

文章编号:1671-6833(2016)01-0001-05

# 基于 ELM 与 Mean Shift 的抗遮挡目标跟踪算法

毛晓波, 郝向东, 梁 静

(郑州大学 电气工程学院, 河南 郑州 450001)

**摘 要:** 针对目标跟踪过程中发生遮挡时跟踪出现偏差的情况, 在传统 Mean Shift 跟踪算法的基础上引入极限学习机(extreme learning machine, ELM), 提出了基于 ELM 与 Mean Shift 的目标跟踪算法。该算法根据过去 3 个时刻的目标位置信息, 利用 ELM 预测出目标当前可能位置, 代替目标前一时刻位置作为 Mean Shift 迭代起始点, 并在邻域范围内进行迭代, 得到目标的真实位置。实验结果表明, 与现有的改进算法相比, 新算法减少了迭代次数和运算时间, 同时在遮挡情况下能够准确定位目标位置并进行跟踪, 提高了跟踪系统的实时性和鲁棒性。

**关键词:** 目标跟踪; 遮挡; Mean Shift; ELM

**中图分类号:** U461.6 **文献标志码:** A **doi:**10.3969/j.issn.1671-6833.201503043

## 0 引言

目标跟踪技术是机器视觉的热门课题, 广泛应用于工业生产、国防军事和生活安防等众多领域。遮挡问题是目前目标跟踪的难题之一, 遮挡尤其是严重遮挡会直接影响跟踪的准确性和持续性, 可能导致目标跟踪发生偏差甚至丢失目标。因此, 研究抗遮挡的目标跟踪算法具有重要的理论和实际意义。

Mean Shift 算法<sup>[1]</sup>、光流算法<sup>[2]</sup>、卡尔曼滤波算法<sup>[3]</sup>、粒子滤波算法<sup>[4]</sup>等都能在一定条件下有效地跟踪目标。其中, Mean Shift 算法以其无需参数、快速模式匹配的特性被广泛地应用<sup>[5]</sup>。但是, 当目标运动速度过快或遮挡较为严重时, 传统 Mean Shift 算法往往会发生目标的定位偏差甚至跟丢。为解决上述问题, 研究人员提出了一些有效方法, 如在算法中引入模板分块匹配<sup>[6]</sup>或将 Mean Shift 算法与滤波算法相结合<sup>[7-10]</sup>等。这些方法能够提高跟踪算法的抗遮挡能力, 但由于需要进行分块跟踪或进行复杂的滤波运算, 增加了算法的复杂度, 跟踪实时性受到影响。针对以上问题, 笔者提出一种基于 ELM 与 Mean Shift 的抗遮挡目标跟踪算法, 利用 ELM 的快速学习和预测能力预

测目标的运动趋势, 以预测出的目标中心作为 Mean Shift 迭代起始点, 通过迭代得到目标当前真实位置。笔者将新算法与基于模板分块匹配的 Mean Shift 算法和基于卡尔曼滤波的 Mean Shift 算法作了比较。实验证明, 新算法能够在目标高速运动或遮挡情况下快速精确稳定地跟踪目标, 具有很好的实时性和鲁棒性。

## 1 Mean Shift 算法

Mean Shift 算法是一种基于核密度估计的非参数模式匹配算法, 通过建立加权直方图来描述目标的颜色分布<sup>[11]</sup>。该方法利用像素概率密度函数梯度最优化方法, 首先选定跟踪目标的区域, 计算核函数加权下的目标区域的直方图分布, 然后用同样的方法对当前帧的数据密度直方图进行漂移, 且通过 Mean Shift 迭代寻找与目标区域相似性最大的区域, 最终实现快速目标定位。

### 1.1 目标模型

Mean Shift 跟踪算法利用目标区域的直方图分布来描述目标, 根据目标模型和候选模型的相似度进行中心的均值漂移。在第一帧中手动确定包含跟踪目标的区域, 设中心坐标为  $\mathbf{d}_0$ ; 区域中包含  $n$  个像素, 位置记为  $\{\mathbf{d}_i\}_{i=1,2,\dots,n}$ ; 特征值  $u =$

收稿日期:2015-03-10; 修订日期:2015-05-20

基金项目:国家自然科学基金资助项目(61473266); 河南省教育厅科学技术研究重点项目资助计划(14A410001)

作者简介:毛晓波(1965—), 男, 河南开封人, 郑州大学教授, 博士, 主要从事仿生机器视觉研究, E-mail: mail-mxb@zzu.edu.cn.

