

# Adaboost 目标跟踪算法<sup>\*</sup>

贾静平<sup>1</sup> 张飞舟<sup>1</sup> 柴艳妹<sup>2</sup>

<sup>1</sup> (北京大学 地球与空间科学学院 遥感与地理信息系统研究所 北京 100871)

<sup>2</sup> (清华大学 计算机科学与技术系 北京 100084)

**摘 要** 从两类模式分类技术的角度看待视频序列中的目标跟踪问题, 提出一种基于 Adaboost 学习技术的跟踪算法. 首先利用像素 RGB 分量的整数系数的线性组合生成备选目标特征, 以类间方差比为标准选出较好的特征来设计若干弱分类器, 进而构造出一个强分类器. 然后对于每帧输入图像, 按照该强分类器对各像素进行分类, 生成目标概率分布图. 再通过结合信任域算法和尺度空间理论实现对分布图中的灰度块定位, 从而完成目标跟踪. 此外还通过在线集成新的弱分类器和对已有弱分类器权值的更新, 提高算法对光照变化等因素引起的特征变化的适应能力. 在大量真实序列图像上进行实验验证, 并与现有算法进行比较, 结果表明该算法不但能更好地应对目标特征变化, 在存在干扰的背景中稳定跟踪目标, 也能更准确地描述目标大小, 显著提高跟踪算法精度.

**关键词** Adaboost 分类器, 图像跟踪, 序列图像分析

中图法分类号 TP 391

## Adaboost Object Tracking Algorithm

JIA Jing-Ping, ZHANG Fei-Zhou, CHAI Yan-Mei

<sup>1</sup> ( Institute of Remote Sensing and Geographic Information System, School of Earth and Space Sciences  
Peking University, Beijing 100871 )

<sup>2</sup> ( Department of Computer Science and Technology, Tsinghua University, Beijing 100084 )

### ABSTRACT

An Adaboost based algorithm for object tracking in image sequences is proposed. In this algorithm, tracking is considered as a binary classification problem. Firstly, the linear combination of R, G and B with integer coefficients is used to generate the candidate features. Features are selected for the design of weak classifiers according to the two class variance ratio. Then, a strong classifier is built on the weak classifiers. For each incoming frame, a likelihood image of the object is created according to the classification results of pixels by the strong classifier. The trust region method and the scale space theory are employed to locate the blobs in the likelihood image, and thus the object tracking is fulfilled. The changes of illumination often cause the changes of features. The adaptability of the proposed algorithm is improved by online integration of new weak classifiers and automated weights update of the used ones. Based on the tracking results of sequence examples, the proposed algorithm can adapt to feature changes, track object in cluttered background and describe the object accurately with better tracking precision.

**Key Words** Adaboost Classifier, Image Tracking, Image Sequence Analysis

<sup>\*</sup> 国家 863 计划资助项目 (No. 2009AA12Z352)

收稿日期: 2007-12-29 修回日期: 2008-07-17

作者简介: 贾静平, 男, 1978 年生, 博士, 主要研究方向为机器视觉、模式识别. E-mail: jiajiping@163.com 张飞舟, 男, 1965 年生, 副教授, 主要研究方向为导航、制导与控制. 柴艳妹, 女, 1978 年生, 博士, 主要研究方向为机器视觉、模式识别.

## 1 引言

目标跟踪是计算机视觉的一个重要分支,也是目前的研究热点.传统的跟踪算法大致可分为三类:1)在每帧中进行目标检测,对检测到的目标进行轨迹关联实现跟踪;2)通过定义目标区域像素的灰度值模板,在随后各帧中进行模板匹配后跟踪;3)获得目标区域像素的颜色或梯度直方图,使用 Mean Shift方法进行跟踪.

