

基于 H. 264 压缩域的实时运动对象分割算法

刘 志¹⁾ 徐剑峰¹⁾ 陆 宇¹⁾ 张兆杨^{1),2)}

¹⁾(上海大学通信与信息工程学院,上海 200072) ²⁾(新型显示技术及应用集成教育部重点实验室,上海 200072)

摘 要 在压缩域内直接分割运动对象对于有实时要求的应用而言是十分必要的,H. 264 以其优越的压缩效率已经在许多应用中逐渐取代了 MPEG-2/4,但有关在 H. 264 压缩域内进行运动对象分割的研究还很少。为此提出了一种从 H. 264 压缩域实时分割运动对象的算法,该算法首先对从 H. 264 视频中提取出的原始运动矢量场进行时域和空域的归一化,接着通过对连续多帧的运动矢量场进行累积来增强显著的运动信息;然后对累积运动矢量场进行全局运动补偿,同时利用快速的统计区域生长算法按照运动相似性将其分割成多个区域;最后利用运动矢量场的方向角直方图来判断出属于运动对象的分割区域,以组成运动对象。通过对多个 MPEG-4 测试序列的实验结果表明,该方法不仅能够从 H. 264 压缩域中实时地分割出运动对象,且具有良好的分割质量。

关键词 运动对象分割 H. 264 压缩域 运动矢量场

中图法分类号: TN941.1 TP911.73 **文献标识码:** A **文章编号:** 1006-8961(2006)11-1614-06

A Real-time H. 264 Compressed Domain Based Moving Object Segmentation Algorithm

LIU Zhi¹⁾, XU Jian-feng¹⁾, LU Yu¹⁾, ZHANG Zhao-yang^{1),2)}

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

²⁾(Key Laboratory of Advanced Display and System Application, Ministry of Education, Shanghai 200072)

Abstract Moving object segmentation in the compressed domain is absolutely necessary for real-time applications. Due to the predominant compression efficiency, the emerging video coding standard H. 264 is replacing MPEG-2/4 in many multimedia applications, but moving object segmentation in the H. 264 compressed domain is rarely investigated until now. In this paper, we present a new approach to segment moving objects from the H. 264 compressed domain. The motion vector (MV) field extracted from the H. 264 compressed video is first normalized in both temporal and spatial domain, and the MV fields of several continuous frames are accumulated to enhance the salient motion. Then the global motion compensation is performed on the accumulated MV field, while the fast statistical region growing algorithm is exploited to segment it into different motion-homogenous regions. Finally, the orientation histogram of the MV field is exploited to determine the moving object regions. Experimental results for several MPEG-4 test sequences demonstrate the proposed approach can segment moving objects from the H. 264 compressed domain in real time, and exhibit good segmentation quality.

Keywords moving object segmentation, H. 264, compressed domain, motion vector field

1 引 言

众所周知,运动对象分割是实现众多基于内容的多媒体应用,如视频索引与检索、智能视频监控、视频编辑等所必需的一个前提条件。自从 MPEG-4

提出基于内容的视频编码以来,有关运动对象分割的研究大多集中在像素域,而基于压缩域的运动对象分割直至近年来才开始引起人们关注。鉴于实际应用中大多数视频序列已经压缩为某种格式,由于直接在压缩域内进行运动对象分割,可免除对压缩视频进行完全解码,而且在压缩域内需要处理的数

基金项目:国家自然科学基金资助项目(60572127);上海市教委发展基金项目(05AZ43);上海市科委科技攻关项目(055115008);上海高校选拔培养优秀青年教师科研专项基金项目(2006)

收稿日期:2006-06-20;**改回日期:**2006-08-05

第一作者简介:刘志(1978~),男。2005年获上海交通大学模式识别与智能系统专业博士学位,现为上海大学通信学院讲师。主要研究方向为视频对象分割、实时对象跟踪、视频编码与多媒体通信。E-mail:liuzhi@staff.shu.edu.cn