引用本文:毛晓波, 郝向东, 梁静. 基于 ELM 与 Mean Shift 的抗遮挡目标跟踪算法[J]. 郑州大学学报(工学版), 2016, 37(1): 1-5.

1, 2, \dots, m 是直方图的颜色索引. 则目标模型的特征估计概率密度可表示为

$$q_u = C_h \sum_{i=1}^n s\left(\left\|\frac{\mathbf{d}_i - \mathbf{d}_0}{h}\right\|^2\right) \delta[b(\mathbf{d}_i) - u]. \quad (1)$$

式中:  $h$  为核函数的窗宽, 即目标区域的大小;  $s(\cdot)$  为核函数的轮廓函数, 对靠近中心的区域像素赋予一个大的权值, 对远离中心的区域像素赋予越来越小的权值;  $\delta[\cdot]$  是克罗内科函数,  $\delta[\cdot]$  用于判断目标区域中像素  $\mathbf{d}_i$  的颜色值是否属于第  $u$  个单元的颜色索引值, 若为 1, 则是, 为 0, 则不是;  $C_h$  为归一化常数, 其取值满足

$$C_h = 1 / \sum_{i=1}^m q_u = 1. \quad (2)$$

## 1.2 候选模型

目标运动过程中, 每一帧图像中目标可能存在的区域称为候选区域, 其核函数窗宽中的特征模型称为目标候选模型. 假设第  $k$  帧时得到的目标中心为  $\mathbf{y}_k$ , 则在第  $k+1$  帧中候选模型的特征值为:

$$p_u(\mathbf{y}) = C_h \sum_{i=1}^n s\left(\left\|\frac{\mathbf{d}_i - \mathbf{y}_k}{h}\right\|^2\right) \delta[b(\mathbf{d}_i) - u]. \quad (3)$$

## 1.3 相似度判定函数

在跟踪过程中, Mean Shift 算法将候选模型与目标模型进行相似性判别, 寻找式(3)中最优的  $\mathbf{y}_k$ , 使得  $p_u$  和  $q_u$  相似度达到最大.  $q_u$  和  $p_u$  的相似度可由如下函数判定:

$$\rho(\mathbf{y}) \equiv \rho(p(\mathbf{y}), q) = \sum_{u=1}^m \sqrt{p_u(\mathbf{y}) q_u}. \quad (4)$$

式中:  $\rho(\mathbf{y})$  是 Bhattacharyya 系数, 其值在 0 ~ 1 之间,  $\rho(\mathbf{y})$  值越大, 表示两个模型相似度越高. 目标发生遮挡时,  $\rho(\mathbf{y})$  变小, 给定遮挡阈值  $T$ , 当  $\rho(\mathbf{y}) < T$  时, 认为目标发生遮挡.

## 1.4 Mean Shift 算法

将上述式(4)在模型初始位置  $\mathbf{y}_k$  处泰勒展开得:

$$\rho[p(\mathbf{y}), q] = \frac{1}{2} \sum_{u=1}^m \sqrt{p(\mathbf{y}_k) q_u} + \frac{1}{2} \sum_{u=1}^m p_u(\mathbf{y}) \sqrt{\frac{q_u}{p_u(\mathbf{y}_k)}}. \quad (5)$$

