结合多种语义信息的半自动视频对象分割

陈韩锋,戚飞虎

(上海交通大学计算机科学与工程系,上海 200030)

摘 要: 本文提出了一种半自动视频对象分割方法.该方法结合了多种视频语义信息,来提高分割的效率和分割方法的通用性.在视频的初始帧中用半自动绘制的多边形初始化目标对象区域;然后在后续帧中自动跟踪对象区域,对于一般性对象采用基于后向块匹配的像素对应方法进行跟踪;针对平移运动的刚体对象和变化很缓慢的对象本文提出了专门的跟踪方法;最后利用一种基于同等组的模板修正方法来修正每一帧的分割模板.利用本文的方法获得了精确稳定的实验结果.

关键词: 视频对象; 半自动分割; 跟踪; 一致区域

中图分类号: TP391 **文献标识码:** A **文章编号:** 0372-2112 (2002) 12A-2012-04

Semiautomatic Video Object Segmentation by Integrating Various Semantic Information

CHEN Han-feng, QI Fei-hu

(*Dept. of Computer Science & Engineer, Shanghai Jiao Tong University, Shanghai* 200030, *China*)

Abstract: This paper proposes a novel semiautomatic method for semantic video object extraction. The proposed method aims at improving the semiautomatic segmentation by integrating various semantic properties of the video. Object region in the first frame is extracted by a user input polygon. Then, a tracking scheme based on backward block-matching is used to automatically extract object region in the rest of frames for general objects. Translational rigid objects and slowly changing objects are specially considered in the tracking. Finally, a mask refinement algorithm based on peer group is applied to extract accurate object boundary and insure reliable tracking. Accurate and consistent results are obtained in the experiments by the proposed method.

Key words: video object; semiautomatic segmentation; tracking; homogeneous region

1 引言

视频对象分割是指从一段视频序列中分割出用户感兴趣的对象区域,这项技术对实现新一代视频编码标准 MPEG-4 有着重要意义.目前的视频对象分割方法可以大致分为两类:自动分割和半自动分割.

自动的视频对象分割在分割过程中不需要人的干预,但是这一类方法有很多先天性的缺点.首先,这类方法只能分割一些很简单的语义对象;其次,目前的计算机智能还不足以自动确定那些由于目标和背景具有很相似的颜色和纹理而带来的不确定的边界;此外,若摄像机是运动的,那么区分目标的局部运动和摄像机带来的全局运动也是一个难题.

视频对象的半自动分割方法通过人机交互可以弥补计算机在语义理解上的缺陷,随意地定义视频中所需的分割对象,也不会受摄像机运动的影响;那些不确定的目标边界也能够通过人机交互来确定.尽管半自动方法不能满足实时的分割需求,但是在许多应用中,如视频合成、视频索引以及视频检索等,并不需要实时的分割.因此,本文将主要研究半自动的视频对象分割方法.

目前的半自动视频对象分割方法基本上都包含三个主要部分:半自动的初始帧分割、后续帧中目标区域的自动跟踪和分割模板自动修正.分割模板是一个与视频图像具有相同宽高的矩阵,矩阵的每个元素用二值符号(如 0 和 1)来表示当前帧中对应位置像素是背景像素还是所需目标的像素.

在初始帧分割中,借助人机交互来确定目标区域的初始位置和形状.目前的初始帧分割方法[1,3,4]大都进行两个步骤:首先用户用鼠标在待分割目标的边界附近描出一个近似的目标轮廓(如图 2 中的虚线轮廓),然后利用形态学-分水岭算法[4]重新确认近似轮廓附近的像素是背景像素还是目标像素.如果待分割目标的形状比较复杂,那么要用鼠标描绘这个近似轮廓,人机交互的工作量很大.

目标区域跟踪就是在视频的后续帧中跟踪初始帧分割时定义的目标区域.最普遍的跟踪方法是为整个目标区域建立参数运动模型[1],通过运动模型将当前帧中的目标区域映射到下一帧中得到新的目标区域.问题在于如果目标对象是非刚体的运动,就很难用一个简单的参数运动模型来描述目标区域在前后帧之间的映射关系.在许多文献[1~4]中,非刚体都被近似为刚体来处理:用比较简单的刚体运动模型来描述非刚体,然后通过分割模板修正来校正边缘像素.但是,如果目标运动较快,这种近似会很不可靠.

