

应用递归最短生成树算法实现 H. 264 压缩域运动对象分割方法

赵 锟^{1),2)} 张文俊^{1),2)} 李 蔚^{1),2)}

¹⁾(上海大学通信与信息工程学院, 上海 200072) ²⁾(上海大学数字媒体技术艺术实验室, 上海 200072)

摘 要 提出了一种基于递归最短生成树算法的 H. 264 压缩域实时分割运动对象的算法。首先将从 H. 264 编码端提取的运动矢量进行归一化、空间内插, 得到稠密运动矢量场, 再采用全局运动补偿技术抵消全局运动的影响, 最后采用改进的“递归最短生成树”(RSST)算法对稠密运动矢量进行聚类, 实现对运动对象的分割。实验结果表明, 该算法对视频序列能实现较准确的分割。

关键词 H. 264 压缩域 运动对象分割 递归最短生成树

中图法分类号: TP301.6 文献标识码: A 文章编号: 1006-8961(2009)10-2154-05

H. 264 Compressed Domain Moving Object Segmentation Used RSST Algorithm

ZHAO Kun^{1),2)}, ZHANG Wen-jun^{1),2)}, LI Wei^{1),2)}

¹⁾(School of Communication and Information Engineering, Shanghai University, Shanghai 200072)

²⁾(Laboratory of Digital Media Arts and Technology, Shanghai University, Shanghai 200072)

Abstract In this paper, we present a novel approach to segmenting moving objects from the H. 264 compressed based on recursive shortest spanning tree(RSST). Firstly, the motion vector is obtained from the H. 264 encoder and then a dense vector field is formed after the vector is unified and accumulated with a backward iterative projection process. Secondly, global motion compensation is used to eliminate the global motion. Thirdly, a modified RSST arithmetic is used to make the blocks clustered. Thus, with the proposed method we achieved moving object segmentation. Experimental results for several video sequences demonstrate the effectiveness of the proposed approach.

Keywords H. 264, compressed domain, moving object segmetation, recursive shortest spanning tree(RSST)

1 引 言

随着计算机和通信技术的发展,大量存储和传输的视频数据都采用不同的压缩标准完成。其中 H. 264 以比 MPEG2 高一倍的编码效率成为主流的视频压缩编码标准。为直接从压缩数据中提取信息以便后续更加快速地进行视频对象的编辑或检索等,压缩域视频对象分割成为首先要解决的问题。

早在上世纪末,Zen 等人就已经凭借从 MPEG2

编码器得到的运动矢量(MV)的大小和夹角完成对噪声的滤除和区域合并^[1]。之后,文献[2]将经典的 EM(期望最大化)迭代算法引入 MPEG 域视频对象分割,被视为这一研究领域的里程碑。文献[3]则将 DCT(离散余弦变换)系数的使用视为在对分割精度有进一步要求时对基于 MV 分割的一种补充。

针对 H. 264 压缩域,分割方法主要基于运动矢量场。文献[4]提出一种“基于块的马尔可夫随机场(MRF)模型”来直接对稀疏运动矢量场进行分割得到运动对象;文献[5]采用“基于匹配矩阵的运动

对象分割方法”有效地处理了对象的合并与分裂、对象的出现与消失等情况。

本文提出一种改进的“递归最短生成树”(RSST)算法对稠密运动矢量场进行聚类,实现运动对象的分割。由于不需要反复迭代,运算效率高于 EM 等传统方法,更加适合实时分割场合。

