

适用于 IPTV 的视频拷贝检测系统

潘 岩¹⁾ 周 钧¹⁾ 董道国²⁾ 金 城¹⁾

¹⁾(复旦大学计算机科学技术学院,上海 200433) ²⁾(上海未来宽带技术及应用工程研究中心,上海 200336)

摘 要 近年来,基于内容的拷贝检测技术伴随着 Internet 和无线网络的快速发展日渐引起人们的重视。针对 IPTV 系统应用的特殊性,提出了一种快速高效的视频拷贝检测系统。该系统利用简洁有效的帧相似度处理方法,对视频进行镜头提取并进行聚类得到镜头簇,最后利用 K-means 树构建所有视频的镜头簇索引,从而实现快速的拷贝检测。实验结果表明,该系统不仅检测速度快,而且针对 IPTV 应用具有良好的检测准确性和覆盖率。

关键词 视频 拷贝检测 网络电视(IPTV)

中图法分类号: TP391.3 文献标识码: A 文章编号: 1006-8961(2009)11-2421-04

Video Copy Detection System Applying to IPTV

PAN Yan¹⁾, ZHOU Jun¹⁾, DONG Dao-guo²⁾, JIN Cheng¹⁾

¹⁾(School of Computer Science, Fudan University, Shanghai 200433)

²⁾(Shanghai Engineering Research Center for Broadband Technologies & Application, Shanghai 200336)

Abstract In recent years, the method of content based copy detection (CBCD) is increasingly attracting our attention as internet and wireless network develops quickly. Our paper presents a new fast and efficient system for video copy detection aiming at the particularity of IPTV. The system takes a laconic method to compare the similarity of video frames, clusters the shots into shot-clusters after selecting them from videos and makes up a tree of shot-clusters named K-means tree. The experiment results show that our system has a satisfying searching speed, as well as good rate of covering and precision.

Keywords video, copy detection, internet protocol television(IPTV)

1 引 言

随着 Internet 及无线网络的快速发展,海量的数字视频信息开始进入人们的生活。随之而来的是越来越严重的数字冗余和数字版权问题,近几年来,基于内容的拷贝检测(CBCD)技术的出现提供了一个可以代替数字水印技术的识别视频和图像拷贝的方法。

已经出现的拷贝检测技术多种多样,包括应用了全局特征的 Ordinal Measurement^[1]技术和它的带时间窗版本 Temporal Ordinal Measurement^[2],以及

使用局部特征的 AJ^[3], ViCopT^[4] 和 STIP^[5] 等。这些方法在处理不同的数据拷贝类型时具有它们各自的优点和不足。在现代数字电视应用,如 IPTV (Internet protocol TV) 系统中,视频拷贝问题具有其独有的特点。

虽然在 IPTV 应用中,可能出现所有的视频拷贝类型,但统计数据表明,实际出现比重最高的几种数据拷贝类型^[6]分别是:尺寸缩放、亮度变换、视频编码变化引起的清晰度变化、人工添加图文。其中尺寸缩放变换通常是将原始图像缩放后嵌入到新的图像中去,产生所谓的“画中画”拷贝类型。另外,在 IPTV 应用中,每天都要处理大量的新数据,因此

基金项目:国家科技支撑计划项目(2007BAH09B03);上海市科委项目(08dz1500109)

收稿日期:2009-06-19;改回日期:2009-08-28

第一作者简介:潘 岩(1988 ~),男。复旦大学计算机科学技术学院计算机应用技术专业硕士研究生。主要研究方向为计算机图形图像处理。E-mail:082024075@fudan.edu.cn

要求检测技术应该具有很高的检测效率。在已有的检测方法中,使用全局特征的检测方法计算效率高,能够较好地处理清晰度变化和人工添加图文问题,在适当的处理之后也可以检测亮度变换,但它对“画中画”问题往往无能为力;而使用局部特征的检测技术虽然可以处理“画中画”问题,对亮度变换和

人工添加图文也具有良好的适应性,但它计算复杂性高,检测结果易受视频清晰度变化的影响而不稳定,也难以适应 IPTV 应用的要求。针对 IPTV 应用的特点,提出一套快速而有效的检测方法,实验结果证明,本系统确实能够快速有效地解决 IPTV 应用中的视频拷贝检测问题。