据量要比像素域少很多,因此计算量大大减少;加之,从压缩视频中仅通过熵解码提取出的运动矢量和离散余弦变换(discrete cosine transform, DCT)系数,可直接作为分割所需的运动特征和纹理特征。由于从压缩域分割运动对象具有快速的特点,可解决传统的像素域分割难于满足实时分割的要求,因此更适用于具有实时性要求的应用场合。

近年来对压缩域内的运动对象分割方法已进行了初步探讨。例如, Sukmarg 等在 DCT 域首先采用自适应 K 均值算法将不同的块组合成匀质区域,然后按照区域的平均变化量分为运动区域或者背景区域^[1]。Mezaris 等和 Venkatesh Babu 等则只使用了运动信息来分割对象^[2,3],其中,前者首先采用迭代拒绝方案来选出属于对象的块,然后提出了一种宏块级的跟踪算法用来分割运动对象;而后者则首先采用时域累积和空域内插的方法估计一个稠密运动矢量场,然后采用期望最大化 EM(expectation maximization) 算法进行聚类,以获得不同的运动对象。然而,上述方法仍停留在传统的 MPEG-2 编码标准的压缩域。

H.264 是最新的视频编码标准^[4],相比于 MPEG-2,其编码效率提高了一倍,目前越来越多的应用都在转向采用 H.264 来取代 MPEG-2,但至今罕见在 H.264 压缩域内进行运动对象分割的研究成果报道。与 MPEG 域相比,H.264 中 I 帧的 DCT 系数不能直接作为用于分割的纹理特征,由于它们是在块的空间预测残差上进行变换的,而不是在原始块上进行变换,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 等提出了一种基于块的 MRF(Markov random field)模型从稀疏运动矢量场中分割运动对象的方法^[5],该方法先根据各个块运动矢量的幅值赋予各个块不同类型的标记,再通过最大化 MRF 的后验概率来标记出属于运动对象的块。但是,这种方法只适用于静态背景的视频序列,而且分割的准确度不高,计算量也较大。

本文针对 H.264 压缩域内的运动对象分割提出了一种新方法,它不但不受限于静止背景的视频序列,而且对于具有运动背景或者静止背景的视频序列,都能快速可靠地分割出运动对象。

2 分割算法

本文提出的 H.264 压缩域内的运动对象分割

方法包括如图 1 所示的 4 个步骤,即首先从 H.264 视频中提取出运动矢量场,并归一化为采样精度为 4×4 块的均匀运动矢量场;同时利用连续多帧的运动矢量场的迭代后向投影来获得更加可靠的累积运动矢量场;接着对累积运动矢量场进行全局运动估计,并在每个 4×4 块上计算累积运动矢量和估计得到的全局运动矢量之差,以作为各 4×4 块经全局运动补偿后的残差;然后采用统计区域生长算法将累积运动矢量场分割成多个具有相似运动的区域;最后利用运动矢量场的方向角直方图来判断出属于运动对象的分割区域,以获得运动对象。

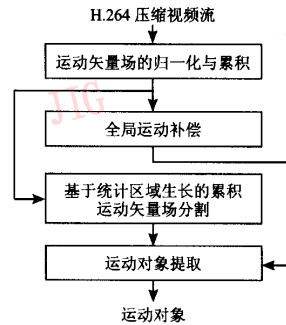


图 1 分割算法流程图

Fig. 1 Flowchart of the proposed segmentation algorithm

2.1 运动矢量场的归一化与累积

由于 H.264 通过引入多种新的编码工具来提高编码效率,一个宏块可被分割成从 4×4 到 16×16 共计 7 种不同尺寸的块,且由于每个块都具有一个单独的运动矢量和参考帧序号,因此从 H.264 压缩域直接提取出的运动矢量场与 MPEG 压缩域有所不同。从 MPEG 视频可以直接提取获得具有相同块尺寸的均匀运动矢量场,而从 H.264 视频直接提取得到的则是具有可变块尺寸的运动矢量场,而且运动矢量的精度为 $1/4$ pixels。