2 对运动矢量场的预处理

2.1 运动矢量场的归一化与累积

H. 264 中宏块的尺寸可能是 $4 \times 4, 4 \times 8, 8 \times 4,$

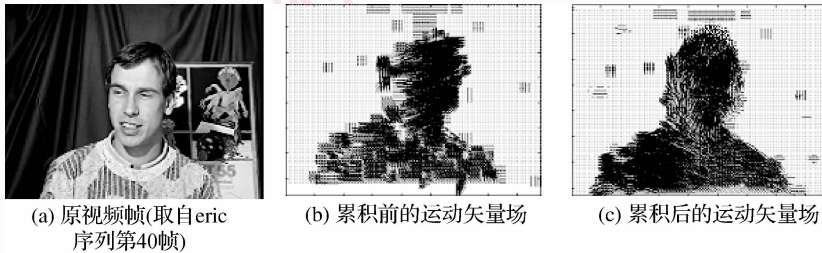


图 1 运动矢量累积效果

Fig. 1 The effect of MV accumulation

2.2 中值滤波与全局运动补偿

考虑到中值滤波能在很好地去掉孤立点和线噪声的同时保持图像的边,尤其能有效去除二值噪声,因此采用 3×3 窗口的中值滤波去除累积后稠密矢量场的噪声。

在累积后的运动矢量场 MV''_i 上进行全局运动估计,去除全局运动的影响。全局运动模型采用 6 参数仿射运动模型通过对如下定义的目标函数 Θ 的最小化来求解全局运动模型参数:

$$\Theta = \sum_i e_i = \sum_i \|mv(x_i, y_i) - MV''_i(x_i, y_i)\|^2 \quad (1)$$

其中, e_i 为任意块 P_i 利用全局运动模型估计的误差。全局运动模型参数的求解采用了迭代求解的方法^[6],每次迭代去除一定比例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。参数的更新采用了 Newton-Raphson 的方法,以保证参数估计能够快速收敛。

在归一化阶段,由于原始运动矢量场的最小块尺寸为 4×4 ,因此将尺寸大于 4×4 的块的运动矢量直接赋给该块覆盖的所有 4×4 块,这样就得到了

$8 \times 8, 8 \times 16, 16 \times 8, 16 \times 16$ 中的一种,在进行对象分割之前,需要对原始的运动矢量场进行时域的归一化^[5],并采用“后向迭代投影法”^[5]对归一化后的运动矢量场进行累积得到稠密运动矢量场。

在归一化阶段,由于原始运动矢量场的最小块尺寸为 4×4 ,因此将尺寸大于 4×4 的块的运动矢量直接赋给该块覆盖的所有 4×4 块,这样就得到了一个采样精度为 4×4 块的均匀运动矢量场。

从图 1(b)和(c)的对比中可以看出:经过累积后,可靠的运动信息得到了显著增强。

一个采样精度为 4×4 块的均匀运动矢量场。

3 基于改进的 RSST 方法的分割算法

在对运动矢量进行了充分的预处理之后进入区域分割步骤,分割方法采用图 2 所示的改进后的 RSST 算法。

3.1 运动矢量场与无向图之间的映射关系

由于 RSST 算法基于“无向图”这种数据结构,因而首先要求出经过预处理的单帧运动矢量场图对应的无向图。

设经过全局运动补偿的运动矢量场为 D 。这时的矢量场 D 是由每 4×4 块一个运动矢量 v 构成。设 v_x, v_y 分别为 v 的横向分量和纵向分量,则矢量场 $D(P, v)$ 与无向图 $G(V, E)$ 间的映射关系定义如下:

$$V = f(P) \quad (2)$$

$$E = f[g(P_i, P_j)] \quad (3)$$

$$W = \bar{v} = \frac{1}{N} \sum_{n=1}^N v^n \quad (4)$$

$$W_E = |W_i - W_j| = \sqrt{(\bar{v}_x^i - \bar{v}_x^j)^2 + (\bar{v}_y^i - \bar{v}_y^j)^2} \quad (5)$$

