

基于 HSV 颜色空间的视频镜头检测

肖治民¹, 林坤辉^{1*}, 周昌乐², 林 强¹

(1. 厦门大学软件学院, 2. 厦门大学信息科学与技术学院, 福建 厦门 361005)

摘要: 为了克服光线变化和运动对视频镜头检测的影响, 本文提出了一种新的视频镜头检测算法. 该算法把视频图像从 RGB 颜色空间映射到 HSV 颜色空间, 再重新量化图像的 H 与 S 分量到 127 个色度区域, 最后通过比较帧间图像在色度区域的重叠程度来检测镜头的变化. 同时采取有效的虚假镜头分割分析机制以减少图像快速变化的影响. 实验结果表明, 本算法具有较高的准确率和查全率, 并且有较快的处理速度, 在视频镜头分割检测领域具有一定的实用价值.

关键词: HSV 颜色空间; 镜头分割; 镜头检测算法

中图分类号: TP 301

文献标识码: A

文章编号: 0438-0479(2008)05-0665-04

随着计算机视频的日益丰富, 如何对海量的视频数据进行高效的计算机处理, 实现视频内容的索引、查找、分析、语义描述和压缩等成为现在关注的焦点. 而目前一致公认的处理手段是视频内容结构化.

视频内容结构化就是对视频流中的连续帧序列进行切分, 把一个连续视频流按其内容展开的不同, 将它分成若干语义段落单元^[1]. 视频流的结构化能够从一部很长的视频中抽象出视频内部隐含的情节发展结构, 它为大数据量视频的导航和浏览提供了一种非常好的手段. 作为视频结构化处理的基本步骤——镜头检测, 它的工作性能与效果将直接影响后续的处理.

目前镜头检测方法朝着两个方向进行. 一是在压缩域内进行检测, 它不需要解码还原视频信号, 而是直接读取视频文件的压缩编码信息, 因而能大大降低计算复杂度. ARMAN^[2]等人提出的用关键帧(I 帧)的 DCT 系数来探测场景变化的技术. Zhang^[3]等提出了一种在 I 帧内逐帧比较技术, 对两帧中对应的 DCT 系数进行匹配, 这种方法类似于像素域中的像素亮度匹配技术. 还有一种是在非压缩域内进行检测, 它需要先还原压缩的视频信息为一般帧图像, 然后再利用普通图像处理技术提取其中的图像特征. 其代表算法有像素对比较方法、模板比较方法、直方图比较方法等^[4]. 总的来讲, 基于压缩域内检测算法比非压缩域内检测算法的速度快, 但这些方法仅限 MPEG 格式的视频(也可推广到 H. 26x 格式), 由于视频格式的多样性,

因此其运用范围也受到了较大的限制; 而目前在非压缩域内进行检测算法的缺点是处理速度慢, 误检率较高.

本文将提出一种非压缩域的镜头检测算法, 该算法将图像从 RGB 颜色空间转换到 HSV 颜色空间, 然后提取图像的色调 H 和饱和度 S 特征, 再对相邻帧之间的特征进行相似度计算来检测镜头切换.

1 HSV 颜色模型介绍

颜色模型是一种表示特定颜色的标准方法, 它通常由一个三维坐标系统及其颜色子空间组成. HSV 颜色模型是由色调 H 、饱和度 S 和亮度 V 的三维坐标系构成的三维颜色空间^[5-6].

文献^[7]指出, 在图像中色调 H 不容易受到亮度变化的影响, 而人眼对于亮度的变化比较敏感. 在本文算法中, 为了避免同一镜头内亮度变化的影响, 因而只选择图像的 H 分量与 S 分量作为检测依据.

2 算法介绍

2.1 主要流程

本文算法的主要流程如图 1 所示. 其主要功能模块包括量化、相似度计算和有效性检验, 而其关键数据类型为特征向量 HS . 这些将会在后述部分详细介绍. 图 1 中 HS_s 和 HS_c 分别表示上一帧图像的特征和当前帧图像的特征.