将式(4)代入式(5)整理得:

$$\rho[p(\mathbf{y}), q] \approx \frac{1}{2} \sum_{u=1}^m \sqrt{p(\mathbf{y}_k) q_u} + \frac{C_h}{2} \sum_{i=1}^n \mathbf{w}_i s\left(\left\|\frac{\mathbf{y} - \mathbf{d}_i}{h}\right\|^2\right).$$

$$\text{式中: } \mathbf{w}_i = \sum_{u=1}^m \delta[b(\mathbf{d}_i) - u] \sqrt{\frac{q_u}{p_u(\mathbf{y}_k)}}. \quad (6)$$

由式(6)可知, 相似度函数只随等式右侧第二项中  $\mathbf{y}$  的变化而变化, 与第一项无关. 通过 Mean Shift 算法在邻域内不断迭代, 对相似度函数求最大值, 即可得到跟踪目标新的位置:

$$\mathbf{y}_{k+1} = \frac{\sum_{i=1}^n \mathbf{d}_i \mathbf{w}_i g\left(\left\|\frac{\mathbf{y}_k - \mathbf{d}_i}{h}\right\|^2\right)}{\sum_{i=1}^n \mathbf{w}_i g\left(\left\|\frac{\mathbf{y}_k - \mathbf{d}_i}{h}\right\|^2\right)}. \quad (7)$$

迭代结束后, 得到的  $\mathbf{y}_{k+1}$ , 即为第  $k+1$  帧的目标中心.

## 1.5 Mean Shift 算法的缺点

在实际应用中, 由于 Mean Shift 算法没有专门有效的抗遮挡机制, 当跟踪目标经过遮挡物时, 往往会发生目标的跟偏甚至跟丢. 另外, 当目标高速运动时, Mean Shift 算法跟踪效果变差. 为此, 笔者在 Mean Shift 算法的基础上引入了 ELM 算法<sup>[12-13]</sup>, 利用 ELM 的预测能力较好地解决了上述问题.

## 2 ELM 与 Mean Shift 相结合的算法

### 2.1 ELM 的基本原理

若一个标准的单隐层前馈神经网络含有  $N$  个训练样本, 记为  $\{\mathbf{x}_i, \mathbf{o}_i\}_{i=1,2,\dots,N}$ , 其中  $\mathbf{x}_i = (x_{i1}, x_{i2}, \dots, x_{in})^T \in \mathbf{R}^n$ ,  $\mathbf{o}_i = (o_{i1}, o_{i2}, \dots, o_{im})^T \in \mathbf{R}^m$ , 含有  $N'$  个隐层节点, 设其激励函数为  $g(x)$ , 则该神经网络的输出可以表示为

$$f(\mathbf{x}_j) = \sum_{i=1}^{N'} \beta_i g(\mathbf{w}_i \cdot \mathbf{x}_j + b_i), j = 1, 2, \dots, N. \quad (8)$$

式中:  $\mathbf{w}_i = (w_{i1}, w_{i2}, \dots, w_{in})^T$  为输入神经元与第  $i$  个隐层节点的输入权值;  $\beta_i = (\beta_{i1}, \beta_{i2}, \dots, \beta_{im})^T$  为第  $i$  个隐层节点与输出神经元的输出权值;  $b_i$  为第  $i$  个隐层节点的偏置.

神经网络的理想目标是实际输出与给定样本相等, 因此存在  $\beta_i$ 、 $\mathbf{w}_i$  和  $b_i$  使得

$$\sum_{i=1}^{N'} \beta_i g(\mathbf{w}_i \cdot \mathbf{x}_j + b_i) = \mathbf{o}_j, j = 1, 2, \dots, N. \quad (9)$$

