

基于 EM 聚类的 H.264 压缩域 视频对象实时分割算法

刘方青¹⁾ 石旭利^{1),2)} 张兆扬^{1),2)}

¹⁾(新型显示技术及应用集成教育部重点实验室, 上海 200072)

²⁾(上海大学通信与信息工程学院, 上海 200072)

摘要 从压缩域直接分割视频对象比传统的像素域分割具有快速高效的特点,目前已有不少从 MPEG 域分割的方法,但从 H.264 压缩域分割的甚少。为此提出一种基于 H.264 域的实时分割运动对象方法,该算法先对当前视频帧进行全局运动估计和补偿,然后对 4×4 的运动矢量场进行分类处理,最后对非零运动矢量使用改进的 EM 聚类分割算法。本文算法对多个视频序列进行了实验,结果表明,该算法针对静止背景和运动背景的视频序列都能达到较精确的实时分割。

关键词 H.264 压缩域 运动对象分割 EM 聚类 全局运动补偿

中图分类号: TN941.1 **文献标识码**: A **文章编号**: 1006-8961(2007)10-1819-04

Real-time Video Object Segmentation in the Compressed Domain Based on EM Clustering

LIU Fang-qing¹⁾, SHI Xu-li^{1),2)}, ZHANG Zhao-yang^{1),2)}

¹⁾(Key Laboratory of Advanced Display and System Application, Ministry of Education, Shanghai 200072)

²⁾(School of Communication and Information Engineering, Shanghai University, Shanghai 200072)

Abstract Moving object segmentation in the compressed domain is fast and efficient compared to former segmentation in pixel domain. Recently some moving object segmentation algorithm in MPEG compressed domain have been reported. However, very few approaches have been proposed in the H.264 compressed domain. In this paper, a real-time video segmentation method in H.264 compressed domain is proposed. Firstly, the current frame is globally compensated in motion after the global motion estimation. Then the 4×4 motion vectors are processed by median filter and classification. Thirdly the non-zero motion vectors are applied by improved EM clustering segmentation algorithm. The performance of the proposed approach is illustrated by simulation carried on several test sequences. Experiment results show that the proposed algorithm can provide satisfactory real-time segmentation results both on static background and moving background video.

Keywords H.264, compressed domain, moving object segmentation, EM clustering, global motion compensation

1 引言

运动对象分割已日益普遍应用于视频检索、视频监控等许多实时视频处理中。传统的像素域分割运动对象是基于每个像素的运动信息,虽然较精确,

但计算量极大,难以达到实时应用的要求。压缩域的运动对象分割算法不用对码流进行完全解码,计算量大大减少,而且从压缩视频中直接提取出的运动矢量和 DCT 系数,可直接作为分割所需的运动特征和纹理特征,因此压缩域对象分割更适合于具有实时性要求的应用场合。

基金项目:国家自然科学基金项目(60572127);上海市重点学科项目(T0102)

收稿日期:2007-07-05;改回日期:2007-07-25

第一作者简介:刘方青(1984 -),男。上海大学通信学院信息与信号处理专业硕士研究生。研究方向为数字视频处理。E-mail: 20019004@163.com

近年来在压缩域内的运动对象分割已取得了一些结果,但绝大部分都停留在 MPEG 压缩域^[1-3]。H.264 是最新的视频编码标准,但 H.264 域的分割研究甚少。相比传统的基于 MPEG 域,有一些新的特性^[4],比如帧内预测让 I 帧的 DCT 系数不能直接用作分割的纹理特征。因此,在 H.264 域进行运动对象分割可直接使用的特征只有运动矢量信息。目前在 H.264 域,只有 Zeng 等人^[5]和 Liu 等人^[6]提出直接从稀疏运动矢量场中分割运动对象的方法,但文献[5]的方法只适用于静态背景的视频序列,而文献[6]的方法较复杂,计算量较大。

本文提出一种 H.264 域的运动对象分割算法,它不受限于静止背景的视频序列,对于运动背景也能快速可靠地分割出运动对象。

2 基于 EM 的 H.264 压缩域的视频对象分割方法

对 H.264 域直接提取的 4×4 运动矢量场进行中值滤波,得到较平滑鲁棒的运动矢量场;接着对该运动矢量场进行全局运动补偿以获得各块的残差;然后采用 EM 算法将运动矢量场分割成多个具有相似运动的区域。系统框图如图 1 所示。