图 1 电视应用中的视频拷贝类型

Fig. 1 Video copy types of TV application

2 系统框架

2.1 帧相似度计算与“画中画”检测

传统的检测“画中画”的方法是利用局部特征检测出被嵌入的图像部分,但这类方法计算复杂性高,易受视频编码的影响,实际结果不稳定。为了更好地解决“画中画”问题,先通过特定的“画中画”检测程序找出被嵌入的视频块,再将被嵌入的视频块与背景图像分离,对两者分别进行拷贝检测。

帧相似度计算使用基于分块直方图的方法。使用分块直方图计算帧相似度的方法对尺寸缩放变换和因编码造成的清晰度变化具有良好的鲁棒性;由于人工添加的图文通常不会覆盖原始图像的主要部分,因此分块直方图方法也可以处理这种类型的问题;通过将帧图像进行灰度化和直方图均衡化,可以有效地降低亮度变化造成的影响。因此分块直方图方法是一种可以良好适用于 IPTV 系统的帧相似度计算方法。

对得到的帧图像,在进行灰度化和直方图均衡化之后,将其划分为 $N \times N$ 的 N^2 个大小均等的子图

(本系统中 N 取 8),对每个子图计算它的 16 阶灰度直方图,再将直方图归一化,这样每帧图像就可以表示成 N^2 个 16 维归一化向量。如下形式:

$$I_i = \{ \mathbf{v}_i^{(1)}, \mathbf{v}_i^{(2)}, \mathbf{v}_i^{(3)}, \dots, \mathbf{v}_i^{(N \times N)} \} \quad (1)$$

$$\mathbf{v}^{(k)} = \langle v_1^{(k)}, v_2^{(k)}, v_3^{(k)}, \dots, v_{16}^{(k)} \rangle \quad (2)$$

由此可将图像相似度的计算转化为两个高维向量相似度的计算。为了更好地反映两幅图片的拷贝程度,使用下面的方法计算 $\mathbf{v}_i^{(k)}, \mathbf{v}_j^{(k)}$ 两个向量之间的相似度。

$$s_{i,j}^k = \frac{\mathbf{v}_i^{(k)} \cdot \mathbf{v}_j^{(k)}}{\|\mathbf{v}_i^{(k)}\| \times \|\mathbf{v}_j^{(k)}\|} \quad (3)$$

$$s_{i,j} = \frac{\sum_{k=1}^{N \times N} (s_{i,j}^k)^2}{N^2} \quad (4)$$

“画中画”检测的基本思想是寻找那些持续稳定出现,且具有“画中画”特点的帧内图像边缘。在实际电视应用中,“画中画”的出现通常具有如下特点:持续一定时间,位置稳定,轮廓多为矩形。为检测出这些特征,先对目标视频帧进行边缘检测,然后对视频进行加窗处理:在长度为 L 帧的窗体内,对出现频率超过 $L \times r$ 的 (r 为阈值因子) 边缘记为候选

“画中画”边缘,并检测候选边缘的形状特征和边缘内部的内容变化幅度,如果候选边缘内部的边缘内容变化幅度过小,则可认为此边缘是原视频中出现的正常框状部分,排除是“画中画”的可能。

2.2 镜头提取和聚类

2.2.1 镜头划分和代表帧提取

对视频进行镜头划分,通过使用上面的帧相似度计算方法寻找镜头边界。为了避免长镜头造成的累积帧差影响检测效果,应在控制帧间相似度的同时设置一个累积帧差阈值。注意到在实际视频中,镜头与镜头之间常常使用渐变帧作为过渡,这会造成检测结果中出现帧数过少的虚假镜头,因此可以抛弃掉那些帧数过少的镜头。另外为了提高检测的效率,可以使用跳帧检测的方法,即每隔数帧比较一次相似性,但这么做也可能会降低以后进行聚类 and 搜索时的精确性。