式(9)即为 ELM 的模型, 其中,  $\mathbf{o}_j$  为第  $j$  个输入样本的对应的输出样本, 该模型可用矩阵表示为

$$\mathbf{H}\beta = \mathbf{O}. \quad (10)$$

式中:

$$\mathbf{H} = \begin{bmatrix} g(\mathbf{w}_1 \cdot \mathbf{x}_1 + b_1) & \cdots & g(\mathbf{w}_{N'} \cdot \mathbf{x}_1 + b_{N'}) \\ \vdots & \cdots & \vdots \\ g(\mathbf{w}_1 \cdot \mathbf{x}_N + b_1) & \cdots & g(\mathbf{w}_{N'} \cdot \mathbf{x}_N + b_{N'}) \end{bmatrix}_{N \times N'};$$

$$\boldsymbol{\beta} = \begin{bmatrix} \boldsymbol{\beta}_1^T \\ \vdots \\ \boldsymbol{\beta}_{N'}^T \end{bmatrix}_{N' \times m}; \quad \boldsymbol{O} = \begin{bmatrix} \boldsymbol{o}_1^T \\ \vdots \\ \boldsymbol{o}_N^T \end{bmatrix}_{N \times m}.$$

$\boldsymbol{H}$  为神经网络的隐层输出矩阵. 在 ELM 中, 输入权重  $\boldsymbol{w}_i$  和隐层偏置  $\boldsymbol{b}_i$  由系统随机确定, 则  $\boldsymbol{H}$  被唯一确定. 由 Moore-Penrose 广义逆定理可求出  $\boldsymbol{H}^+$ , 则

$$\boldsymbol{\beta} = \boldsymbol{H}^+ \boldsymbol{O}.$$

(11)

2.2 基于 ELM 与 Mean Shift 的抗遮挡目标跟踪算法

在目标运动过程中, 第  $t$  时刻目标的位置与其上一时刻的位置、速度和加速度有关, 即可以通过这 3 个参数确定目标当前时刻的位置. 而这 3 个参数包含在第  $t-1$ 、 $t-2$ 、 $t-3$  时刻的目标位置信息中, 因此, 由前 3 时刻目标的位置可以预测出第 4 时刻目标的位置.

首先, 通过对无遮挡目标的精确跟踪得到一组目标运动轨迹, 为了提高样本精确度, 采用手工逐帧确定. 取连续 200 组中的前 160 组位置坐标作为 ELM 的训练样本, 将连续 3 组坐标作为输入, 第 4 组作为输出训练 ELM 神经网络. 利用后 40 组作为测试样本, 通过不断调节隐层节点数使预测达到最优. 同理, 对遮挡目标进行训练, 训练样本和测试样本依然分别取 160 组和 40 组. 调节隐层节点数, 使两个训练集合的预测结果最佳. 通过试验, 当隐层节点设为 30 时, 预测值达到最优.

用 ELM 进行目标位置预测时, 首先假设初始时刻目标的速度和加速度均为零, 同时假设前 3 时刻目标位置均为初始时刻选定的目标位置, 即 ELM 的 3 组输入均为初始时刻目标位置. 目标跟踪过程中, 由前 3 时刻的目标位置  $\boldsymbol{d}_1$ 、 $\boldsymbol{d}_2$ 、 $\boldsymbol{d}_3$  通过训练好的 ELM 网络模型预测出当前时刻的目标位置, 记为  $\boldsymbol{y}'$ , 用  $\boldsymbol{y}'$  代替式 (3) 中的  $\boldsymbol{y}_k$  作为 Mean Shift 迭代的起始点, 在邻域范围内进行迭代, 从而得到目标当前的真实位置  $\boldsymbol{y}_{k+1}$ . 式 (3) 变为

$$p_u(\boldsymbol{y}) = C_h \sum_{i=1}^n s\left(\left\|\frac{\boldsymbol{d}_i - \boldsymbol{y}'}{h}\right\|^2\right) \delta[b(\boldsymbol{d}_i) - u]. \quad (12)$$

于是, 式 (7) 变为

$$\boldsymbol{y}_{k+1} = \frac{\sum_{i=1}^n \boldsymbol{d}_i w_i g\left(\left\|\frac{\boldsymbol{y}' - \boldsymbol{d}_i}{h}\right\|^2\right)}{\sum_{i=1}^n w_i g\left(\left\|\frac{\boldsymbol{y}' - \boldsymbol{d}_i}{h}\right\|^2\right)}.$$