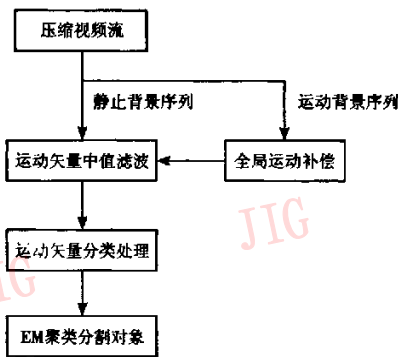


图 1 本文算法流程图

Fig. 1 Flowchart of the proposed algorithm

2.1 运动矢量的滤波和分类处理

由于 H.264 支持可变块尺寸的运动估计,一个宏块可能被分成多个块并拥有多个运动矢量。因此 H.264 视频能直接提取得到可变块尺寸的运动矢量场,这和 MPEG 域具有固定 8×8 块尺寸的均匀运动矢量场不同。原始运动矢量场的最小块尺寸为 4×4 ,将尺寸大于 4×4 的块的运动矢量直接赋给该块所覆

盖的所有 4×4 块,这样就得到了 4×4 块的运动矢量场。然后对其进行 3×3 中值滤波以消除噪声运动矢量,同时平滑相邻块之间运动矢量的差异。

为降低聚类算法的计算量,用分类处理使运动矢量场更加规则。分类后的非零运动块根据 EM 算法被分割到不同的层中。定义 3 种 MV 类型:(1)背景 MV(BMV), (2)前景 MV(FMV), (3)噪声 MV(NMV)。分类后其中只有 FMV 块通过 EM 算法被分配到不同的层。如果 MV 值小于预置的阈值 T_b ,则被定义为 BMV;如果 MV 值大于阈值 T_f ,则定义为 FMV,FMV 的块表示它们有真实的 MV。如果 MV 值在 T_b 和 T_f 之间,则定义为 NMV。这两个阈值由 MV 场的噪声程度和具体序列情况而定,可以经过统计后的数据结合经验确定。

2.2 全局运动估计和补偿

在背景运动的视频序列中,无法直接对运动矢量场进行处理。为消除由摄像机引起的背景变化的影响,需要先对当前视频帧进行全局运动估计和补偿^[7]。

在原始运动矢量场 MV 上进行全局运动补偿,采用六参数的仿射运动模型来估计全局运动模型,该仿射运动模型如下:

$$\begin{cases} x'_i = a_0 x_i + a_1 y_i + a_2 \\ y'_i = a_3 x_i + a_4 y_i + a_5 \end{cases} \quad (1)$$

其中, $[a_0, a_1, a_2, a_3, a_4, a_5]$ 是模型参数向量, (x_i, y_i) 为 MV 中任意 4×4 块的中心坐标,利用上述全局运动模型估计所得的运动矢量为 $EMV(x_i, y_i) = [x'_i - x_i, y'_i - y_i]^T$ 。通过对如下定义的目标函数 φ 的最小化来求解全局运动模型参数。

$$\varphi = \sum_i e_i = \sum_i \|EMV(x_i, y_i) - MV(x_i, y_i)\|^2$$

其中, e_i 为全局运动模型估计的误差,全局运动模型参数采用迭代法求解,每次迭代去除一定比例的残差较大的运动矢量(可能属于运动对象),剩余的运动矢量则用来求解更新后的模型参数,直至模型参数收敛。参数的更新采用 Newton-Raphson 方法,以保证参数估计能够快速收敛。图 2(a)与图 2(b)分别为 coastguard 序列的原始运动矢量场和经过全局运动补偿后的矢量场。

2.3 EM 算法分割运动对象

前面已按图 1 的流程经过中值滤波、全局运动补偿和 MV 的分类处理,接着使用 EM 算法来进行运动对象分割^[1]。此时,已将 BMV 和 NMV 排除在外,应用 EM 算法将剩余的 FMV 分割成不同的层。

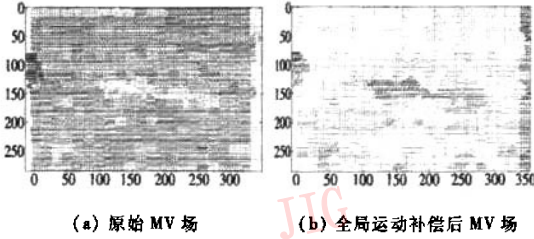


图2 运动补偿前后矢量场对比

Fig.2 MV field after motion compensation

给定运动模型的数量 n 和用仿射参数矢量 $\{a_1, a_2, \dots, a_k\}$ 表示相应的初始运动假设,算法在 E 步骤和 M 步骤间交替直到收敛。