Tomasi<sup>[1]</sup>指出好的目标特征和好的跟踪算法同等重要.对特征而言,最重要的是能从背景中区分出目标. Hart<sup>[2]</sup>使用 PCA变换方法,从预先定义的 RGB和 rgb通道的子集所构成的特征集合中,选出最好的特征用以跟踪. Slem和 Efros<sup>[3]</sup>则从 5个特征空间中选取最好特征,并在其中切换来提高跟踪性能.这些算法的局限性在于备选特征数量有限. Avidan<sup>[4]</sup>提出使用 SVM的算法,但 SVM的计算量限制了算法的应用.针对这些问题,本文从一个新的角度来看待目标跟踪问题,将它视为一个两类模式分类问题加以解决.提出使用 Adaboos特征分类器来将目标从背景中区分出来.首先,为参考图像中每一个像素构造特征向量,然后训练一个分类器将属于目标的像素和属于背景的像素划分开来.给定一个新的视频帧,可使用这个特征分类器测试各个像素,生成一个概率分布图,概率分布图中的灰度块就是目标移动到的位置.可利用循环优化算法来定位它.

能否准确决定目标的大小直接影响着跟踪算法的精度.当前已有的算法<sup>[5-7]</sup>大多是靠试探的方式来决定目标尺寸,但是这种在参数空间里取若干离散值的方式并不能很准确地描述目标的大小. Li<sup>[8]</sup>提出使用一个共变椭圆来描述目标,并且采用二元正态分布在椭圆内的部分作为特征的加权函数.使用连续的变量来描述目标大小,但所使用的特征是预先定义的颜色分布和边界密度信息,这两个固定的特征不一定是最好的.针对这一问题,本文将信任域算法<sup>[9]</sup>和尺度空间理论<sup>[10]</sup>结合,通过检测概率分布图尺度空间中多尺度 Laplacian滤波器输出的极值,来自适应地决定目标尺度,从而可在连续的尺度空间里精确刻画目标大小的连续变化.

本文算法有如下优点:1)它将费时的特征分类器训练步骤分解为一系列简单易于计算的学习过程,可以在线进行;2)它自动调整不同弱分类器的权值,不断集成新的弱分类器,因此提高了跟踪器对光照等变化的鲁棒性;3)它连续精确地描述目标的大小变化,极大地提高跟踪精度.

## 2 Adaboos分类器设计

Adaboos学习技术<sup>[11]</sup>将一组弱分类器组合为一个强分类器.每个弱分类器在标记了正负的样本上进行训练.我们将来自于目标的样本标记为正,将来自于背景的样本标记为负.同时,每个样本都被赋予一个权值,反映了该样本在当前弱分类器中被用来训练的概率.如果某个样本已被正确分类,它在下一个弱分类器的训练中就被赋予小一些的权值.相反的,那些被错分的样本将有大一些的权值.这样,下一个弱分类器将关注于相对较难划分的样本.组合得到的强分类器被用来划分下一帧中的所有样本(对应于像素),产生一幅概率分布图,分类的归属程度就被用作概率测量值.

### 2.1 弱分类器设计

Adaboos算法的前提条件是每个弱分类器至少应该比随机猜测好.实际中这个要求并不总是容易满足.事实上 Adaboos算法的核心问题就是弱分类器的设计.在此,本文提出通过构造一个特征集合,基于此特征集合构造一组备选弱分类器来解决这一问题.因为只需要将目标和紧邻的背景分清楚就可以了,所以应选择和相邻背景有明显区分的特征.本文选择目标窗内各像素的 RGB值组合的直方图作为目标的特征.之所以选择这种直方图作为特征,是因为它对遮挡、非刚体运动和视角变化相对不太敏感.基于文献[12],备选特征由像素值的 RGB分量组合而成:

$$F = \omega_1 R + \omega_2 G + \omega_3 B \quad \omega_i \in \{\pm 2, \pm 1, 0\},$$

其中,  $F$ 是 RGB分量的整数系数线性组合,但排除  $(\omega_1, \omega_2, \omega_3) = (0, 0, 0)$ 这种无意义的情况,以及排除了其他诸如  $(2, 2, 2)$ ,  $(1, 1, 1)$ 的重复情况,最后剩余 49个备选特征.许多常用的特征都包含在这个空间中,比如原始的三分量 RGB亮度  $R+G+B$ 近似色度特征  $R-B$ 以及过彩色特征  $2G-R-B$ 等.所有备选特征都被归一到  $[0, 255]$ 范围内,并离散化成长度为 255的直方图.类似于文献[12],以目标为中心,覆盖目标的矩形窗内的像素被用来生成目标的备选特征.环绕目标矩形窗背景的像素被用来生成背景的备选特征.具体地说,如果目标矩形框为  $h \times w$ ,那么围绕该框、宽度为  $\max(h, w)$ 的边界内的背景像素被用来生成背景的备选特征.