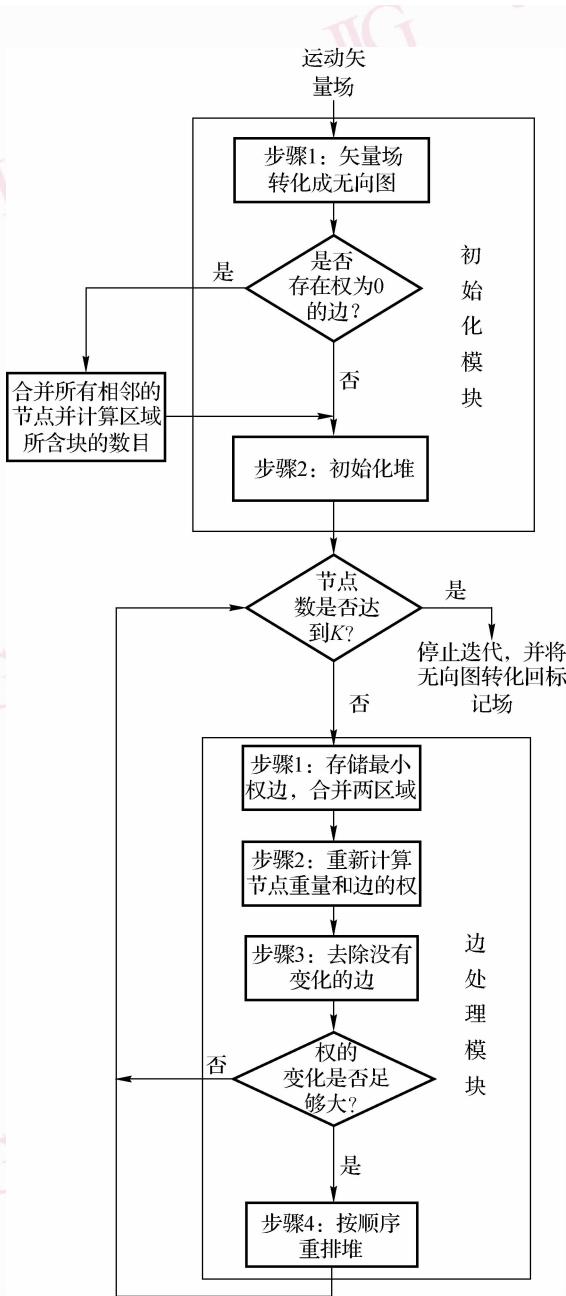


图 2 改进后的 RSST 算法框图

Fig. 2 The block diagram of improved RSST algorithm

式中, V 和 E 分别表示无向图的节点和边, W 和 W_e 则分别表示节点 V 的重量和边 E 的权值; f 表示一种映射关系, 而 g 表示相邻区域的某种相互关系; P 表示矢量场中的区域, 初始化时每个区域只包含一个 4×4 块; N 表示区域所含的 4×4 块的数量; \bar{v} 表示区域的平均运动矢量; $\bar{v}_x^i, \bar{v}_x^j, \bar{v}_y^i, \bar{v}_y^j$ 分别表示两相邻区域 P_i 和 P_j 的平均横向分量和平均纵向分量。

边的权值直观地反映出了相邻区域平均运动矢量的相似程度。这种表述方式简洁有效地将运动矢量场表述成描述运动一致性的关系图。

3.2 0 权值边的检测和节点合并

在进入初始化堆 1 之前, 首先判断无向图内是否存在权值为 0 的边, 若有则先将这些节点合并 (图 3)。具有 0 权值边的节点实际上代表了具有相同运动矢量的块。比如静止背景中的大片区域和运动完全一致的某些区域。优先合并这些区域后, 如堆的节点会有明显减少, 大大提高了算法的效率。