针对以往这些方法中存在的问题,本文提出了一种新的半自动视频对象分割方法.该方法主要包括四个部分(如图 1):语义定义、半自动的初始帧分割、自动的目标区域跟踪和分割模板修正.在分割开始的时候,用户首先进行语义定义.

收稿日期:2002-06-10;修回日期:2002-10-15
基金项目:国家自然科学基金(No.60072029)

提供有关视频的一些语义特性信息;再在初始帧中手动绘制一个多边形,用以定义待分割目标区域,在绘制多边形之前,先对图像进行自动的一致区域分割,以减少多边形顶点数;然后在后续帧中自动跟踪目标区域,对于一般性的目标,提出了一种基于像素而不是整个区域的目标跟踪方法,避免了复杂的运动模型,而针对平移运动的刚体则采用二参数平移运动模型,以简化跟踪过程,此外,本文还提出了自适应的跟踪步长,来提高变化缓慢的目标的跟踪效率;最后,每个跟踪得到的目标区域分割模板都被自动修正,本文采用了一种基于同等组概念的分割模板修正方法.

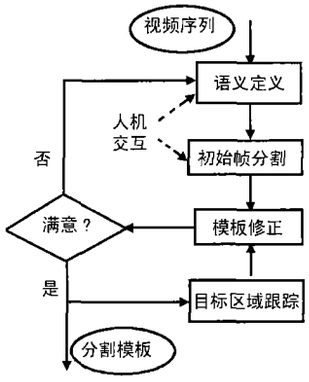


图 1　本文方法的框图

2　语义定义

所谓语义定义是指由用户提供一些视频场景和待分割目标的语义特性信息.利用这些语义特性信息可以大大的提高分割的准确性、稳定性和分割的速度.因为根据不同的语义特性,可以在分割过程的每个阶段采取相应的处理方法,以很好地利用场景和目标本身的一些特点.

在本文的方法中,主要关注三种语义信息.第一种是场景复杂度信息,我们定义了两个级别:平滑和纹理丰富.对于平滑场景,在初始帧分割时将自动采用颜色一致作为标准来进行一致区域分割;对于纹理丰富的场景,那么在初始帧分割时将自动采用纹理一致作为标准来进行一致区域分割.第二种是待分割目标的形态信息,即是一般性物体还是作平移运动的刚体.一般性物体的形状和位置都随着时间的推移而变化,而作平移运动的刚体的形状不会改变,只是位置发生变化.针对这两类物体本文提出了不同的目标区域跟踪方法.第三种信息是目标运动速度的信息,对于运动或变化很慢的目标,本文提出了一种自适应跟踪步长的方法来提高跟踪的速度和稳定性.我们定义了两种选择,由用户来决定是否启用自适应跟踪步长.这几种语义特性的定义均设计成简单选择性的,因此只需很少量的人机交互.

3　初始帧分割

初始帧分割就是在初始帧中提取待分割的目标区域.令 $\boldsymbol{I}_1$ 为视频序列的初始帧,初始帧分割的第一步是对 $\boldsymbol{I}_1$ 进行一致区域分割.一致区域分割的目的是将图像分割为一些颜色或纹理一致的区域(如图 2),从而达到简化图像的目的.根据用户在语义定义中对场景复杂度的估计,计算机自动选择颜色一致区域分割或纹理一致区域分割.在本文中,颜色一致区域分割采用一种基于均值漂移算法的图像分割方法[5],这种方法的特点是对于图像中的噪声和一些细微的沟壑纹理不敏感,不易出现过分割.纹理一致区域分割则采用 Deng Y.N. 等人提出的基于 J 值度量的纹理分割方法[6],这种纹理分割方法可以比较准确地确定纹理区域的边界.限于篇幅,这两种方法的具体内容将不在本文中详细介绍.一致区域分割是一个自动的过程,无需人工操作.如果目标与背景具有很相似的颜色或纹理,那么在分割结果中可能会有一些目标与背景之间的边界不能被发现,这就需要在分割结果中通过人机交互(如鼠标)来"剪开"这些被合并的区域(如图 2),一般来说这种被合并的边界是很少的.