我们使用目标像素的加权特征值来得到目标的直方图  $Hist_{tgt}$ .使用背景像素的加权特征值来得到背景的直方图  $Hist_{bg}$ .因为背景面积一般比目标面积大,所以为了防止分类器偏向于背景,我们将目标和背景的样本加权值的和都规整到 0.5.从  $Hist_{tgt}$ 得到

目标离散概率分布  $P(i), i=1, 2, \dots, 255$  从  $Hist_{bkg}$  得到背景离散概率分布  $Q(i)$ . 通过计算  $P(i)$  和  $Q(i)$  的对数比可得到一个直方图  $L$ :

$$L(i) = \max(-1, \min(1, \log \frac{\max(P(i), \delta)}{\max(Q(i), \delta)})),$$

其中  $\delta=0.001$ . 这样, 具有特征值  $x=\omega_1 r+\omega_2 g+\omega_3 b$  的像素, 就可按照  $L(x)$  的符号进行分类. 也就是说, 给定一个  $(\omega_1, \omega_2, \omega_3)$ , 我们构造如下一个弱分类器  $h(x), x=(r, g, b), x$  是像素值,  $r, g, b$  按  $\omega_1, \omega_2, \omega_3$  权值的加权, 即  $x=\omega_1 r+\omega_2 g+\omega_3 b$ . 如果  $L(x)>0$  则该像素值属于目标; 如果  $L(x)\leq 0$  则该像素值属于背景.

2.2 弱特征分类器选择

我们将从 49 个备选的弱分类器中选出能最好区分目标和背景的  $T$  个, 来生成 AdaBoost 强分类器. 选择的标准为  $L$  对目标和背景分类所得到的两类方差比.  $L(i)$  相对于  $P(i)$  的方差可计算如下:

$$\begin{aligned} \text{var}(L, P) &= E[L(i)^2] - (E[L(i)])^2 \\ &= \sum_i P(i) L(i)^2 - [\sum_i P(i) L(i)]^2. \end{aligned}$$

相对于  $Q(i)$  的方差用类似的公式计算, 两类方差比为

$$VR(L, P, Q) \equiv \frac{\text{var}(L, \frac{P+Q}{2})}{\text{var}(L, P) + \text{var}(L, Q)}$$

比值越高, 目标和背景间的区分度越高. 因此, 将 49 个备选的弱分类器按  $VR(L, P, Q)$  从大到小排序, 使用前  $T$  个用来生成 AdaBoost 强分类器.

2.3 特征分类器更新

算法在随后每一帧中, 除了跟踪, 还要进行分类器更新. 从已有的  $T$  个弱分类器中选出最好的  $K$  个, 并从备选弱分类器集合中选出  $T-K$  个新的加入到强分类器中. 具体特征分类器算法如下.

输入  $M$  个图像帧  $I_1, \dots, I_M$  目标在第一帧中的区域  $r$   
输出 目标在各帧中的区域  $r_1, \dots, r_M$

step1 初始化 (在第一帧  $I_1$  中进行).

step1.1 训练弱分类器并按  $VR(L, P, Q)$  从大到小排序.

step1.2 选出最好的前  $T$  个加入到强分类器中.

step2 对随后每一帧, 执行下面的更新操作.

step2.1 使用当前的强分类器对  $I_t$  中的各像素进行分类, 生成目标概率分布图  $L_t$ .

step2.2 使用灰度块定位算法 (见第 3 节) 定位  $L_t$  中目标的新区域  $r_t$ .

step2.3 标记  $r_t$  内所有像素为目标像素, 外像素为背景像素.

step2.4 保留强分类器中最好的  $K$  个弱分类器.

step2.5 从备选弱分类器中选出  $VR(L, P, Q)$

较大的前  $T-K$  个在帧  $I_t$  上进行训练, 加入到强分类器中.

