

从 h.264 压缩码流提取运动对象的一种方法

张智福, 康志伟

(湖南大学计算机与通信学院, 长沙 410082)

摘 要: 提出一种基于 h.264 压缩域视频流中运动对象的分割方法。对提取的运动矢量场进行噪声处理, 在空间检测和校正、中值滤波等处理后, 得到可靠的运动矢量场。为进一步增强可靠的运动向量, 利用后向投影迭代技术, 采用均值聚类算法获得视频对象分割。仿真实验证明, 该方法可以得到良好的分割质量, 且码流无须解码, 实时性良好。

关键词: 压缩域; 运动对象分割; 运动矢量累积

Method of Moving Object Extraction from h.264 Compressed Stream

ZHANG Zhi-fu, KANG Zhi-wei

(School of Computer and Communication, Hunan University, Changsha 410082, China)

【Abstract】 This paper proposes a video moving object segmentation algorithm based on h.264 compressed domain. Motion vector field extracted from h.264 compressed stream is processed by noise processing and median filter. An iteratively backward projection scheme is proposed to obtain an accumulated motion vector field. Moving object region is extracted in turn based on k-means cluster. Experimental results show moving object is segmented well without decoding video compressed stream.

【Key words】 compressed domain; moving object segmentation; motion vector accumulation

1 概述

视频对象分割是从视频序列帧中分离出具有一定意义的实体。近年来随着智能视频监控系统、目标识别、视频数据库检索等领域的蓬勃发展, 视频对象分割技术的研究已经成为当前视频研究领域的热点问题。而如何从数字视频信号中提取出具有意义的运动视频对象, 是视频运动对象分割的关键技术。当前, 视频数据的存储、传输都是压缩域的形式, 特别是最新一代的压缩编码技术 h.264 的兴起, 研究基于 h.264 压缩域的视频对象分割成为一个热门课题。

在压缩域中, 研究视频对象分割的方法不是很多。基于压缩域的分割方法, 由于基于的不是像素而是对象宏块, 因此在很大程度上能够减少计算的复杂度, 提高分割对象的实时速度, 但是同时也降低了分割的精度。在 MPEG 压缩域可以利用的分割对象的信息有宏块运动矢量和 DCT 系数, 而在 h.264 压缩域能直接利用的就只有运动矢量场。文献[1]通过时域累积和空域内插来估计一个稠密的运动矢量场, 再利用 EM 算法进行聚类, 以此分割运动对象; 文献[2]在区域分割中, 利用宏块级的 DCT 系数, 通过 2 个阶段的步骤, 提取出近似于像素域的分割精度, 但是方法非常复杂, 计算时间比较长。文献[3]提出直接从稀疏运动矢量场中分割运动对象的方法, 虽然能够应用于动态背景序列, 但是方法较复杂, 计算量较大。文献[4]用基于双线性运动模型的迭代拒绝方法进行前景/背景的分割, 通过检测迭代拒绝输出的时域一致性将得到的前景宏块聚类成连通区域, 最终进行区域跟踪, 构成有意义的前景时空对象。文献[5]提出了一种基于块的 MRF 模型从稀疏运动矢量场中分割运动对象的方法, 根据各个块运动矢量的幅值赋予各个块不同类型的标记, 通过最大化

MRF 的后验概率标记出属于运动对象的块, 但这种方法只局限于静态背景序列。

2 算法概述

本文提出的 h.264 压缩域内的运动视频对象分割算法无须解压原始视频数据, 直接从压缩域码流中提取运动矢量场, 再对运动矢量场进行累积、去噪、补偿等处理, 得到可靠的运动矢量场后利用 k-means 聚类算法检测出运动视频对象。系统算法框图如图 1 所示。

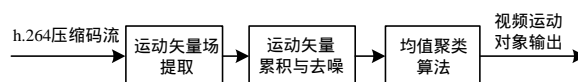


图 1 系统算法框图

3 运动向量处理

在 h.264 压缩码流中, 运动矢量的目的是为了减少码流, 提高编码的压缩比, 所以运动矢量场是不可靠的、有噪声的。另外, 压缩域的运动矢量是基于宏块的, 一帧图像的运动矢量场比较稀疏, 为了提高运动对象分割的准确性, 必须对运动矢量场进行处理。