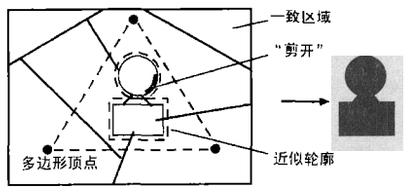


图 2　初始帧分割

一致区域分割完成后,用户需要绘制一个多边形来提取待分割目标区域.这个多边形(如图 2 中的虚线三角形)的边要求穿越了所有与待分割目标区域相邻的一致区域.由于一致区域分割已经将初始帧分割成了一些一致区域的集合,因此只需要一个顶点很少的多边形就能满足上述要求,而且这些顶点不必要紧邻着目标区域边界,这就简化了人机交互.然后,这个多边形的各条边(与图像边界重合的边除外)沿着一致区域自动收缩,直到两个一致区域之间的边界,收缩的结果就是初始帧中待分割的目标区域(记为 $\boldsymbol{R}_1$),用分割模板 $\boldsymbol{M}_1$ 来标记目标区域和背景区域.最后,利用本文第 5 节将要介绍的模板修正方法对 $\boldsymbol{M}_1$ 进行修正.

4　目标区域跟踪

在初始帧中得到目标区域的初始位置和形状后,开始在后续帧中自动跟踪目标区域.所谓的跟踪通常是指,根据前一帧的目标区域分割模板 $\boldsymbol{M}_{i-1}$ 得到当前帧的分割模板 $\boldsymbol{M}_i$.

4.1　一般性物体的跟踪

对于一般性物体,其形状和位置都随着时间的推移而随意变化,因此不适宜为其建立一个全局一致的运动模型.本文采用了化整为零的办法,对每个像素分别进行对应跟踪.

设 $\boldsymbol{R}_{k-1}$ 和 $\boldsymbol{M}_{k-1}$ 分别为前一个帧 $\boldsymbol{I}_{k-1}$ 的目标区域和分割模板,$\boldsymbol{I}_k$ 为当前帧.由于目标总是连续运动和变化的,这就使得 $\boldsymbol{R}_{k-1}$ 和 $\boldsymbol{R}_k$ 在位置和形状上都比较接近,因此在跟踪的时候不需要对所有的像素都进行后向对应跟踪的计算,只需要考虑"外跟踪区域"内的像素.$\boldsymbol{I}_k$ 的"外跟踪区域" $\boldsymbol{T}_k$ 定义为像素坐标位于 $\boldsymbol{R}_{k-1}$ 边缘附近的区域.相应地,$\boldsymbol{I}_k$ 的"内跟踪区域" $\boldsymbol{T}'_k$ 定义为 $\boldsymbol{R}_{k-1}$ 中除去包含在 $\boldsymbol{T}_k$ 中那部分以外剩下的区域.$\boldsymbol{T}_k$ 可以通过对 $\boldsymbol{R}_{k-1}$ 的边缘进行 t 次形态学的"膨胀"操作得到.t 的值可以根据目标运动和变化的速度在语义定义中预先设定,目标运动速度越快 t 的值越大,对于通常的运动来说,4 次"膨胀"已经足够(本文的实验中 $t=4$);t 越小跟踪速度就越快,因为所需考虑的像素点越少.

确定了"外跟踪区域" $\boldsymbol{T}_k$ 之后,为 $\boldsymbol{I}_k$ 中所有像素坐标位于 $\boldsymbol{T}_k$ 内的像素在 $\boldsymbol{I}_{k-1}$ 中寻找匹配点,每个像素在前一帧中的匹配点根据后向块匹配方法[9]得到.设 $p_k(i,j)$ 为 $\boldsymbol{I}_k$ 中一个落在 $\boldsymbol{T}_k$ 内的像素,$\boldsymbol{B}_k(i,j)$ 是以 $p_k(i,j)$ 为中心的 7×7 块.若 $\boldsymbol{B}_{k-1}(m,n)$ 是 $\boldsymbol{B}_k(i,j)$ 在 $\boldsymbol{I}_{k-1}$ 中的最佳匹配块,那么 $p_{k-1}(m,n)$ 就被认为是 $p_k(i,j)$ 的匹配点.本文中的块匹配准则采用

最小绝对差值之和[9]，搜索最佳匹配块时的搜索半径 W 的大小与 t 相同.

有了对应点，即可确定 I_k 中目标区域 R_k. 若 $p_{k-1}(m,n)$ 落在 R_{k-1} 内，那么就认为 $p_k(i,j)$ 属于 R_k，否则 $p_k(i,j)$ 属于背景. 此外，整个“内跟踪区域” T'_k 都被无条件地认为属于 R_k，这主要是考虑到 R_{k-1} 和 R_k 在位置和形状上比较接近. 最后用分割模板 M_k 来标记 R_k，并用第 5 节的方法修正模板.