图 1 例示了本文提出的特征分类器所产生的目标概率分布图. 图 1(a) 中的小方框代表目标, 大方框和小方框之间为背景. 注意, 这里  $T=10$ .

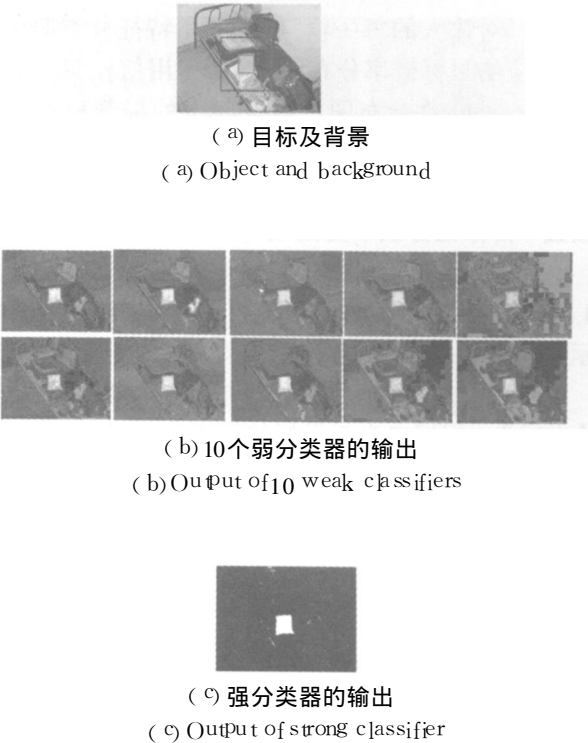


图 1 本文算法产生的目标概率分布图  
Fig 1 Object likelihood image generated by proposed algorithm

3 灰度块定位算法

利用第 2 节设计得到的强分类器对当前帧像素进行分类, 可得到当前帧的目标概率分布图, 目标区域在此图中取得较大灰度值. 我们基于尺度空间理论, 使用信任域方法来定位这些灰度块, 实现目标跟踪.

灰度图像  $I(x, y)$  的多尺度规范化 Laplacian 滤波函数定义为

$$\varphi(x, y, t) = (I_{xx}(x, y, t) + I_{yy}(x, y, t))^2,$$

其中,  $L(x, y, t)$  为图像的线性尺度空间表示,

$$\begin{aligned} L(x, y, t) &= g(x, y, t) * I(x, y) \\ g(x, y, t) &= \frac{\exp(-\frac{(x^2 + y^2)}{2t})}{2\pi t} \end{aligned}$$

为二维高斯核,  $t$  为尺度参数. 灰度块定位转变为优化问题.

$$\arg \max_{x, y, t} (\varphi(x, y, t))$$

即以  $\varphi(x, y, t)$  为目标函数, 用信任域算法检测其在灰度图像中不同位置、不同尺度下的极大值, 极大值所对应的  $(x, y)$  即为灰度块的中心位置,  $t$  为相应尺度。

## 4 Adaboost目标跟踪算法

对输入的每一帧, 首先利用特征分类器生成其对应的目标概率分布图, 然后运用信任域算法搜索  $\varphi(x, y, t)$  在分布图尺度空间中的局部极大值来完成跟踪。具体算法描述如下。

输入  $n$  个图像帧  $I_1, \dots, I_n$  目标在第一帧中的区域  $r$   
输出 目标在各帧中的区域  $r_1, \dots, r_n$

step 1 初始化 (在第一帧  $I_1$  中进行)。以  $w_1$ 、 $h_1$  分别表示目标区域  $r$  的宽和高, 以  $w$  和  $l$  分别表示图像的宽和高。按如下各步骤初始化特征分类器。

step 1.1 提取目标和背景的像素构成像素集合  $\{x_i\}_{i=1}^N$ , 标记来自于目标的像素为 1, 来自于背景的像素为 -1, 这样得到标记集合  $\{y_i\}_{i=1}^N$ 。