在使用运动矢量场进行对象分割之前,还需要对原始的运动矢量场进行时域和空域上的归一化,下面以某个 P 帧为例加以说明。假设当前帧 I_t 中任意块 B_i 的运动矢量为 $V_i(B_i)$,首先根据与参考帧 I_r (下角 r 代表 reference) 之间的时域距离和方向进行归一化,使归一化后运动矢量 $V_i^{\text{normal}}(B_i)$ 的等效参考帧为前一帧 I_{t-1} ,即

$$V_i^{\text{normal}}(B_i) = \frac{V_i(B_i)}{t-r} \quad (1)$$

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所覆盖的所有 4×4 块,这样就得到了一个在空域上采样精度为 4×4 块的均匀运动矢量场。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。

接下来通过对连续多帧的运动矢量场进行累积来获得更加可靠的累积运动矢量场,以便于分割运动对象。由于任一当前帧 I_i 的累积运动矢量场 V_i^A (上角 A 代表 accumulate,下同)利用了其后若干帧的运动矢量场,因此可先在相邻帧间进行运动矢量场的后向投影,然后再通过迭代累积后即可获得累积运动矢量场。通过相邻帧间运动矢量场的后向投影,可获得后一帧的归一化运动矢量场 V_{i+1}^{normal} 在前一帧的投影运动矢量场 V_i^P (上角 P 代表 project,下同),具体的后向投影方法可参见文献[6]。

假设当前帧 I_i 的运动累积过程总共利用了其后连续 m 帧的运动信息,累积过程从最后一帧 I_{i+m} 开始,即 $V_{i+m}^A = 0$,则可通过如下的迭代过程来获得当前帧的累积运动矢量场 V_i^A

$$V_{i,i}^A = V_{i,i+1}^A + V_{i,i}^P, \quad i = m-1, \dots, 1, 0 \quad (2)$$

其中,参与运动累积过程的帧数 m (即迭代次数)对最后获得的累积运动矢量场 V_i^A 有直接的影响,本文取值为 $m=3$ 。通过实验发现,如果 m 取值过小,那么对于运动平缓的视频序列,往往累积不到足够的可靠运动信息;而 m 取值过大,则由于每次相邻帧之间的运动矢量投影都存在一定的误差,因此多次迭代同样会造成误差的累积,进而导致最终累积运动矢量场的准确度降低。通过运动累积获得的运动矢量场 V_i^A 不仅增强了对对象运动以及摄像机全局运动这类可靠的运动信息,而且抑制了单帧运动矢量场通常存在的噪声。

2.2 全局运动补偿

在累积运动矢量场 V_i^A 上进行全局运动估计时,全局运动模型采用如下 6 参数的仿射运动模型,因为它能够描述常见的摄像机平移、倾斜和缩放等运动

$$\begin{aligned} \hat{x}_i &= a_0 x_i + a_1 y_i + a_2 \\ \hat{y}_i &= a_3 x_i + a_4 y_i + a_5 \end{aligned} \quad (3)$$

其中, $\mathbf{a} = [a_0, a_1, a_2, a_3, a_4, a_5]^T$ 是模型参数向量, (x_i, y_i) 为当前帧累积运动矢量场 V_i^A 中任意 4×4 块 B_i 的中心坐标,则利用上述全局运动模型估计所得的运动矢量为 $\mathbf{V}^E(x_i, y_i) = [\hat{x}_i - x_i, \hat{y}_i - y_i]^T$ (上角 E 代表 estimate,下同)。通过将如下定义的目标

函数 φ 最小化来求解全局运动模型参数

$$\varphi = \sum_i e_i = \sum_i \|\mathbf{V}^E(x_i, y_i) - \mathbf{V}_i^A(x_i, y_i)\|^2 \quad (4)$$

其中, e_i 为任意块 B_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^[7],即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),而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 方法^[8],以保证参数估计能够快速收敛。

全局运动模型参数确定后,即可计算出估计的全局运动矢量场 $\hat{\mathbf{V}}^E$,对 V_i^A 中的任意块 B_i 进行全局运动补偿后的残差为

$$l(x_i, y_i) = \|\mathbf{V}_i^A(x_i, y_i) - \hat{\mathbf{V}}^E(x_i, y_i)\| \quad (5)$$