在得到一个镜头之后,需要选取此镜头的代表帧。虽然可以简单地选取靠近镜头中间部分的帧来作为代表帧,但为了进一步提高检测的覆盖率,计算所有帧的一个加权平均来作为镜头的代表帧。

$$S_i = \{I_1, I_2, I_3, \dots, I_K\} \quad (5)$$

$$I'_i = \frac{\sum_{n=1}^K (\omega_n * I_n)}{\sum_{n=1}^K \omega_n} \quad (6)$$

ω_n 可依据具体数据类型取不同的取值分布,本系统中的 ω_n 依据正态分布 $\sigma\left(\frac{K}{2}, \frac{1}{K}\right)$ 来获得。

2.2.2 镜头聚类

在将一个视频划分为众多镜头之后,对所有镜头进行聚类操作得到镜头簇。

利用镜头代表帧之间的相似度来代表镜头之间的相似度,并以此为依据进行聚类操作。即

$$SS_{i,j} = sim(I'_i, I'_j) \quad (7)$$

式中, I'_i 与 I'_j 分别为镜头簇 i, j 的代表帧。

2.2.3 构建类索引树

为了快速地从大量视频数据库找出存在拷贝数据的视频,将数据库中所有视频包含的镜头类进行量子化操作。构建一棵以相似度衡量为基础的索引树(K-means tree),索引树使用反转文件记录每个类所属的视频^[7]。

该方法通过一个分级递归的 K-means 聚类算法构建一棵索引树。在这棵树中,“血缘关系”越是亲

密的镜头簇的相似性就越高。通过构建索引树,可以比较两个镜头簇在索引树中的相对位置来代替直接计算两个镜头簇之间的相似性。在对新视频的镜头簇进行拷贝检测时,由于在对索引树进行搜索时每次只需要比较新簇和同一结点下的 K 子女的相似性,搜索的复杂性降为 $\Theta(\log(M))$, M 为数据库中所有镜头簇的个数。

在计算镜头簇的相似性时,使用如下方法:

每个镜头簇可以表示为

$$SC_i = \{S_i^{(1)}, S_i^{(2)}, S_i^{(3)}, \dots, S_i^{(M)}\} \quad (8)$$

两个镜头簇的相似度为

$$SCS_{i,j} = \max_{0 < m < N, 0 < n < N} (SS_{i,j}^{(m,n)}) \quad (9)$$

式中, M, N 分别为镜头簇 i, j 包含的镜头数, $SS_{i,j}^{(m,n)}$ 表示镜头簇 i 中的第 m 个镜头与镜头簇 j 中第 n 个镜头的相似度。

2.3 新视频的拷贝检测

在处理一个新视频时,首先对其进行镜头划分和聚类,对得到的每个镜头簇在数据库上已经构建好的索引树中查找拷贝目标。每个镜头簇找到的拷贝目标所属的视频就是存在拷贝数据的视频。系统的检测流程如图 2 所示。

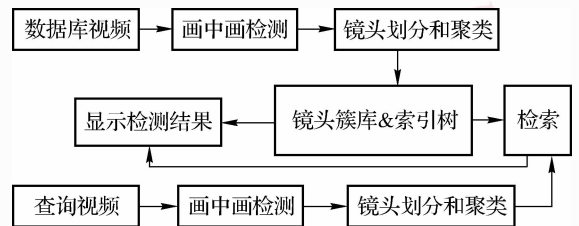


图 2 视频拷贝检测系统流程

Fig. 2 Flow of video copy detection system

3 实验结果及分析

本实验通过模拟现实电视应用中出现的视频拷贝类型,在由 3 320 个来自不同类型视频的镜头簇组成的数据库中进行搜索测试。实验在硬件配置为 Pentium 4 CPU 3.00 GHz, 1G 内存的 PC 机上进行。

为了全面地反映本系统的检测效果,使用如下标准衡量搜索结果:

检测覆盖率

$$recall = \frac{N_{copy}}{N_{allcopy}}$$

检测准确率

$$precision = \frac{N_{copy}}{N_{alldetected}}$$

式中, $N_{alldetected}$ 为系统检测结果总数, $N_{allcopy}$ 为数据库实际存在的拷贝总数, N_{copy} 为系统检测结果中正确的结果数目, 三者单位均为镜头数。