step 1.2 初始化权值集合  $\{\omega_i\}_{i=1}^N$ , 每个权值均为  $1/N$

step 1.3 训练弱分类器并按  $VR(L, P, Q)$  从大到小排序。

step 1.4 对于  $t = 1, 2, \dots, T$

step 1.4.1 将  $\{\omega_i\}_{i=1}^N$  规整为一个概率分布。

step 1.4.2 从排序的弱分类器集合中, 选取弱分类器  $h_t$  使其最小化

$$\text{err} = \sum_{i=1}^N \omega_i |h_t(x_i) - y_i|.$$

step 1.4.3 设置当前弱分类器权值

$$\alpha_t = \frac{1}{2} \log \frac{1 - \text{err}}{\text{err}}.$$

step 1.4.4 更新样本权值

$$\omega_i = \omega_i \exp(\alpha_t |h_t(x_i) - y_i|).$$

step 1.5 强分类器由  $\text{sign}(H(x))$  给出, 其中

$$H(x) = \sum_{t=1}^T \alpha_t h_t(x).$$

step 2 对每一帧图像  $I_t$

step 2.1 提取像素集合  $\{x_i\}_{i=1}^N$ 。

step 2.2 使用当前强分类器  $H(x)$  对  $I_t$  中所有像素进行分类, 产生概率分布图  $L_t$

step 2.3 以  $r_1$  为初始区域, 使用灰度块定位算法定位  $L_t$  中目标所在的新区域  $r_t$

step 2.4 基于  $r_t$  定义新的标记集合  $\{y_i\}_{i=1}^N$ 。

step 2.5 将  $\{\omega_i\}_{i=1}^N$  初始化为  $1/N$

step 2.6 对于  $t = 1, 2, \dots, K$  (选取最好的  $K$  个

弱分类器)。

step 2.6.1 将  $\{\omega_i\}_{i=1}^N$  规整为一个概率分布。

step 2.6.2 从强分类器现在所用的弱分类器  $\{h_1(x), \dots, h_K(x)\}$  中选取具有最小  $\text{err}$  的  $h_t$

step 2.6.3 更新  $\alpha_t$  和  $\{\omega_i\}_{i=1}^N$ 。

step 2.7 对于  $t = K+1, \dots, T$  (新增弱分类器)。

step 2.7.1 训练弱分类器并按  $VR(L, P, Q)$  从大到小排序。

step 2.7.2 将  $\{\omega_i\}_{i=1}^N$  规整为一个概率分布。

step 2.7.3 从排序的弱分类器中, 选择具有最大  $VR(L, P, Q)$  的  $h_t$  进行训练加入强分类器。

step 2.7.4 更新  $\alpha_t$  和  $\{\omega_i\}_{i=1}^N$ 。

step 2.8 更新后的强分类器由  $\text{sign}(H(x))$  给出, 其中

$$H(x) = \sum_{t=1}^T \alpha_t h_t(x).$$

其中, step 2.3 中的灰度块定位算法, 可由如下具体步骤给出。

step 1 将  $L_t$  进行缩放, 使得其宽高比为  $\rho = w_{h_1}/h/w_1$ , 如果保持宽度  $w$  不变,  $L_t$  的高度将变为  $w/\rho$ 。

step 2 初始化向量  $x_0 = (x_{\text{pre}}, y_{\text{pre}}, t_{\text{pre}})^T$ , 其中  $(x_{\text{pre}}, y_{\text{pre}})$  为  $r_{t-1}$  的中心,  $t_{\text{pre}}$  为上一帧中的尺度参数。设置信任域算法初始半径  $\Delta_0 = 9$  最小半径  $\Delta_{\text{end}} = 0.1$  最大循环次数  $\text{MAX}_{\text{iter}} = 20$

step 3 以  $f(x) = \varphi(x, y, t)$  为目标函数, 使用信任域算法进行优化, 得到最大化  $f(x)$   $x_{\text{opt}} = (x_{\text{opt}}, y_{\text{opt}}, t_{\text{opt}})^T$ , 其中,  $(x_{\text{opt}}, y_{\text{opt}})$  即为  $r_t$  的中心位置,  $t_{\text{opt}}$  为其尺度参数。