2.3.1 E 步骤

E 步骤计算每个运动矢量 $v(B)$ 属于第 j 个运动模型 u_j 的后验概率:

$$P_j(B) = P_j \quad (2)$$

其中, $B = [p \ q]$ 为运动矢量对应的 4×4 块的位置坐标,而 $u_j^T = [v_x^j, v_y^j]$ 为第 j 个运动模型的参数向量。设

$$r_j(B) = \|u_j - v(B)\|^2 \quad (3)$$

是第 j 个运动模型 u_j 和 B 位置上运动矢量 $v(B)$ 的平方差。所以运动矢量 $v(B)$ 属于第 j 个运动模型 u_j 的后验概率

$$P_j(B) = \frac{e^{-r_j(B)/2\sigma^2}}{\sum_{i=1}^n e^{-r_i(B)/2\sigma^2}} \quad (4)$$

这里, n 是给出运动模型的个数, σ^2 用来控制适合于运动矢量模型的逼真度。

2.3.2 M 步骤

M 步骤重新估计每个运动模型的参数,利用最小二乘估计使下列的误差函数最小

$$u_i = \arg \min \sum_B P_j(B) r_j(B) \quad (5)$$

从而得到更新的运动模型参数为

$$u_i = \left[\sum_B P_j(B) \right]^{-1} \sum_B P_j(B) v(B) \quad (6)$$

算法的具体步骤如下:

- (1) 定义 N 个运动参数 $u_1^{(1)}, u_2^{(1)}, \dots, u_N^{(1)}$ 。
- (2) 在第 i 次迭代,计算每个运动矢量属于每个运动模型的概率 $P_j^{(i)}(B)$ 。
- (3) 按照式(6)更新运动参数。
- (4) 如果 $u_j^{(i+1)} = u_j^{(i)}$, $j = 1, 2, \dots, N$, 迭代终止。

否则跳至步骤 2。

经过 E 步骤和 M 步骤的多次迭代,最后通过硬阈值化后验概率 $P_j(B)$ 得到最后的对象掩模。一般经过 4~6 次迭代就足够分割出对象掩模。运动矢量根据下式被分配给相应的模型:

$$Z_j(B) = \begin{cases} 1 & P_j(B) > T_p \\ 0 & \text{其他} \end{cases} \quad (7)$$

其中, T_p 为判定运动矢量是否属于第 j 运动矢量模型的阈值,第 j 个运动模型的最终对象掩模由 $Z_j(B)$ 给出。

3 实验结果与分析

以 table tennis, hall 和 coastguard(CIF 格式)作为实验测试序列,3 个序列都有对象的复杂运动。实验使用 JM8.6 版本的 H.264 编码器对序列进行编码。

在 EM 聚类分割阶段,参数 σ^2 和 T_p 的取值范围是 0.01~0.09 和 0.7~0.9,在实验中根据不同序列的分割效果进行调整。table tennis 序列中,由图 3(b)、(e)可以看出,虽然球的尺寸很小,但本文算法仍能清晰准确的分割出来,而文献[1]的算法对运动对象有明显的漏分割和空洞现象(图 3(c)、(f))。本文算法在主观上效果更好。

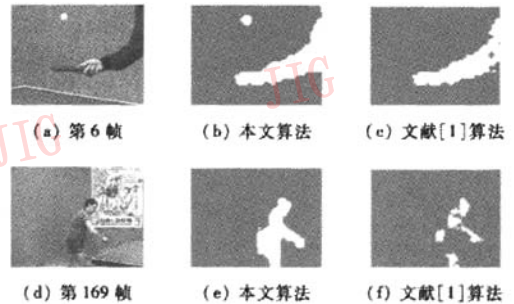


图3 Table tennis 序列的运动对象分割结果

Fig.3 Segmentation results of sequence table tennis

hall 序列包含复杂的背景和多个较小的运动对象,对象有一些剧烈的运动。原始图像为单个对象进入场景和提着物体离开场景的情况。本文算法分割结果(图 4(b)、(e))在视觉上明显优于文献[1]的分割结果(图 4(c)、(f))。