残差的大小反映了该块与全局运动匹配的程度,残差越小,表明其属于背景的可能性越大。

2.3 累积运动矢量场分割

累积运动矢量场可以视作为具有两个分量的图像,两个分量图像分别对应于运动矢量场的水平和垂直分量。由于一般运动对象通常具有显著而且独立的运动,因此利用运动矢量的相似性就可以把运动矢量场分割为不同的运动一致性区域。本文通过对最近提出的用于彩色图像分割的统计区域生长算法^[9]做了适当的修改,然后将其引入到运动矢量场的分割。该算法的最大优点是在整个区域生长过程使用了静态合并次序,因此可以省去传统的区域生长算法中最为耗时的区域邻接图更新的计算过程,是一种相当快速的分割算法。对于本文的运动矢量场分割,4 邻域内任意相邻块组 B_i 和 B_k 的运动差异性度量定义如下:

$$\delta(i, k) = \|\mathbf{V}_i^A(x_i, y_i) - \mathbf{V}_k^A(x_k, y_k)\| \quad (6)$$

将运动矢量场的所有相邻块组按照运动差异性度量从小到大进行排序,这样在区域生长开始前就决定了区域生长过程中邻接块的合并次序,并从运动差异性度量最小的相邻块组的合并开始区域生长过程。在每次区域生长时,假设当前块组 B_i 和 B_k 分别属于相邻的两个区域 R_a 和 R_b (单个块被认为是面积为 1 的区域),则判断 R_a 和 R_b 是否合并的条件如下:

$$|\mathbf{V}_{i,x}^A(R_a) - \mathbf{V}_{i,x}^A(R_b)| < \Delta(R_a) + \Delta(R_b) \quad (7)$$

$$|\mathbf{V}_{i,y}^A(R_a) - \mathbf{V}_{i,y}^A(R_b)| < \Delta(R_a) + \Delta(R_b)$$

其中, $[\mathbf{V}_{i,x}^A(R_a), \mathbf{V}_{i,y}^A(R_a)]^T$ 是区域 R_a 的平均运动矢量,而 $[\mathbf{V}_{i,x}^A(R_b), \mathbf{V}_{i,y}^A(R_b)]^T$ 是区域 R_b 的平均运动矢量,上述两个不等式中的函数 $\Delta(\cdot)$ 定义为

$$\Delta(\mathbf{R}) = \frac{R_{\text{search}}^2}{2Q|\mathbf{R}|} (\min(R_{\text{search}}, |\mathbf{R}|) \log(1 + |\mathbf{R}|) + 2\log 6wh) \quad (8)$$

其中, R_{search} 表示运动矢量的动态范围, 对于 H.264 编码器通常采用的运动搜索范围为 $[-32, 32]$, R_{search} 取为 64。运动矢量场的尺寸为 $w \times h$, $|\mathbf{R}|$ 表示区域 \mathbf{R} 所包含的运动矢量数目。实际上, 只需要通过调整参数 Q 来控制运动矢量场的分割程度(在实验中 Q 取为 16), 就可以将运动矢量场适度地分割成若干具有相似运动的区域。

2.4 运动对象提取

结合前两小节的结果, 对于累积运动矢量场 \mathbf{V}_i^A , 可先计算其每个分割区域 \mathbf{R}_i 在全局运动补偿后的平均残差

$$\bar{E}_{\mathbf{R}_i} = \frac{\sum_{(x,y) \in \mathbf{R}_i} l(x,y)}{|\mathbf{R}_i|} \quad (9)$$

然后在面积大于 T_{area} 的若干分割区域中选择式(9)定义的平均残差最小的区域, 作为最可靠的参考背景区域, 标记为 \mathbf{R}_0 。其中, 参数 T_{area} 取为 $0.1wh$, 即只有覆盖了整个运动矢量场 10% 以上的区域才有可能被选作为参考背景区域。这是因为如果只按照式(9)定义的平均残差来选择背景区域, 就有可能将面积很小而平均残差恰巧也很小的噪声区域误选择为背景区域, 而设置这一限制区域面积的参数则可以避免上述问题。