(13)

同时计算  $\rho(\boldsymbol{y})$  并与遮挡阈值  $T$  进行比较, 若

$\rho(\boldsymbol{y}) \geq T$ , 则继续执行; 若  $\rho(\boldsymbol{y}) < T$ , 说明发生遮挡, 则以  $\boldsymbol{y}'$  代替  $\boldsymbol{y}_{k+1}$  作为当前帧目标的实际位置, 然后进行下一帧的预测, 直到再次  $\rho(\boldsymbol{y}) \geq T$ , 说明遮挡结束, 跟踪正常, 如此就完成了对目标的跟踪.

3 仿真结果分析

在 2.10 GHz 主频 4 GB 内存的 Windows8 操作系统电脑上使用 Matlab2010, 对笔者提出的算法进行了测试实验, 并在相同条件下与基于模板分块匹配的 Mean Shift 算法和基于卡尔曼滤波的 Mean Shift 算法跟踪效果进行了对比.

为验证跟踪结果的实时性, 我们对公路上高速行驶的汽车进行了 20 次跟踪, 统计 3 种算法对第 11 帧到 110 帧共计 100 帧目标进行 20 次跟踪的平均迭代次数, 并对跟踪效果进行了对比. 表 1 显示了 3 种不同算法的平均迭代次数对比结果. 通过对比可以发现, 由于新算法采用了位置预测, 迭代起始位置与目标实际位置的距离大大减少, 因此比基于模板分块匹配的 Mean Shift 算法迭代次数平均减少 1.20 次; 同时, 新算法的位置预测方法较卡尔曼滤波更为简便, 一定程度上简化了融合算法且计算效率有所提高, 比基于卡尔曼滤波的 Mean Shift 算法平均减少 0.76 次.

表 1 3 种算法迭代次数

Tab.1 Iteration number of different algorithms

| 算法   | 迭代次数 |
|------|------|
| 分块算法 | 3.34 |
| 滤波算法 | 2.90 |
| 新算法  | 2.14 |

表 2 为 3 种算法分别对 100 帧视频进行 20 次跟踪的平均时间. 由表 2 可知, 新算法平均跟踪时间是 4 918 ms, 比两种已有改进算法分别减少 17.9% 和 7.2%.

表 2 3 种算法跟踪时间

Tab.2 Tracking time of different algorithms

| 项目   | 时间/ms |
|------|-------|
| 正常播放 | 4 203 |
| 分块算法 | 5 993 |
| 滤波算法 | 5 302 |
| 新算法  | 4 918 |

图 1 为 3 种算法跟踪效果, 分别取第 20、40、60、80 帧图片. 表 3 为 3 种算法的跟踪误差, 表示跟踪位置和实际位置中心的距离, 实际位置采用手工逐帧确定. 由表 3 可知, 新算法与基于模板分

块匹配的 Mean Shift 算法跟踪误差近似,明显小于基于卡尔曼滤波的 Mean Shift 算法.

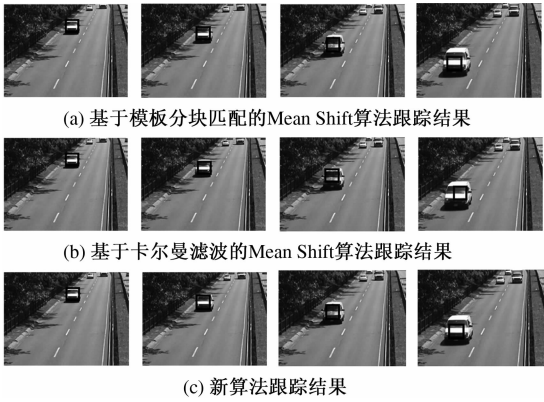


图 1 高速运动车辆跟踪结果对比

Fig.1 Tracking results comparison of high speed car

表 3 3 种算法跟踪误差

Tab.3 Tracking errors of different algorithms

| 算法   | Frame20 | Frame40 | Frame60 | Frame80 |
|------|---------|---------|---------|---------|
| 分块算法 | 5.385   | 18.601  | 40.792  | 52.631  |
| 滤波算法 | 6.708   | 24.839  | 61.221  | 107.415 |
| 新算法  | 6.000   | 17.804  | 44.689  | 54.918  |