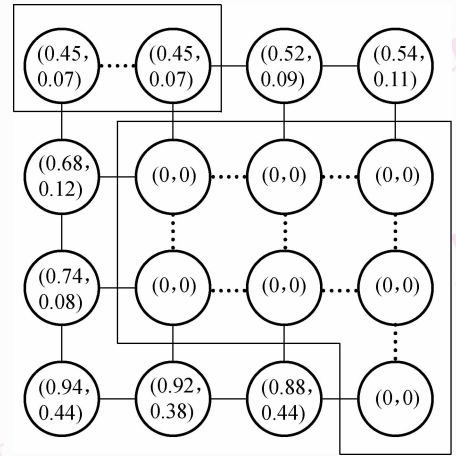


图 3 无向图中存在大量权值为 0 的边

Fig. 3 Edges of which weight values are 0 in the undirected graph

3.3 对权值变化量的阈值处理

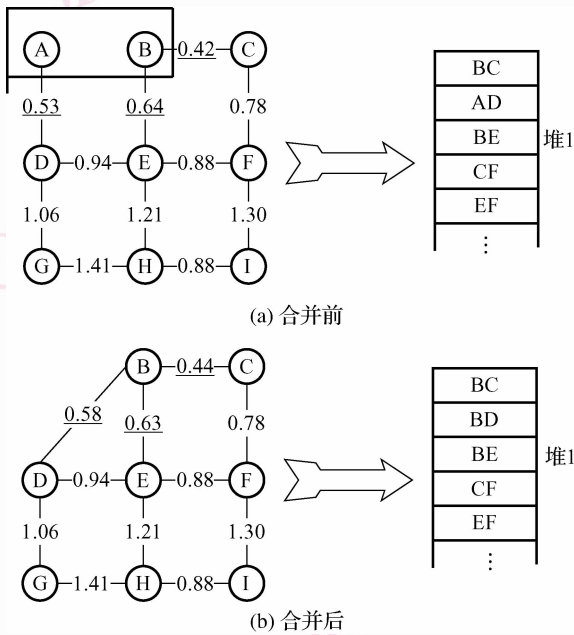
在将大量 0 权值边的节点合并为少数几个节点之后, 进入初始化堆 1 的步骤, 即将所有的边和边的权值按照由小到大的顺序存储在堆 1 里。至此, 初始化模块结束。

边处理模块的原理与传统 RSST 算法^[7]类似: 循环下列步骤直到整个无向图的节点数减少到用户手动设定的对象数目 K 为止。

- (1) 将堆 1 里权值最小的边存储在堆 2 中, 并将此边的两个节点合并;
- (2) 在更新的图中重新计算各节点重量个边的权值;
- (3) 将端节点和权值都没有改变的边去除;
- (4) 将留下的边按照权值从小到大的顺序存储在堆 1 里 (即更新了堆 1)。

上述迭代步骤中, 最后一步对堆 1 的更新需要按照更新后的权值大小重新对边进行排序, 是计算量比较大的步骤。为尽可能减少重排边的数目以进

一步提高分割效率, 设置一个阈值来判断权值的改变量是否足够大, 仅当 $\frac{W'_E - W_E}{W'_E} \geq m (m > 0)$ 时重排堆 1。其中阈值 m 的具体数值由实验得出。这里推荐为 10%。实验结果证明, 在合并节点的过程中合并后权值没有明显改变的情况经常发生, 如图 4 所示。这些边进行重排后顺序没有发生任何变化, 重排显得没有必要。本文的阈值处理跳过了大量这种无效重排步骤, 为分割节省了大量时间。



有下划线的数字表示权值有变化的边

图 4 堆 1 重排后没有发生变化

Fig. 4 Nothing changes in stack 1 after being rearranged

4 实验结果及评价

实验硬件平台为普通 PC 机, 软件平台为 VC6.0; 运动矢量场从 JM8.6 版本的 H. 264 编码器提取, 编码器配置如下: Baseline Profile, IPPP, 每 30 帧插入一个 I 帧, 3 个参考帧, 运动估计的搜索范围为 $[-31, 32]$, 量化参数 30, 编码帧数为 300 帧; 所用测试序列为国际标准测试序列。

序列 Coastguard 具有明显的全局运动。由实验结果图可以看出, 算法能够比较准确地分割出两个运动对象所在的区域。图 5 以第 15 帧为例出示了分割结果。

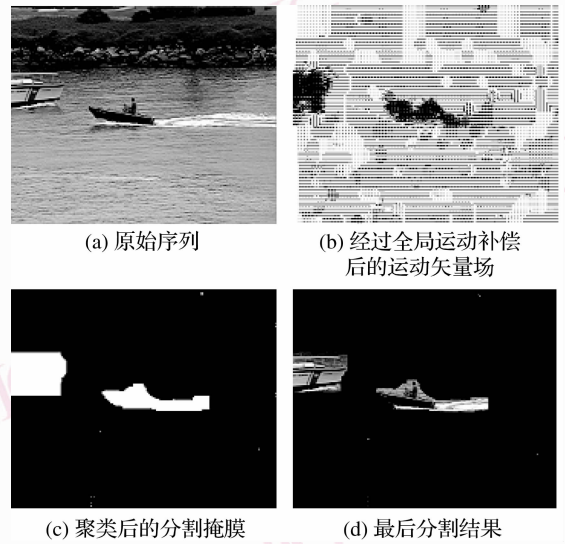


图 5 Coastguard 序列第 15 帧分割过程

Fig. 5 The segmentation of the 15th fram of Coastguard

补偿后的运动矢量中仍存在的噪声导致分割后背景中的个别噪点; 剧烈复杂的运动使分割不能非常精确地与实际运动对象边缘匹配起来; 对溅起的水花等与运动对象具有相似运动的干扰误当做运动对象分割出来。这些还有待进一步改进。图 6 给出了序列其他部分帧的分割结果。