4.2 平移运动的刚体的跟踪

作平移运动的刚体是现实中很常见的一类物体，它们的形状不随时间的推移而改变，而且具有简单的二参数平移运动模型. 利用这些特点，我们通过为目标区域建立一个二参数平移运动模型来简化它们的跟踪过程. 区域 R_{k-1} 的二参数平移运动模型可以描述为：

$$\forall\, p_{k-1}(i,j)\in R_{k-1},\ p_{k-1}(i,j)=p_k(i+d_1,j+d_2) \quad (1)$$

参数 d_1 和 d_2 通过“内跟踪区域” T'_k 的前向区域匹配来估计，匹配准则也采用最小绝对差值之和，最佳匹配搜索半径的大小也为 W. 得到 d_1 和 d_2 后，R_k 就可以通过对 R_{k-1} 平移一个矢量 (d_1,d_2) 后得到. 同样，最后需要对 M_k 进行修正.

需要指出的是，并不是整个 R_{k-1} 区域，而只是 T'_k 部分参与了前向区域匹配，这是考虑到与 I_{k-1} 相比，在 I_k 中目标区域的部分边缘区域可能会被遮挡或离开场景范围，从而影响区域匹配的可靠性. 此外，既然 4.1 节的方法可以适用于一般性物体的跟踪，当然也可以跟踪平移运动的刚体. 但是针对平移运动的刚体特殊的形态和运动模型，利用本小节的方法可以使跟踪更加快速和稳定.

4.3 自适应跟踪步长

前面提到，目标区域跟踪的时候，根据当前帧的目标区域分割模板 M_k 可以得到下一帧的 M_{k+1}. 事实上，从 M_k 也可以跟踪得到 M_{k+l}，l 称为跟踪步长. 在目前的文献中 l 大都被固定为 1，也就是逐帧往下跟踪. 但是如果目标运动和变化非常缓慢，达到子像素级，即经过连续很多帧，目标的累计运动和变化才达到一个像素的距离，那么 $l=1$ 并不是一个好的选择. 这是因为子像素级的运动不易在连续两帧之间被发现，这就会导致跟踪误差的不断积累；而且从效率上来说，如果 $l>1$，跟踪速度将得到提高. 针对这一点，本文提出了自适应的跟踪步长来跟踪运动非常缓慢的目标区域.

假设 I_k 中的目标区域 R_k 已经得到，R'_{k+l} 为当前的 l 值下利用 4.1 或 4.2 节的方法跟踪得到的 I_{k+l} 中的目标区域. 为了确定是否接受这个 R'_{k+l} 为 I_{k+l} 中真正的目标区域 R_{k+l}，需要引入“跟踪误差” E_{kl}. E_{kl} 定义为：

$$E_{kl}=\frac{1}{N_T}\sum_{p_{k+l}(m,n)\in T}|p_k(i,j)-p_{k+l}(m,n)| \quad (2)$$

上式中，$p_k(i,j)$ 是 $p_{k+l}(m,n)$ 的对应像素，这种对应关系可以由 4.1 节中的块匹配或 4.2 节中的区域匹配来确定. 如果利用 4.1 节的跟踪方法，T 就取“外跟踪区域”与 R'_{k+l} 的交集；如果利用 4.2 节的跟踪方法，T 就取“内跟踪区域”. N_T 是 T 中的像素数. 确定是否接受 R'_{k+l} 为 I_{k+l} 中的真正的目标区域 R_{k+l} 的过程也就是确定 l 值的过程（如图 3），描述如下三步：(1)在当前的 k 值和 l 值下，计算 R'_{k+l} 和 E_{kl}；(2)若 $E_{kl}>T_1$ 且 $l>1$，则放弃 R'_{k+l}，令 $l=0.5l$，转第(1)步；(3)否则，接受 R'_{k+l} 为 I_{k+l} 中的目标区域 R_{k+l}，修正模板 M_{k+l}，令 $k=k+l$，然后令 $l=2l$，转第(1)步. T_1 为一个阈值（该阈值的取值范围较大，本文中 $T_1=10$）.

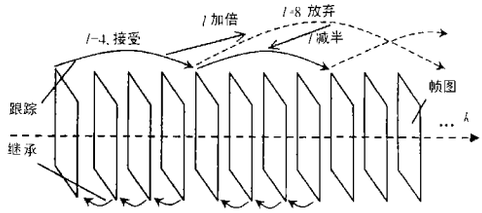


图 3 自适应跟踪步长

在分割过程中，一旦确定了 M_{k+l} 以后，就可以很简单地估计 $M_i\ (k<i<k+l)$：令 $M_i=M_{i-1}$（称之为继承），然后对 M_i 进行模板修正. 由于不需要计算从 M_{i-1} 到 $M_i\ (k<i<k+l)$ 的匹配关系，可以提高整个视频的分割速度，而且 l 的值越大，分割的速度就越快. 启用自适应跟踪步长时，l 可初始化为 2. 对于子像素级运动的目标，参数 t 取最小值即可. 因此在分割过程中，$l>1$ 时 t 将自动设置为 1.