3.1 矢量场噪声处理

由于图像噪声等原因, 用块匹配法估计运动向量时, 静止的区域往往产生随机噪声运动向量且其幅值比较小(通常

基金项目: 国家自然科学基金资助项目(60702065); 湖南省自然科学基金资助项目(06JJ50114); 湘潭市科技计划基金资助重点项目(ZJ20071008)

作者简介: 张智福(1982 -), 男, 硕士研究生, 主研方向: 视频对象处理; 康志伟, 教授

收稿日期: 2010-04-30 **E-mail:** dzxx2008@yahoo.com.cn

位于零值附近),于是可以设定一个阈值 T (阈值过高则有可能将一些运动的宏块漏掉, 阈值过低则有可能将有背景噪声的宏块当成了运动块)将噪声运动向量滤除。为了在实际中降低复杂度和提高计算效率, 可以求运动矢量的 2 个分量的绝对值之和作为一个参数, 如下式所示:

$$MVi = \begin{cases} MVi & |MVi_x| + |MVi_y| \geq T \\ 0 & |MVi_x| + |MVi_y| < T \end{cases} \quad (1)$$

其中, MVi 表示第 i 个宏块的运动矢量; MVi_x 、 MVi_y 分别代表第 i 个宏块的 x 、 y 方向的运动矢量。

3.2 矢量场空间检测和校正

运动向量是有噪声的, 部分运动向量并不能代表宏块的运动向量, 为了得到可靠的运动矢量场, 需要对运动向量空间进行空间检测和校正。强度和方向的检测是看当前运动向量背离相邻运动向量的程度。如果背离程度大(方向和强度突然变化), 则认为当前运动向量是不可靠的, 将不可靠的运动向量用邻域运动向量的中值进行修正。

$$MV_Dev_{i,j} = \frac{|MV_{i,j} - \sum_{(i',j') \in \theta} MV_{i',j'} / N|}{\sum_{(i',j') \in \theta} |MV_{i',j'}| / N} \quad (2)$$

其中, $MV_Dev_{i,j}$ 为索引 (i,j) 处的运动矢量强度 $MV_{i,j}$ 背离相邻非零运动矢量的程度; θ 表示一种 8 邻域或 4 邻域结构; N 为邻域中非零运动矢量的个数; $MV_{i',j'}$ 为邻域运动矢量。

定义当前运动矢量与相邻运动矢量的方向背离值:

$$MV_DirDev_{i,j} = \frac{\sum_{(i',j') \in \theta} |MV_Ang_{i,j} - MV_Ang_{i',j'}|}{N} \quad (3)$$

其中, $MV_Ang_{i,j}$ 为 (i,j) 处的运动矢量方向角度幅值; $MV_Ang_{i',j'}$ 为邻域运动矢量幅值; N 、 θ 同上; $MV_DirDev_{i,j}$ 为运动矢量方向角度背离值。

于是可以对通过运动矢量强度背离值和方向背离值将不可靠的运动矢量检测出来。接下来对运动矢量场进行二维中值滤波。中值滤波可以平滑运动向量, 让与周围相差比较大的运动矢量替换为与周围接近的运动矢量, 可以消除椒盐噪声、脉冲噪声等, 还有中值滤波不是线性的滤波可以较好地保持边缘。

$$MV(x, y) = \text{median}[mv_n(i, j)] \quad (4)$$

其中, $MV(x, y)$ 为 (x, y) 处的运动矢量; $mv_n(i, j)$ 为大小 n 窗口内所有的运动矢量。

视频运动对象帧在经过上述运动矢量空间检测和校正、中值滤波一系列处理之后可以得到比较精确的运动矢量场。

图 2 所示是 tennis 视频序列的第 12 帧经过上述处理之后的运动矢量场。图 2(b) 是第 12 帧的原始运动矢量场, 图 2(c) 是经过处理之后的运动矢量场。