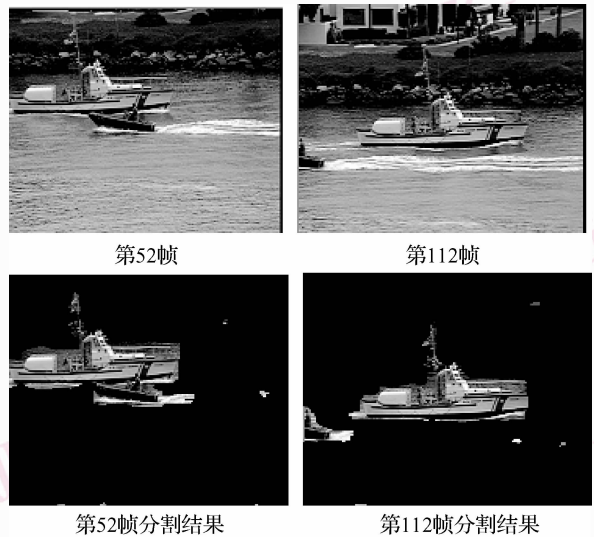


图 6 Coastguard 序列部分分割结果

Fig. 6 Part segmentation results of Coastguard

通过对传统 RSST 方法的改进, 使运算量大幅度减少, 分割速度有了很大提升。表 1 给出了使用传统 RSST 算法和本文算法对多个序列分割时间的比较。可见使分割速度提高了近一倍。

表 1 传统 RSST 算法与改进后算法分割时间比较

Tab. 1 Compare of segmentation time between the RSST algorithm before and after improvement

	序列			
	Eric	Moth. & Daug.	Children	Coastguard
格式/帧数	CIF /50	CIF/300	CIF/300	CIF/300
运动对象数目	1	2	3	2(有全局)
传统 RSST 每帧时间(ms)	62	69	72	78
改进 RSST 每帧时间(ms)	33	32	34	39

5 结 论

提出了一种基于递归最短生成树算法的 H. 264 压缩域运动对象分割算法。在对运动矢量场进行累积和滤波等一系列预处理之后,使用基于改进的 RSST 方法的分割算法对运动矢量场进行分割,最终获得满意的分割结果。实验结果表明该算法对静态背景和运动背景的视频序列都能实现较准确的分割,并且改进的 RSST 方法比使用传统 RSST 实现分割的时间缩短了一半左右。

参考文献 (References)

- 1 Heitou Z E N, Tameharu HASEGA W A, Shinji OZA WA. Moving object detection from MPEG coded picture [A]. In: Proceedings of IEEE International Conference on Image Processing, 1999 [C], Kobe, Japan, 1999, 4: 25-29.
- 2 Venkatesh Babu R, Ramakrishnan K R, Srinivasan S H. Video object segmentation: A compressed domain approach [J]. IEEE Transactions on Circuits System and Video Technology, 2004, 14(4): 462-474.
- 3 Eng H L, Ma K K. Spatiotemporal segmentation of moving video

objects over MPEG compressed domain [A]. In: Proceedings of IEEE International Conference on Multimedia and Expo, 2000 [C], New York, NY, USA, 2000, 3: 1531-1534.

- 4 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.
- 5 Liu Zhi, Lu Yu, Zhang Zhao-yang. Real-time spatiotemporal segmentation of video objects in the H. 264 compressed domain [J]. Journal of Visual Communication and Image Representation, 2007, 18(3): 275-290.
- 6 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 System and Video Technology, 2005, 15(2): 232-241.
- 7 Morris O J, Lee M J, Constantinides A G. A unified method for segmentation and edge detection using graph theory [A]. In: Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 1986 [C], Tokyo, Japan, 1986, 11: 2051-2054.
- 8 Morris O J, Lee J M, Constantinides A G. Graph theory for image analysis: An approach based on the shortest spanning tree [J]. IEEE Proceedings F: Communication, Radar and Signal Processing, 1986, 133(2): 146-152.