## 5 实验

为了验证本文算法的性能, 在真实序列上进行大量实验, 并将本文算法和 Mean Shift 以及三步搜索算法进行比较。首先我们验证本文算法对光照剧烈变化的适应性。图 2 显示了对一个玻璃瓶跟踪的结果。该玻璃瓶先在光照较亮处运动, 随后从第 50 帧开始, 玻璃瓶进入阴影, 表面迅速变暗, 颜色特征发生剧烈变化。由于本文算法不断更新弱分类器, 适应了这一变化, 可以看出, 目标方框始终套牢目标, 取得较高的跟踪精度。在跟踪过程中, 在初始帧中选定目标后, 就无需再输入任何参数。图像大小为  $320 \times 240$  使用  $T=10$  个弱分类器。

图 3 图 4 是 Mean Shift 算法和本文算法对“行人”序列的跟踪结果。实验中, 行人从明亮处进入暗

处,做快速运动之后,再次回到明亮处的跟踪结果.由于光照改变使得目标特征发生明显变化,使得 Mean Shift算法在 180 帧时 (图 3 (b)) 中心位置精度明显下降.背景中窗帘的黑色部分对目标构成很大干扰,使得 Mean Shift算法在第 219 帧 (图 3 (c)) 完全丢失目标,不得不在第 220 帧重新初始化目标.目标在跟踪过程中不仅存在简单的平移,而且进行了剧烈的左右摇摆和侧转.从 330 帧 ~ 355 帧还有一次突然的下蹲动作,导致 Mean Shift算法跟踪精度下降 (图 3 (d)、(e)).由于第 220 帧重新初始化目标时, Mean Shift算法使用的特征是光线较弱时的特征,而自第 400 帧起,目标重新进入光照强的区域,结果跟踪精度不断下降 (图 3 (f)、(g)),直至丢失目标偏离至窗帘暗红色区域.

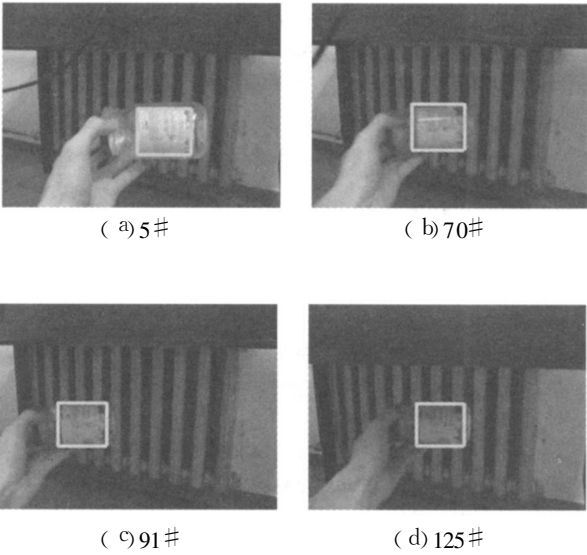


图 2 本文算法对玻璃瓶的跟踪结果

Fig 2 Tracking results of sequence glass by proposed algorithm

由于在每次成功定位目标后,本文算法都集成新的弱分类器并更新已有分类器的权值,提高了强分类器对目标特征变化的适应能力,所以当人从明亮处进入暗处 (图 4 (a)、(b)、(c)),剧烈机动 (图 4 (d)、(e)),以及最后回到光照强的区域时 (图 4 (f)、(g)、(h)),算法都保持了较高的跟踪精度.这个实验验证了本文算法对存在干扰的背景中,进行强机动快变化目标的跟踪能力.

图 5 显示在“行人”序列跟踪过程中,本文算法所使用的各弱分类器权值随时间的变化情况.我们将 49 个弱分类器依次编号 (图中 Weak classifier number 轴),将它们在各帧 (沿 Frame 轴) 中的权值沿着 weight 轴绘出.由图可见,本文算法为适应目标特征变化,对分类器更新的情况.