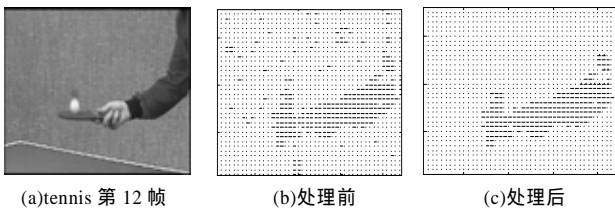


图 2 tennis 视频帧处理

3.3 矢量场的累积

h.264 中可以对连续的多帧运动矢量场进行累积, 以增强每一场的运动矢量信息, 以此分割出更加可靠的运动对象。

任意帧的运动矢量场的累积都需要利用其后的多帧运动矢量场。这些运动矢量场相邻两帧采用后向投影, 让后迭代多帧的运动矢量场得到当前帧的运动矢量场, 如图 3 所示。

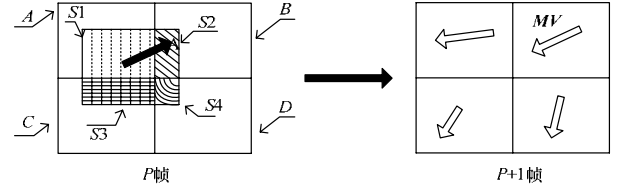


图 3 运动矢量累积过程示意图

图 3 所示的是 2 帧图像的运动矢量后向累积过程。右图右上块在 P 帧中的经过运动矢量匹配之后的最佳匹配块为画有不同纹理的阴影块。在 P 帧中根据阴影块所占的不同面积来计算运动矢量场。在 $P+1$ 帧右上块的运动矢量为 MV , 采用后向投影, 所以阴影块的运动矢量如图中所示。 $S1$ 、 $S2$ 、 $S3$ 、 $S4$ 分别代表相应纹理块面积, A 、 B 、 C 、 D 代表相应块。所以可以得到 A 块的累积运动矢量为:

$$MVA = mva + MV \times (S1/16)$$

其中, mva 代表原来的运动矢量; MVA 为累积之后的运动矢量, 这里的 16 为一个 4×4 块的面积。同理可以得到 MVB 、 MVC 、 MVD , 即为 P 帧累积后的相应块的运动矢量。

4 k-means 聚类算法分析

为了对多视频运动对象场景中的多个运动对象进行分割, 可以利用聚类分析算法对运动矢量场进行分析, 以此分割出场景中不同的运动对象。

本文采用 k-means 聚类对经过累积后的非零运动矢量场进行分析。假设在场景内的运动对象的个数为 k 个, 那么聚类的过程如下:

(1) 选出 k 个非零运动矢量聚类初始平均值:

$$mv_1^{(1)}, mv_2^{(1)}, \dots, mv_{k-1}^{(1)}, mv_k^{(1)}$$

(2) 在第 i 次迭代中, 根据下列关系, 把每个宏块的运动矢量 mv 赋给 k 个矢量聚类中的一个; $\|mv - mv_j^{(i)}\| < \|mv - mv_l^{(i)}\|$, 对于所有的 $l, l=1, 2, \dots, k-1, k, l \neq j, mv \in \wedge_j^{(i)}$, 其中, $\wedge_j^{(i)}$ 表示第 i 次迭代时, 以 $mv_j^{(i)}$ 为聚类中心的运动矢量分组。

(3) 修改矢量聚类均值, $mv_l^{(i)}, l=1, 2, 3, \dots, k-1, k$, 把它修改成相应非零运动矢量分组的平均值:

$$mv_l^{(i+1)} = 1/N_l \left(\sum_{mv \in \wedge_l^{(i)}} mv, l=1, 2, \dots, k-1, k \right)$$

其中, N_l 是 $\wedge_l^{(i)}$ 中包含的非零运动矢量个数。

(4) 如果对于所有的 $mv_l^{(i+1)} = mv_l^{(i)}$, 过程终止。否则, 跳到第(2)步继续执行。

用 k-mean 聚类算法得到的 k 个非零运动矢量聚类中心对应着场景中 k 个非零运动矢量的集合。因为运动矢量代表了当前帧像素块相对于其参考帧块的位移, 聚类了运动矢量, 也就聚类了像素块。所以每个非零运动矢量集合对应的非零运动矢量区域就是相对应的运动对象区域。

5 实验结果

本文通过 2 个典型的标准测试序列来验证本文提出的算法: tennis 序列(QCIF, 176×144)和 Claire 序列(QCIF, 176×144)。压缩视频流来自于 JM8.6 编码器, baseline profile, 每 20 帧插入一个 P 帧, 量化数为 28, 搜索范围为 $(-16, 16)$ 。在 tennis 序列中, 选择第 4 帧作为实验数据, 设定强度背离值

(下转第 236 页)