5 分割模板修正

由于噪声、遮挡问题、块匹配模型或平移运动模型的假设，在初始帧分割和目标区域跟踪时得到的分割模板中可能会有少量的错误，即把一些背景像素标记为目标像素，而把一些目标像素标记为背景像素. 由于相邻两帧非常相似，因此这些标记错误的像素数量很少而且大都集中在目标区域的边缘附近. 尽管这些被标记错误的像素很少，但是不能忽略，因为目标区域的跟踪是个串行的过程，前面的误差会在后续帧中积累起来，影响整个方法的稳定性和可靠性. 本文采用了一种基于同等组[7]（Peer Group）的模板修正方法，利用同等组来强化局部颜色信息对目标区域边界的约束.

设 $x_0(i,j)$ 为当前帧中的一个像素，将以该像素为中心的一个 $N\times N$（本文中 $N=7$）大小窗口内的所有像素按照它们与 $x_0(i,j)$ 的距离从小到大进行排序，并且标记为 $x_k(i,j)$，$k=1,2,\cdots,N^2-1$，也就是说，

$$d_k(i,j)=\|x_0(i,j)-x_k(i,j)\|,\ k=0,1,2,\cdots,N^2-1 \text{ 且}$$
$$d_0(i,j)\leqslant d_1(i,j)\leqslant\cdots\leqslant d_{N^2-1}(i,j) \quad (3)$$

这里的距离定义为 RGB 颜色空间中的欧拉距离，它标明了两个像素的颜色相似度. 于是像素 $x_0(i,j)$ 的大小为 s 的同等组就可以定义为：

$$\boldsymbol{P}(i,j)=\{x_k(i,j),k=0,1,\cdots,s-1\} \quad (4)$$

$\boldsymbol{P}(i,j)$ 包含了像素 $x_0(i,j)$ 本身以及以它为中心的 $N\times N$ 大小窗口内与它最相似的 $s-1$ 个像素，s 的值可以自动确定[7].

模板修正过程考察当前目标边界的外侧像素和内侧像素，重新确认这些像素应该标记为目标像素还是背景像素. 目标边界的外侧像素是指与当前目标区域边界相邻但是标记为属于背景区域的像素，所有外侧像素的集合记为 C_{out}；而目标边界的内侧像素是指与当前目标区域边界相邻而且标记为属于目标区域的像素，所有内侧像素的集合记为 C_{in}.

设 a 是一个内侧像素，$\boldsymbol{P}(a)$ 是它的同等组（包含 s 个像

素).ρ 是一个在 0 和 1 区间内的值,定义为

$$\rho = S_o / s \tag{5}$$

S_o 是 $\boldsymbol{P}(a)$中已经被标记为目标区域的像素数.于是根据以下准则重新确定分割模板中像素 a 的标记:

$$mask(a) = \begin{cases} 1 , & \text{if} \quad \rho \geqslant T_2 \\ 0 , & \text{if} \quad \rho \leqslant T_2 \end{cases} \tag{6}$$

上式中,$mask(a)$为分割模板中的标记值,$mask(a)$等于 1 表示像素 a 属于目标像素,等于 0 表示像素 a 属于背景像素.T_2 是一个阈值,本文的实验中等于 0.4.对于目标边界的外侧像素,可以用相同的方法重新确定它们是属于目标区域还是背景区域,唯一不同的是将 T_2 的值改为 $1 - T_2$.

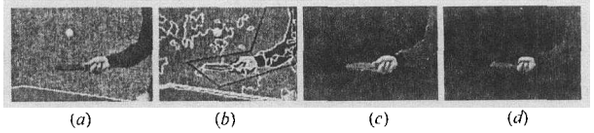
因此,模板修正的过程可以归结为:(1)对 $\boldsymbol{C}_{\text{in}}$中的每个像素,根据式(6)重新确定分割模板中相应位置的标记;然后更新 $\boldsymbol{C}_{\text{in}}$和 $\boldsymbol{C}_{\text{out}}$;(2)对 $\boldsymbol{C}_{\text{out}}$中的每个像素,根据式(6)重新确定分割模板中相应位置的标记;然后更新 $\boldsymbol{C}_{\text{out}}$.