在获得了参考背景区域后, 就可根据该区域的平均残差以及运动矢量的方向角直方图来逐一判断其他分割区域是运动对象区域, 还是背景区域。运动矢量的方向角在 $[0^\circ, 360^\circ]$ 之间均匀量化为 16 个级别, 即每隔 22.5° 一个量化级。然后通过分别计算来得到各个分割区域 \mathbf{R}_i 所覆盖的运动矢量场的方向角直方图 $H_i (i=0, 1, \dots, N)$, 并进行直方图归一化。以区域 \mathbf{R}_0 的平均残差和方向角直方图 H_0 为参考进行判定, 如果某个分割区域 $\mathbf{R}_i (i=1, \dots, N)$ 同时满足式(10)、式(11)两个条件, 则认为 \mathbf{R}_i 是属于背景的区域, 否则判断为运动对象区域。

$$\bar{E}_{\mathbf{R}_i} < \alpha \bar{E}_{\mathbf{R}_0} \quad (10)$$

$$1 - \frac{\|H_i - H_0\|}{2} > T_{\text{histogram}} \quad (11)$$

其中, 参数 α 和 $T_{\text{histogram}}$ 根据实验确定为 3.0 和 0.75, 以便能够确保正确判断出背景区域。对于一般视频序列所表现出的有规律的摄像机运动, 属于

背景的任意区域的平均残差接近于 \mathbf{R}_0 的平均残差, 而且方向角直方图 H_i 也与 H_0 十分相似, 通常表现为单峰性, 并集中在相邻的 1~2 个量化级中。因此, 属于背景的分割区域能够符合上述 2 个相对宽松的条件, 而属于运动对象的区域则至少不能满足上述 2 个条件之一。这样将所有属于运动对象的区域组合起来即可得到最终的运动对象分割结果。

3 实验结果与分析

本文在使用 JM8.6 版本的 H.264 编码器对标准测试序列进行编码后, 将其作为测试用的 H.264 压缩视频流。H.264 编码器的配置如下: Baseline Profile, IPPP, 每 30 帧插入 1 个 I 帧, 3 个参考帧, 运动估计的搜索范围为 $[-32, 32]$, 量化参数为 30。在实验中, 先每隔 3 帧(运动矢量场累积过程中使用的帧数) 计算一次累积运动矢量场, 然后在累积运动矢量场上通过运动对象分割来测试本文所提出算法的分割性能。

图 2 所示为测试序列 Coastguard (CIF 格式, 352×288) 的运动对象的分割结果。由于该序列具有明显的全局运动, 因此摄像机首先通过自右向左的平移来跟踪画面中间的小船, 然后通过自左向右的运动来跟踪画面左边出现的大船。图 2(a) 所示为序列 Coastguard 的第 4 帧图像, 其相应的累积运动矢量场如图 2(b) 所示(为了显示清楚进行了重新抽样)。对应于图 2(b), 利用仿射运动模型估计得到的全局运动矢量以及经过全局运动补偿后的残差运动矢量分别如图 2(c) 和图 2(d) 所示。图 2(e) 所示为累积运动矢量场的分割结果, 其中白色区域为最可靠的参考背景区域, 最终提取出的两个运动对象如图 2(f) 所示。

用文献[2]中提出的 MPEG-2 域内的运动对象分割算法与本文算法在序列 Coastguard 上进行了测试, 均同样只使用运动矢量信息来分割对象。图 3(a) 和图 3(b) 所示为序列 Coastguard 的第 37 帧和第 250 帧图像, 文献[2]中的分割结果如图 3(c) 和(d) 所示, 而本文分割算法的分割结果如图 3(e) 和(f) 所示。通过对比可以看出, 本文的分割算法更为准确地分割出了两个运动对象, 本文的分割结果完整地包括了大船的桅杆和旗帜, 并未附带小船激起的浪花, 在主观视觉效果上要优于文献[2]中的分割算法。