图 2 为 3 种算法对公路上被遮挡车辆进行跟踪的结果,分别取第 5、17、29、43 帧图片.表 4 为 3 种算法的跟踪误差.由表 4 可以看出,当目标发生遮挡时,3 种算法跟踪误差变化较为剧烈,此时新算法的跟踪误差为 66.129,明显小于另外两种改进算法.遮挡结束后恢复到正常跟踪,3 种算法跟踪误差均变小.

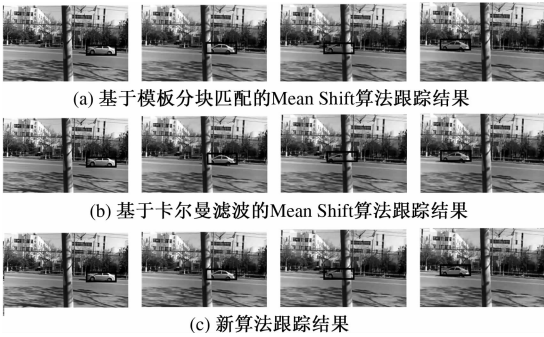


图 2 遮挡车辆跟踪结果对比

Fig.2 Tracking results comparison of occluded car

表 4 3 种算法跟踪误差

Tab.4 Tracking errors of different algorithms

| 算法   | Frame5 | Frame17 | Frame29 | Frame43 |
|------|--------|---------|---------|---------|
| 分块算法 | 3.000  | 31.321  | 81.025  | 49.980  |
| 滤波算法 | 4.123  | 37.121  | 110.023 | 77.621  |
| 新算法  | 2.828  | 32.249  | 66.129  | 41.976  |

表 5 为 3 种算法跟踪 60 帧车辆遮挡视频所

用时间,新算法跟踪时间比另外两种改进算法分别减少 18.5% 和 7.5% .

表 5 3 种算法跟踪时间

Tab.5 Tracking time of different algorithms

| 项目   | 时间/ms |
|------|-------|
| 正常播放 | 2 187 |
| 分块算法 | 4 151 |
| 滤波算法 | 3 659 |
| 新算法  | 3 384 |

图 3 为 3 种算法对被遮挡的行人进行跟踪的结果,取第 20、40、50、70 帧进行比较.由图 3 可以看出,由于跟踪过程中行人被完全遮挡,基于模板分块匹配的 Mean Shift 算法无法进行子块匹配,造成了错误的跟踪,丢失了目标;而基于卡尔曼滤波的 Mean Shift 算法和新算法在目标被遮挡时自动以预测的目标位置为起点继续跟踪,遮挡结束后恢复正常跟踪,因此能够较好地跟踪目标.

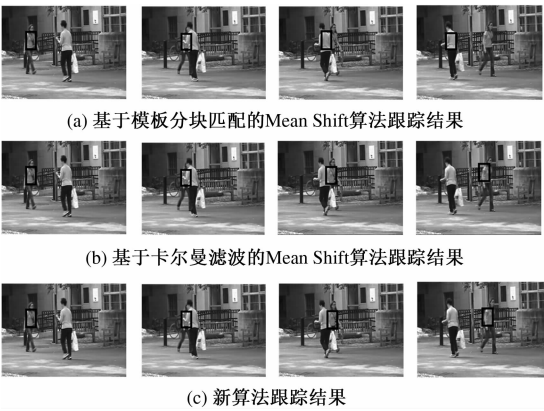


图 3 行人遮挡跟踪结果对比

Fig.3 Tracking results comparison of occluded people

以上实验表明,当目标高速运动或发生遮挡时,新算法具有更好的实时性和鲁棒性.