2.2 主要数据与模块

2.2.1 特征向量

特征向量 $HS = \{a_0, a_1, \dots, a_{126}\}$ 是一个 127 维的向量. 根据本文算法需要, 将 HSV 颜色空间的 H 与 S

收稿日期: 2008-03-26

基金项目: 国家 863 计划项目(2006AA01Z129), 国家自然科学基金(60672018), 厦门大学“985”二期信息创新平台项目(0000-X07204)资助

*通讯作者: khlin@xmu.edu.cn

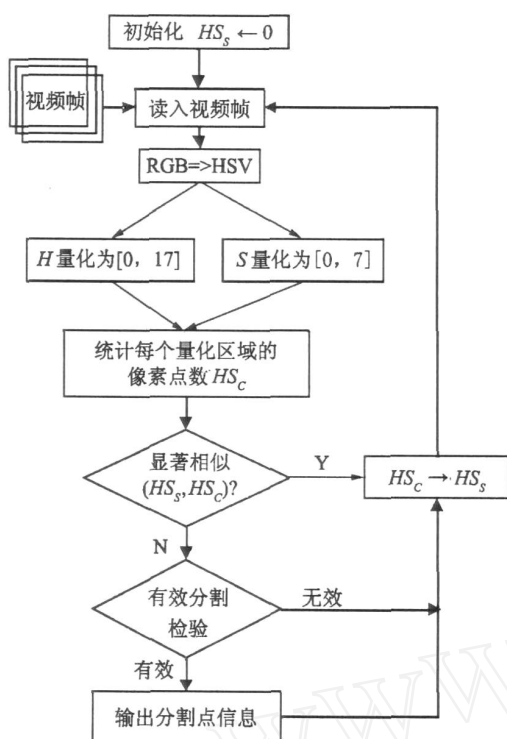


图1 主要算法流程

Fig.1 Flow chart of detecting shot

分量组成的色轮进行区域划分,然后给每个区域指定一个编号. H 分量按每 20° 一级共分为 18 个扇形区域 $\{S_0, S_1, \dots, S_{17}\}$, S 分量按每 12.5% 一级共分为 8 个同心圆区域 $\{C_0, C_1, \dots, C_7\}$, 由此交叉可产生 144 个区域. 由于 C_0 区域内颜色都非常接近, 人眼不能区分, 因而将整个 C_0 区域作为一个整体, 不做色调划分, 并将此部分区域的编号定为 0. C_1 与 S_0 的重叠区域的编号为 1, C_1 与 S_1 的重叠区域的编号为 2, ..., C_2 与 S_0 的重叠区域的编号为 19, 以此类推则 C_7 与 S_{17} 的重叠区域的编号为 126. 其计算公式为

$$\begin{cases} 0, S \text{ 分量区域号} = 0 \\ S \text{ 分量区域号} \times 18 + H \text{ 分量区域号} - 17, S \text{ 分量区域号} > 0 \end{cases} \quad (1)$$

特征向量中每个元素的值 a_n 表示图像中有 a_n 个像素点的 H 与 S 分量处于 n 区域. 图 1 中特征向量 HS_s 与 HS_c 分别表示相邻两帧图像 F_{n-1} 和 F_n 的特征值.

2.2.2 量化

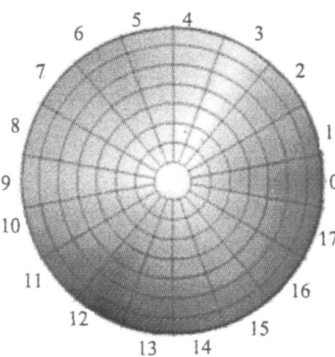
量化过程是对图像中的 H 分量与 S 分量进行区域指定, 以提高整个算法的处理效率.

H 分量的量化按以下公式

$$H = (h \bmod 350 + 10) \div 20, h \in [0, 360] \quad (2)$$

$$H \in [0, 17]$$

式中 h 表示量化前的色调值, H 表示量化后的值.

图2 $V = 100\%$ 时色轮区域Fig.2 Slice of color wheel while $V = 100\%$

S 分量的量化按以下公式

$$S = (s - \varepsilon) \div 0.125, s \in [0, 1], S \in [0, 7] \quad (3)$$

式中 s 表示量化前的饱和度值, S 表示量化后的值.

在式(2), (3)中, $n \bmod m$ 表示取 n 除以 m 的余数, $n \div m$ 表示取 n 除以 m 结果的整数部分, ε 表示机器所支持的最小浮点数.

通过量化, H 与 S 分量便可与特征向量 HS 方便地进行映射. 假设某像素点的 $h = 200, s = 0.5$. 经过公式(2), (3)计算可得 $H = 10, S = 3$. 再经公式(1)计算可得 $N = 47$. 因此该点特征应计入向量元素 a_{47} .