图 5 是对序列 coastguard 的运动对象的分割实验,该序列有明显的全局运动,对象移动,加上水波的影响,对分割有一定干扰。

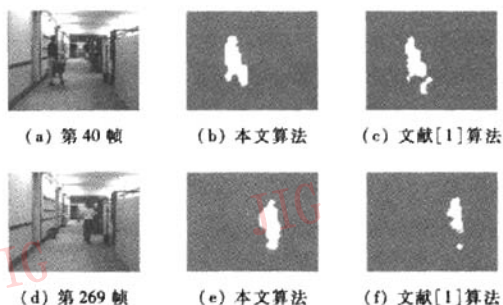


图 4 Hall 序列的运动对象分割结果

Fig. 4 Segmentation results of sequence hall

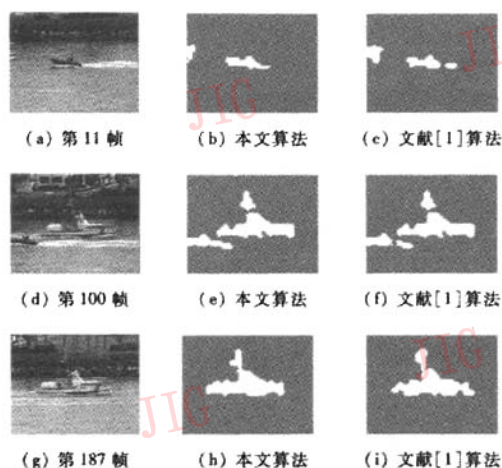


图 5 Coastguard 序列的运动对象分割结果

Fig. 5 Segmentation results of sequence coastguard

由于摄像机和船基本都做水平移动,而船的底部运动在水平方向与水面基本相似,少量由小船激起的水花被误分割为运动对象如图 5(b)。图 5(d)为两船相遇后小船即将离开场景的情况。本文算法未采用 k-means 算法来估计更新运动对象数量,但由于采用的运动矢量预处理滤除了绝大部分的噪声块,只要根据场景情况手工指定合适的运动对象数量,仍能较精确和鲁棒跟踪对象的变化如图 5(e)。在序列后期摄像机在经过向上的剧烈运动后,从跟踪小船转而跟踪大船后,场景中对象的水平运动逐渐稳定,加之小船运动目标较大,通过本文算法得到了较清晰准确的分割结果如图 5(h),完整地包括了船体轮廓、桅杆和旗帜,在主观视觉效果上要优于文献[1]中的分割算法如图 5(i)。

在未经任何优化的情况下,在 CPU 为

AMD1800、内存 256M 的微机上测试了本文算法的运行时间,其中,table tennis 为 15fps, hall 为 13fps, coastguard 为 10fps,可见本文算法的处理速度可以满足视频对象分割应用的实时性要求。

4 结 论

本文提出了一种 H. 264 压缩域中分割运动对象的方法,该算法先对运动矢量场进行全局运动估计并利用运动矢量的幅值进行分类处理,最后对非零运动矢量使用改进的 EM 聚类。实验结果显示算法针对静态背景和运动背景的视频序列都能取得较好的分割效果,并能实现实时分割。

参考文献 (References)

- 1 Venkatesh Babu R, Ramakrishnan K R, Srinivasan S H. Video object segmentation: A compressed domain approach [J]. IEEE Transactions on Circuits System for Video Technology, 2004, 14(4): 462 ~ 474.
- 2 Mezaris V, Kompatsiaris I, Boulgouris N V, et al. Real-time compressed-domain spatiotemporal segmentation and ontologies for video indexing and retrieval[J]. IEEE Transactions on Circuits and System for Video Technology, 2004, 14(5): 606 ~ 621.
- 3 Sukmaro O, Rao K R. Fast object detection and segmentation in MPEG compressed domain[A]. In: Proceedings of IEEE TENCON (IEEE Region 10 International Conference) [C], Kuala Lumpur, Malaysia, 2000, 3: 364 ~ 368.
- 4 Wiegand T, Sullivan G J, Bjontegaard G, et al. Overview of the H. 264/AVC video coding standard [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2003, 13(7): 560 ~ 576.
- 5 Zeng W, Du J, Gao W, et al. Robust moving object segmentation on H. 264/AVC compressed video using the block-based MRF model [J]. Real-Time Imaging, 2005, 11(4): 290 ~ 299.
- 6 Liu Zhi, Lu Yu, Zhang Zhao-yang. Real-time spatiotemporal segmentation of video objects in the H. 264 compressed domain[J]. Visual Communication and Image Representation, 2007, 18(3): 275 ~ 290.
- 7 Su Y P, Sun M T, Hsu V. Global motion estimation from coarsely sampled motion vector field and the applications [J]. IEEE Transactions on Circuits and System for Video Technology, 2005, 15(2): 232 ~ 241.