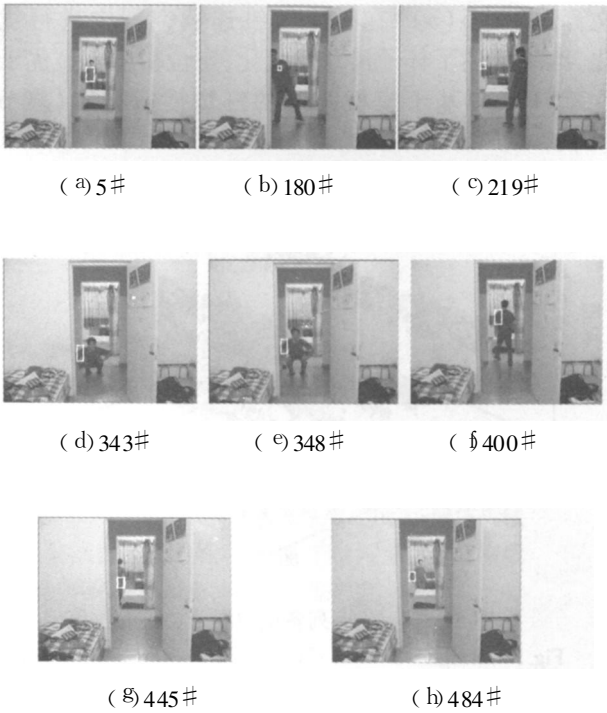


图 3 Mean Shift算法对“行人”序列的跟踪结果  
Fig 3 Tracking results of sequence pedestrian by Mean Shift method

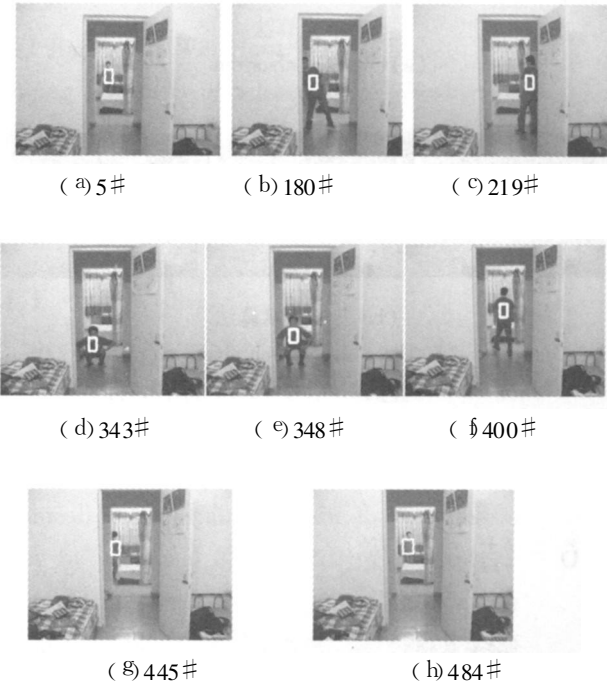


图 4 本文算法对“行人”序列的跟踪结果  
Fig 4 Tracking results of sequence pedestrian by proposed algorithm

在图 6 所示的实验中,我们验证本文算法处理缩放目标的能力.在这个序列中,杯子被逐渐放大.对于基于模板匹配的跟踪算法而言,由于目标的尺

寸不断变化,难以准确定位杯子.而由图6可见,本文算法(图6(a))在一个三维连续空间中进行优化,准确描述了杯子尺寸的变化,定位精度明显优于文献[13]中基于模板匹配的三步搜索算法(图6(b)).