2.2.3 特征相似性计算

算法分别使用特征向量 HS_s 与 HS_c 保存相邻两帧图像 F_{n-1} 和 F_n 的特征值, 则相邻两帧图像的相似性 $S(F_{n-1}, F_n)$ 计算方法如下

$$S(F_{n-1}, F_n) = \frac{1}{2L \cdot h} \sum_{i=1}^{126} |HS_c(a_i) - HS_s(a_i)| \quad (4)$$

式中 L 和 h 分别表示以像素为单位的视频图像宽度和高度, $HS_c(a_i)$ 表示向量 HS_c 编号为 i 的元素. $S(F_{n-1}, F_n)$ 的取值范围为 $[0, 1]$.

为了达到更好的检测效果, 本文算法需要引入两个阈值变量 V_1, V_2 且 $0 < V_1 < V_2 < 1$. 当 $S(F_{n-1}, F_n) < V_1$ 时, 则认为显著相似, F_{n-1}, F_n 在同一镜头内; 当 $S(F_{n-1}, F_n) > V_2$ 时, 则认为显著不相似, F_{n-1}, F_n 不在同一镜头内; 当 $V_1 < S(F_{n-1}, F_n) < V_2$ 时, 则需要引入有效分割检验过程, 以尽可能减少漏检与误检的发生.

2.2.4 有效分割检验

当 $V_1 < S(F_{n-1}, F_n) < V_2$ 时, F_{n-1} 与 F_n 是否属于同一镜头, 则需要运用其它手段来检验. 为此本文提供了一种检验手段, 并引入链 list. list 中的结点类型为一帧图像的特征向量. 假设有帧序列 $(\dots, F_{m-1}, F_m,$

表 1 视频综合检测实验结果
Tab.1 Results of integrated experiments

序号	视频题材	实际镜头数	正确检测数	误检数	漏检数	准确率/ %	查全率/ %
1	战斗	120	108	12	5	90	90
2	动作	100	89	10	3	89.9	89
3	生活	120	106	12	10	89.8	88.3
4	历史题材剧	80	72	10	2	87.8	90
5	古装战斗	60	53	5	5	91.4	88.3
6	感情剧	60	55	3	4	94.8	91.7
合计		540	483	52	29	90.3	89.4

$F_{m+1}, \dots, F_{n-1}, F_n$). 其中 F_{m-1} 与 F_m 属于不同镜头内的两帧,而帧序列 $F_m, F_{m+1}, \dots, F_{n-1}$ 属于同一镜头内. list 中保存了从帧序列 $F_m, F_{m+1}, \dots, F_{n-1}$ 中 $F_m, F_{m+k}, F_{m+2k}, \dots, F_{m+ck}$ 的特征向量(k 为采样距离,本文实验中 $k=12$). 当 $V_1 < S(F_{n-1}, F_n) < V_2$ 时,按如下算法进行检验:

- step1> 检查 list 是否为空. 如果空则判 F_{n-1} 与 F_n 属同一镜头. 非空则转 step2;
- step2> 从尾至头遍历 list,并用向量 HS_x 表示 list 中当前位置的结点值;
- step3> 运用公式(4)计算 HS_x 与 HS_c 的相似度 S_{xc} ;
- step4> 检查 $S_{xc} < V_1$ 是否成立. 如果成立则判 F_{n-1} 与 F_n 属同一镜头并转至 step7;
- step5> 检查 list 是否访问完,如果全部已访问,则判 F_{n-1} 与 F_n 属不同镜头并转至 step7;
- step6> 更新 HS_x 为 list 中当前结点的前驱结点值并转至 step3;
- step7> 结束.

3 实验分析

3.1 实验数据集

由于没有视频镜头检测方面专用的测试视频集,本文实验采用的测试集是从影视片中截取的片段组成的. 测试集共由 6 部影片片段组成(“空军一号”、“国家宝藏”、“Forrest Gump”、“三国演义”、“投名状”、“Healing Hand”),共计 540 个镜头. 片段类型包括快速运动、日常生活、打斗、爆炸等场面. 镜头变换类型包括有切变、淡入淡出、叠化和滑入等几种主要方式. 因此该测试集具有一定的代表性. 测试集中视频内容均被转换为 320 ×240 像素的标准尺寸.

3.2 评价标准

实验过程中对镜头分割检测采用了通用的评价标

准:准确率和查全率^[8].
准确率 = 正确检测数/ (正确检测数 + 误检数);
查全率 = 正确检测数/ (正确检测数 + 漏检数).