表 1 为本系统对于不同类型数据的测试结果。

表 1 不同拷贝类型数据的测试结果

Tab. 1 Results of different types of copy data

类型	N_{copy}	$N_{alldetected}$	$N_{allcopy}$	recall (%)	precision (%)
亮度增强 25%	172	193	200	86	89.1
亮度减弱 25%	178	195	200	89	91.2
画中画	148	157	200	74	94.2
模糊	185	202	200	92.5	91.6
添加字幕	176	184	200	88	95.6
合计	859	931	1 000	85.9	92.3

图 3 是单次搜索花费的平均搜索时间随数据库内镜头数的增长曲线。

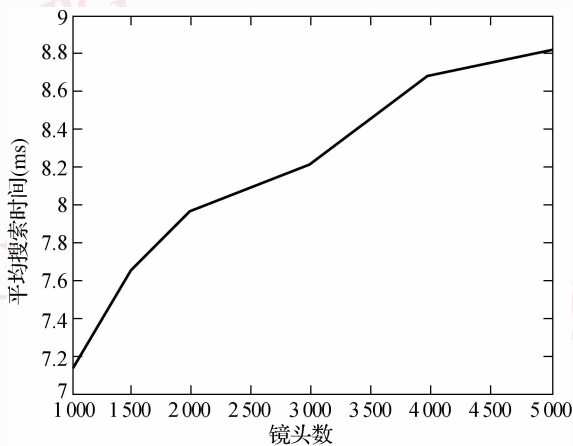


图 3 单次搜索的平均时间随数据库规模的增长曲线

Fig. 3 Curve of search once time as data base scale increasing

从表 1 中可以看出,该系统对于拷贝视频的亮度变换,清晰度变化和人工添加字幕都有良好的检测效果,而对“画中画”拷贝检测的覆盖率不高,但这一点更多的是由于“画中画”本身检测的不足造

成的。总体来看,系统无论是准确性还是覆盖率都有不错的表现。

在检测速度方面,由于使用了基于索引树的检索方法,系统可以在数据库规模的对数时间内完成检索过程。从图 3、表 1 中可以看到,单次搜索的平均时间随着数据库镜头数量的增长基本成对数模式增长,表明该系统的检测时间效率的确很高。

4 结 论

提出了一种适用于 IPTV 系统的视频拷贝检测系统,系统针对 IPTV 应用中处理的特殊问题类型,使用了一种简洁高效的帧相似度比较方法,并通过聚类产生镜头簇和搭建 K-means 索引树形成了一个快速的拷贝检测流程。实验结果表明,该系统确实具有高效性和准确性。

参考文献 (References)

- Hampapur A, Bolle R. Comparison of sequence matching techniques for video copy detection [A]. In: Proceedings of Conference on Storage and Retrieval for Media Databases 2002 [C], San Jose, California, USA, 2002:194-201.
- Chen L, Stentiford F. Video Sequence Matching Based on Temporal Ordinal Measurement [R], UCL Adastral, TR001, London, U. K. US: University College London, 2006.
- Joly A, Buisson O, Frelicot C. Content-based copy retrieval using distortion-based probabilistic similarity search [J]. IEEE Transactions on Multimedia, 2007, 9(2): 293-306.
- Law-to J, Buisson O, Gouet-brunet V, et al. Voting algorithm based on labels of behavior for video copy detection [A]. In: Proceedings of ACM Multimedia, 2006 [C], Santa Barbara, CA, USA, 2006: 835-844.
- Laptev I, Lindeberg T. Space-time interest points [A]. In: Proceedings of International Conference on Computer Vision, 2003 [C], Nice, France, 2003: 432-439.
- Law-to J, Chen L, Joly A. Video copy detection: A comparative study [A]. In: Proceedings of Conference on Image and Video Retrieval, 2007 [C], Amsterdam, The Netherlands, 2007: 371-378.
- Nister D, Stewenius H. Scalable recognition with a vocabulary tree [A]. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2006 [C], New York, USA, 2006:2161-2168.