \right) & x \neq y \\ 1 & x = y \end{cases} \quad (6)$$

式中, $InMid(SC_x, SC_y)$ 函数表示在所有的两个相邻 SC_x 间隔之中, 出现的 SC_y 的个数之和;

$Count(SC_x) - 1$ 表示两个相邻 SC_x 形成的间隔的个数, 其值为 SC_x 的个数减一。

例如, 由 20 个镜头组成的镜头序列, 如图 6 所示。

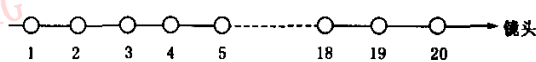


图 6 镜头序列

假设经过 2.3 节中的组合过程得到 6 个镜头类: A (包括镜头 1、2、3、4、5、8)、 B (包括镜头 6、10、11)、 C (包括镜头 7、9、12)、 D (包括镜头 13、14、15、16) 以及 E (包括镜头 17、18、19) 和 F (包括镜头 20)。对于原来的镜头序列, 如果把镜头的标识符替

换为其所在的镜头类的标识符, 则重建后的镜头序列可以表示为

AAAAABCACBBCDDDDDEEFF

对于例中的镜头序列, $InMid(B, C) = 2 + 0 = 2$, $Cor(B, C) = 1$; 而 $Cor(A, D) = 0$, $Cor(D, E) = 0$ 。表明 B 和 C 两个镜头类之间存在很强的相关性, 而 A 与 D , D 与 E 之间没有相关性。

显然, $Cor(SC_x, SC_y)$ 是对称函数, 所有镜头类之间的相关系数组成了一个对称矩阵 Q 。根据镜头类的构造算法 (时间特性) 和视频编辑的原理, 相隔较远的镜头类之间的相关系数为 0, 所以此矩阵是稀疏矩阵。

2.5 视频场景的检测

对于顺序进展的场景模式, 一个镜头类便是一个场景, 此镜头类与别的镜头类的相关系数均为 0, 如上面例子中的镜头类 D 和镜头类 E , 它们单独构成场景。

对于交错进展的模式, 场景中所属的镜头具有很强的交错性, 即这些镜头所对应的镜头类之间具有很强的相关性, 如上例中的镜头类 B 和 C , 它们同属于一个场景。

混合进展的模式, 其镜头对应的镜头类之间也有相关性, 但是可能不如交错模式的相关系数值大。如上例中的镜头类 A 、 B 和 C 。首先来自镜头 A 的镜头显示出来, A 的内容即将表达完的时候, 出现了属于镜头类 B 的镜头, 然后来自镜头类 B 、 C 的镜头交错地出现, 但是并不是说 A 不再出现。这是一种在视频中常见的场景模式, 是视频情节发展的一种常用技法。

因此, 构置场景的原则是判断相关矩阵中的元素: 如果某个镜头类与其他镜头类的相关系数均为 0, 则它单独构成一个场景, 否则将相关系数不为 0 的镜头类联系起来, 形成一个场景。

3 实验结果与分析

选取一些视频片段对视频场景检测算法进行检测。图 7 是《白雪公主与七个小矮人》开始阶段 13 个镜头的场景检测结果示例 (每个镜头用一个关键帧表示), 最终检测得到两个场景, 其中每个场景都是由两个镜头类组成。

使用改进的“双阈值”方法进行镜头边界的检测; 关键帧提取的阈值设定为 $T_k = 0.75$, 取得了比

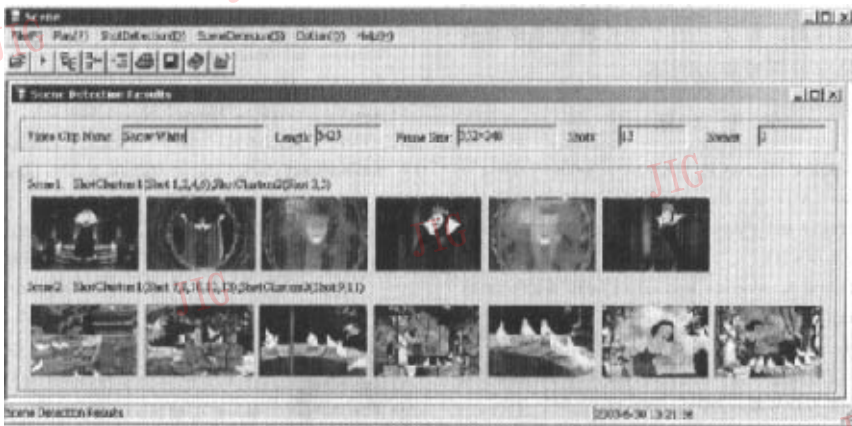


图 7 《白雪公主与七个小矮人》片段场景检测示例

较好的效果,相对于其他的一些关键帧提取算法,这种算法能够最小冗余同时又最大限度的反映镜头内容的变化。在基于滑动镜头窗的镜头类形成算法中,窗宽 L 是依赖于视频类型的,但是为了简便起见,根据统计得到的场景包含的镜头数,只需选用一个足够大的值,不至于使一些应该组合在一起的镜头被排除在窗口之外即可。相对于文献[5]中的算法来说,这种方法不受镜头实际长度的影响,避免了时间阈值设定比较困难的问题。在本文的实验中,当 $L=12$ 时,能够同时兼顾减少比较次数和满足场景构成之需要。镜头组合过程中的阈值 T_{sc} 是个经验值,通过对不同数值进行测试的结果得出, $T_{sc}=0.63$ 结果比较好。