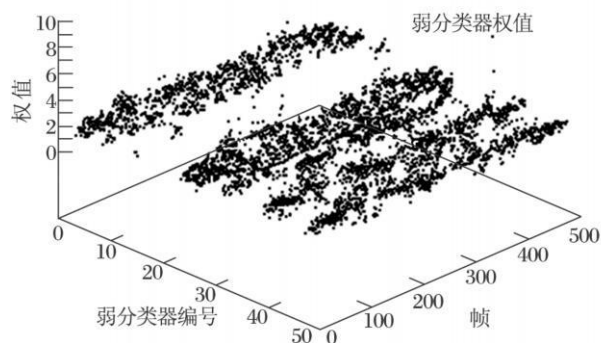


图5 “行人”序列各帧中弱分类器的权值

Fig 5 Weights of weak classifiers for sequence pedestrian



(a) 本文算法

(a) Proposed algorithm



(b) 三步搜索算法

(b) Three-step search algorithm

图6 2种算法对“杯子”的跟踪结果对比

Fig 6 Tracking result comparison of sequence cup between 2 algorithms

## 6 结束语

在线更新特征分类器,不断集成新的弱分类器,调整不同弱分类器的权值,使得本文算法得以适应目标特征的连续变化,提高了跟踪的鲁棒性和精度.结合信任域方法和尺度空间理论,进一步提高算法对目标大小变化的描述精度.真实序列上的大量实验也表明了本文算法的有效性.由于弱分类器设计

为使用计算量很小的直方图,所以用C程序实现的本文算法,在Mobile Sempron 1.79GHz Windows平台上运行时,速度达到了10 FPS.

## 参考文献

- [1] Shi Jianbo, Tomasi C. Good Features to Track // Proc of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, USA, 1994: 593-600.
- [2] Han Bo-hyung, Davis L. Object Tracking by Adaptive Feature Extraction // Proc of the International Conference on Image Processing, Singapore, Singapore, 2004, III: 1501-1504.
- [3] Stern H, Efros B. Adaptive Color Space Switching for Face Tracking in Multi-Colored Lighting Environments // Proc of the 5th IEEE International Conference on Automatic Face and Gesture Recognition, Washington, USA, 2002: 249-254.
- [4] Aviaian S. Subset Selection for Efficient SVM Tracking // Proc of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Madison, USA, 2003, I: 85-92.
- [5] Comaniciu D, Ramesh V, Meer P. Real-Time Tracking of Non-Rigid Objects Using Mean Shift // Proc of the IEEE Conference on Computer Vision and Pattern Recognition, Hilton Head Island, USA, 2000, II: 142-149.
- [6] Comaniciu D, Ramesh V, Meer P. Kernel-Based Object Tracking // IEEE Trans on Pattern Analysis and Machine Intelligence, 2003, 25(5): 564-577.
- [7] Jia Jingping, Zhao Rongchun. Tracking of Objects in Image Sequences Using Bandwidth Matrix Mean Shift Algorithm // Proc of the 7th International Conference on Signal Processing, Beijing, China, 2004, III: 918-921.
- [8] Liu T L, Chen H T. Real-Time Tracking Using Trust Region Methods // IEEE Trans on Pattern Analysis and Machine Intelligence, 2004, 26(3): 397-402.
- [9] Bealeen F V. Intermediate Report on the Development of an Optimization Code for Smooth Continuous Objective Functions When Derivatives Are Not Available [EB/OL]. [2003-08-09]. [http://www.optimization-online.org/DB\\_HTML/2003/08/704.html](http://www.optimization-online.org/DB_HTML/2003/08/704.html)
- [10] Lindeberg T. Feature Detection with Automatic Scale Selection // International Journal of Computer Vision, 1998, 30(2): 79-116.
- [11] Duda R Q, Hart P E, Stork D G. Pattern Classification, 2nd Edition. New York, USA: John Wiley & Sons, 2001: 385-386.
- [12] Leordeanu M, Collins R T, Liu Yanxi. Online Selection of Discriminative Tracking Features // IEEE Trans on Pattern Analysis and Machine Intelligence, 2005, 27(10): 1631-1643.
- [13] Ren Jinchang, Zhang Wenzhe, Zhao Rongchun, et al. An Automatic Target Tracking Method Based on Self-Adaptive Threshold under Complex Background. Application Research of Computers, 2003, 20(4): 55-57 (in Chinese).

(任金昌, 张文哲, 赵荣椿, 等. 一种基于自适应阈值的复杂背景下自动目标跟踪方法. 计算机应用研究, 2003, 20(4): 55-57)