3.3 软硬件环境

硬件环境:HP Compaq 6515b 笔记本,AMD Turion64 1.8 G ×2 CPU,1.00 GB 内存;
操作系统:Microsoft Windows XP;
编程语言:Microsoft Visual C++ 6.0,OpenCV beta 5.

3.4 实验结果及分析

根据选定的实验环境,设阈值 $V_1 = 0.15, V_2 = 0.5$. 本文进行了两组实验. 实验 1 主要是检测算法对视频整体性能的检测,其结果见表 1.
实验 2 主要是检测算法对各种镜头转换形式的检测性能,其结果见表 2.

表 2 镜头切换形式分类检测实验结果
Tab.2 Results of classed shot-changing experiments

序号	类型	实际镜头数	正确检测数	准确率/ %
1	切变	50	48	96.0
2	淡入淡出	30	23	76.7
3	叠化	30	26	86.7
4	滑入	30	25	83.3
5	其它	30	21	70.0

由实验结果可以看出,本文算法表现出了较好实验性能. 不仅对视频的整体检测有比较好准确率和查全率,而且对多种镜头切变形式表现出了较高的正确率. 从表 1 可以看出,本文算法对于现代感情剧检测效果最好,而对战斗片和动作片效果稍差. 这主要是因为感情剧的画面内容变化比较缓,而战斗片和动作片则较快,特别是人物较多的混战场面. 同一镜头内的连续两帧变化越大,引起误检的可能性就越大. 从表 2 可以看出,本文算法对于切变检测率最高,而对淡入淡出较

差. 根据观察, 视频镜头切换的最主要的形式还是切变. 因此本文算法具有一定的应用价值. 结合所有实验数据, 可见相邻帧之间的变化快慢对本文算法有较大的影响.

4 结束语

本文在非压缩视频镜头检测分割方面提出一种基于 HSV 颜色空间的检测算法. 该算法能忽略图像亮度变化和运行带来的影响. 同时算法采取了能进行虚假镜头分割的分析机制, 最大限度地降低误检率和查全率. 但该算法受同一镜头内相邻帧的快速变化的影响. 如何克服这种快速变化的影响, 降低误检率将是下一步要进行的工作.

参考文献:

- [1] 庄越挺, 潘云鹤, 吴飞. 网上多媒体信息分析与检索[M]. 北京: 清华大学出版社, 2002.
- [2] Arman F, Hsu A, Chiu M Y. Image processing on compressed data for large video databases[C]//Proceedings of First ACM International Conference on Multimedia. Anaheim: ACM, 1993: 267 - 272.
- [3] Zhang H J, Low C Y, Gong Y H, et al. Video parsing using compressed data[C]//Proceedings of SPIE Conference on Image and Video Processing II. San Jose, CA, USA: SPIE Digital Library, 1994: 142 - 149.
- [4] 钱刚, 曾贵华. 典型视频镜头分割方法的比较[J]. 计算机工程与应用, 2004(32): 51 - 55.
- [5] 丁海波, 薛质, 李生红. 基于 HSI 空间的肤色检测方法[J]. 计算机应用, 2004, 24(B12): 210 - 211.
- [6] Domik G, Schauble C J C, Fosdick L D, et al. Tutorial: color in scientific visualization[OE/OL]. 1999. <http://www-ugrad.cs.colorado.edu/~csci4576/SciVis/SciVisColor.html>.
- [7] Alasdair M. Introduction to digital image processing with MATLAB[M]. 重庆: 重庆大学出版社, 2007.
- [8] 方勇, 戚飞虎. 一种新的视频镜头边界检测及关键帧提取方法[J]. 华南理工大学学报: 自然科学版, 2004, 32(Z1): 18 - 23.

Shot Segmentation Based on HSV Color Model

XIAO Zhi-min¹, LIN Kun-hui^{1*}, ZHOU Chang-le², LIN Qiang¹

(1. Software School, Xiamen University, 2. School of Information Science and Technology, Xiamen University, Xiamen 361005, China)

Abstract: To overcome the interference produced by illumination changing and movement during shot detection, this paper proposed a new method for shot segmentation. It firstly transforms the video pictures from RGB space to HSV space, then requantitates the *H* and *S* components of pictures to 127 chroma areas, finally detects the shot's change by computing the coincident degree in chroma areas between frames. In addition, it adopts a false-shot-analysis strategy to weaken the interference brought by fast change of pictures. The experiment results indicated that the algorithm had high accuracy and recall ratio, as well as fast processing speed. So it is suitable for shot segmentation.

Key words: HSV color model; shot segmentation; shot boundary detection algorithm