由于场景是个语义级别的概念,不同的浏览者可能理解不同,因此很多算法的实验结果中,并没有给出错判和漏判的评判结果。但是为了说明算法的效果,这两项指标是不可或缺的,因此在本文的实验结果中给出了这两个评判结果。为了减少语义理解的主观性,实验中通过 4 个人的共同商讨来最后确定实验视频片段中的场景起始和终止边界。

表 1 给出了本文实验中 4 个比较具有代表性的视频片段的实验结果。这 4 个视频片段分别是:喜剧片《Four Weddings and A Funeral》片段(记做 FW)、枪战片《Leon》片段(LE)、科幻片《Man In Black》片段(MB)和英语教学片《Family Album USA 1》第一幕(FA),其中每个片段 5~20 分钟。表中的“+ m , - n ”表示错判数目为 m ,漏判数目为 n 。

通过分析表 1 中的数据可见,本文场景检测算

表 1 本文场景检测算法的部分实验结果

	视频片段			
	FW	LE	MB	FA
帧数	27 125	13 752	12 960	7 813
镜头数	97	110	81	49
镜头类	64	31	27	11
检测场景数	18(+3, -1)	8(+1, -0)	9(+1, -0)	4(+0, -0)
实际场景数	16	7	8	4
STG 算法 检测场景数	20(+5, -1)	8(+1, -0)	9(+1, -0)	5(+1, -0)

法的效果是有效的;另外,类似于其他一些场景检测算法,错判数目要多于漏判数目,并且漏判的情况比较少。相对于比较经典的、效果也比较好的基于时间约束的 STG 算法(表中简称 STG 算法),本文的算法能够达到等同或者优于它的效果,而本文算法的优点在于简单性和快速性,这对于长时间的大量视频数据来说,颇具意义,同时相关函数不但可以衡量镜头类之间相关性,还有利于视频的结构化,如视频中对话片段的检测(镜头类的相关系数约等于 1)。因此,本文的视频场景检测算法具有良好的扩展性。

4 结 论

提出了一种快速有效的视频场景检测算法。通过分析视频场景的发展模式给出了场景构造的原则。根据此原则,提出了基于滑动镜头窗的镜头类形成算法,定义了镜头类相关性函数,用它们来完成视频场景的检测。实验结果表明,镜头类形成算法是快速有效的,相关性函数能够反映组成视频场景的基

本单元的关联性,从而保证了算法能快速准确的实现视频场景的检测。将文中的方法扩展至其他视频结构的检测是今后要考虑的问题。

参 考 文 献

- 1 Swanberg D, Shu C F, Jain R. Knowledge guided parsing in video databases [A]. In: Proceedings of SPIE Storage and Retrieval for Image and Video Databases(1908)[C], San Jose, CA, USA, 1993:13~21.
- 2 Boykin S, Merlino A. Machine learning of event segmentation for news on demand [J]. Communications of the ACM, 2000, 43(2):35~41.
- 3 徐骏,周晓峥,于俊清等. 基于事件流的新闻视频场景分割方法[J]. 计算机辅助设计与图形学学报, 2003, 15(2):228~232.
- 4 Lu Hong, Tan Yap-peng. An unsupervised approach to dominant video scene clustering[A]. In: Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS'03) [C], Bangkok, Thailand, 2003:680~683.
- 5 Yeung M 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.
- 6 Sundaram H, Chang S F. Determining computable scenes in films and their structures using audiovisual memory models[A]. In: Proceedings of ACM Multimedia Conference [C], Los Angels, CA, USA, 2000:95~104.
- 7 Chen S C, Shyu M L, Liao W, et al. Scene change detection by audio and video clues[A]. In: Proceedings of IEEE International Conference on Multimedia and Expo (ICME2002) [C], Lausanne, Switzerland, 2002:365~368.
- 8 Boreczky J S, Wilcox L D. A hidden Markov model framework for video segmentation using audio and image features[A]. In: Proceedings of International Conference on Acoustics, Speech, and Signal Processing [C], Seattle, WA, USA, 1997: 3741~3744.

- 9 Iyengar G, Lippman A. Models for automatic classification of video sequences [A]. In: Proceedings of SPIE Storage and Retrieval for Image and Video Databases VI(3312) [C], San Jose, CA, USA, 1998:3312~3334.
- 10 Rui Y, Thomas S, Huang, Sharad Mehrotra. Exploring video structure beyond the shots [A]. In: Proceedings of IEEE International Conference on Multimedia Computing and Systems [C], Austin, Texas, USA, 1998:237~240.
- 11 Sánchez J M, Binefa X, Vitrià J, et al. Linking visual cues and semantic terms under specific digital video domains[J]. Journal of Visual Languages and Computing, 2000, 11(3): 253~271.

程文刚 1977年生。1998年于山东科技大学获学士学位, 2001年于中国矿业大学(北京校区)获硕士学位, 目前在北京交通大学计算机应用专业攻读博士学位。主要研究方向为多媒体信息处理、基于内容的视频分析和机器学习。



须德 1944年生。教授、博士生导师。1967年毕业于中国科技大学应用数学系, 1982于北方交通大学获计算机应用专业硕士学位。主要研究方向为基于内容的视频检索、多媒体信息处理和数据库系统。



郎丛妍 1978年生。2001年于内蒙古大学获信息系统专业学士学位, 2001年开始在北方交通大学攻读硕士学位, 2003提前攻博, 现为北京交通大学计算机软件与理论专业博士研究生。主要研究方向为模式识别和图像处理。