本文的所有实验都是在一台 CPU 为 3.0GHz, 内存为 512M 的计算机上进行, 对于 CIF 格式的视

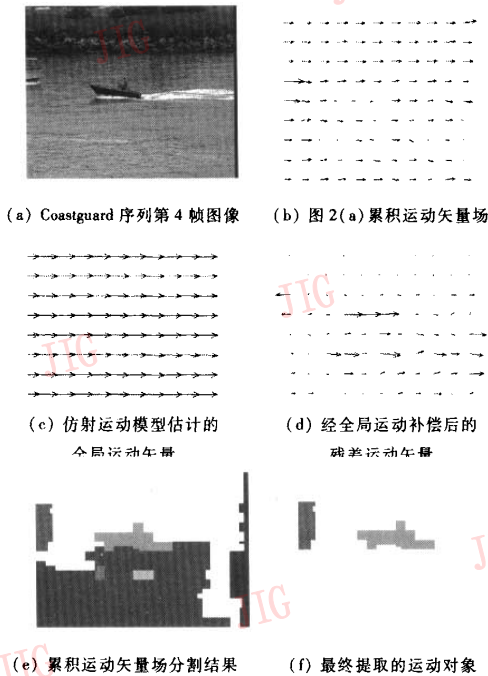


图 2 序列 Coastguard 的运动对象分割结果
Fig. 2 Segmentation results for the sequence Coastguard

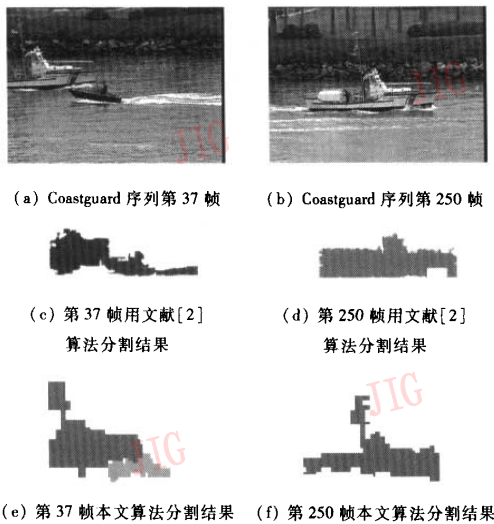


图 3 本文与文献[2]的分割算法的分割结果比较
Fig. 3 The segmentation comparison of two algorithms

频序列,平均每帧的处理时间约为 32ms,已能满足大多数实时应用 25fps 的要求。另外还用测试序列 Mobile 进行了实验,也取得类似的结果,因篇幅所

限,这里不再详述。

4 结 论

本文基于 H. 264 压缩域提取出的运动矢量场,提出了一种实时运动对象分割算法。该算法先通过运动矢量场累积来增强可靠的运动信息,然后在累积运动矢量场上进行全局运动补偿,同时利用基于统计区域生长算法对其进行分割,最后利用运动矢量场的方向角直方图和分割区域的平均残差来提取出运动对象。实验结果表明,本文的运动对象分割方法可适用于具有实时要求的应用领域,并具有良好的分割质量。由于本文的方法只使用了运动矢量场信息,因此也可将其应用于 MPEG 压缩域的运动对象分割。

参考文献 (References)

- 1 Sukmarg O, Rao K R. Fast object detection and segmentation in MPEG compressed domain [A]. In: Proceedings of IEEE TENCEN (IEEE Region 10 International Conference) [C], Kuala Lumpur, Malaysia, 2000, 3: 364 ~ 368.
- 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 Venkatesh Babu R, Ramakrishnan K R, Srinivasan S H. Video object segmentation: A compressed domain approach [J]. IEEE Transactions on Circuits and System for Video Technology, 2004, 14(4): 462 ~ 474.
- 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 System 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 Z, Shen L Q, Zhang Z Y. Automatic video object segmentation from MPEG compressed domain [A]. In: Proceedings of. 7th International Workshop on Image Analysis for Multimedia Interactive Services [C], 2006: 333 ~ 336.
- 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.
- 8 Press W H, Flannery B P, Teukolsky S A, et al. Numerical Recipes in C: The Art of Scientific Computing [M]. Cambridge. UK: Cambridge University Press, 1992.
- 9 Nock R, Nielsen F. Semi-supervised statistical region refinement for color image segmentation [J]. Pattern Recognition, 2005, 38(6): 835 ~ 846.