6　实验结果

利用本文提出的半自动视频对象分割方法,我们对三段 MPEG-4 的测试视频序列进行了实验.这三个视频序列是"Mother-Daughter"、"Tennis"和"Container",他们覆盖了多种类型的场景和目标.图 4 - 6 给出了部分的实验结果,表 1 是分割每段视频时的部分语义定义信息.

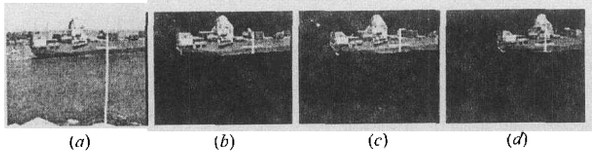
(a)　(b)　(c)　(d)

图 4　"*Mother-daughter*"序列.(a)第 1 帧;(b)第 1 帧的颜色一致区域分割结果及提取目标的多边形;(c)第 1 帧分割结果;(d)第 70 帧分割结果



(a)　(b)　(c)　(d)

图 5　"*Tennis*"序列.(a)第 1 帧;(b)第 1 帧的纹理一致区域分割结果及提取目标的多边形;(c)第 1 帧分割结果;(d)第 30 帧分割结果



(a)　(b)　(c)　(d)

图 6　"*Container*"序列.(a)第 1 帧;(b)第 1 帧分割结果;(c)第 100 帧分割结果;(d)第 300 帧分割结果

由图 4 和 5 不难看出,由于对初始帧进行了自动的一致区域分割,只需一个顶点很少的多边形就能在初始帧中提取出目标区域.而且本文的方法还适用于不同场景下的视频对象分割.在给出的这些结果中,只在语义定义和初始帧分割的时候进行了人机交互,后续帧的分割全是自动进行的,这说明了本文的跟踪方法比较稳定.

表 1　语义特性信息

	Mother-Daughter	Tennis	Container
一致区域分割	颜色一致	纹理一致	颜色一致
跟踪方法	一般性物体	一般性物体	平移刚体
跟踪步长	1	1	自适应

7　结束语

视频对象分割是视频应用研究中一个很棘手的问题.本文提出了一种新的而且比较通用的半自动视频对象分割方法.该方法通过结合多种简单的语义信息,来提高分割的效率和分割方法的通用性.实验结果显示,本文的跟踪方法可以适用于多种背景和目标.

在研究中我们发现模板修正对于整个分割过程的稳定性十分重要,在将来的研究中我们将进一步简化修正算法,以提高跟踪效率.此外,本文并没有考虑多个目标之间的遮挡和显露,这一问题也将在后续的工作中研究.

参考文献:

[1] Ju Guo, Jongwon Kim, Kuo C C J. An interactive object segmentation system for MPEG video[A]. Proceedings of ICIP 99[C]. ICIP, 1999.

[2] Dong Kwon Park, Ho Seek Yoon, Chee Sun Won. Fast object tracking in digital video[J]. IEEE Trans. on Consumer Electronics, 2000, 46(3):785 - 790.

[3] Grinias I, Tziritas G. A semi-automatic seeded region growing algorithm for video object localization and tracking[J]. Signal Processing: Image Communication, 2001.

[4] Chuang Gu, Ming-Chieh Lee. Semiautomatic segmentation and tracking of semantic video objects[J]. IEEE Trans. on Circuits and Systems for Video Technology, 1998, 8(5):572 - 584.

[5] Comaniciu D, Ramesh V, Meer P. The variable bandwidth mean shift and data-driven scale selection [A]. Proceedings of ICCV01 [C]. ICCV, 2001.

[6] Yining Deng, Manjunath B S. Unsupervised segmentation of color-texture regions in images and video[J]. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2001, 23(8):800 - 810.

[7] Yining Deng, Kenney C, Moore M S, Manjunath B S. Peer group filtering and perceptual color image quantization[A]. Proceedings of ISCAS99[C]. ISCAS, 1999.4:21 - 24.

作者简介:

陈韩锋　男,1976 年生于浙江义乌,上海交通大学计算机系博士研究生,主要研究方向为图像分割,视频分割,运动分析和目标跟踪.

戚飞虎　男,1938 年生于浙江余姚,上海交通大学计算机系博士生导师,主要研究方向为图像处理,模式识别和视频监控等.