4 结论

笔者提出一种基于 ELM 与 Mean Shift 的抗遮挡目标跟踪算法,解决了 Mean Shift 算法在目标发生遮挡时跟踪性能变差的问题.该算法利用目标的历史位置信息,通过神经网络预测出当前位置信息,所需信息少,计算量小,有效地减少了迭代次数,在目标发生遮挡时以预测位置信息继续跟踪,能够准确跟踪目标.实验证明,新算法能够在目标发生严重遮挡时准确跟踪目标,具有很好的实时性和鲁棒性,并且与现有的改进算法相比,具有一定的优势.

神经网络具有强大的自学能力、联想能力和

计算能力,将神经网络应用到目标跟踪技术中,突破传统算法的限制,使跟踪系统更加智能化,为解决目标跟踪问题提供了新方法,对于处理目标跟踪过程中的遮挡问题,提高复杂场景下目标跟踪的稳定性具有重要意义.笔者并没有考虑多目标跟踪的问题,因此下一步将考虑对多个目标进行跟踪时如何利用神经网络实现有效跟踪.

## 参考文献:

- [1] 何志勇,蔡乐才,许继家.基于 Mean Shift 算法跟踪视频中运动目标[J].郑州大学学报(理学版),2010,42(1):38-40,48.
- [2] 李剑峰,黄增喜,刘怡光.基于光流场估计的自适应 Mean-Shift 目标跟踪算法[J].光电子·激光,2012,23(10):1996-2002.
- [3] WENG S K, KUO C M, TU S K. Video object tracking using adaptive Kalman filter[J]. Journal of visual communication and image representation, 2006, 17(6):1190-1208.
- [4] 屈鉴铭,刘志镜,贺文骅.结合有向场景运动模式的粒子滤波行人跟踪方法[J].西安交通大学学报,2014,48(12):74-79.
- [5] 袁霄,王丽萍.基于 MeanShift 算法的运动人体跟踪[J].计算机工程与科学,2008,30(4):46-49.
- [6] 颜佳,吴敏渊,陈淑珍,等.应用 Mean Shift 和分块的抗遮挡跟踪[J].光学精密工程,2010,18(6):1413-1418.
- [7] 章学静,陈禾,杨静.结合卡尔曼滤波和 Mean Shift 的抗遮挡跟踪算法[J].北京理工大学学报,2013,33(10):1056-1061.
- [8] 张伟丽,江春华,郝宗波.基于改进的均值漂移和卡尔曼滤波的目标跟踪算法[J].计算机应用与软件,2014,31(5):240-243.
- [9] HE J W, YANG Y Y. Signal Processing, communications and computing: ICSPCC2014[C] // Multi-interactive tracking method using meanshift based on kalman filter. Guilin, 2014:22-27.
- [10] 权义萍,金鑫,张蕾,等.基于 Mean-Shift 的卡尔曼粒子滤波车辆跟踪算法研究[J].计算机应用与软件,2014,31(9):236-252.
- [11] 杨杰,张翔.视频目标检测和跟踪及其应用[M].上海:上海交通大学出版社,2012.
- [12] 王杰,毕浩洋.一种基于粒子群优化的极限学习机[J].郑州大学学报(理学版),2013,45(1):100-104.
- [13] 胡义涵,张小刚,陈华,等.一种基于鲁棒估计的极限学习机方法[J].计算机应用研究,2012,29(8):2926-2930.

## Target Tracking Algorithm Based on Mean Shift with ELM

MAO Xiaobo, HAO Xiangdong, LIANG Jing

(School of Electrical Engineering, Zhengzhou University, Zhengzhou 450001, China)

**Abstract:** In view of the problem of object deviation when occlusions occur during the target tracking, a new algorithm using Mean Shift with ELM is proposed. According to the formal information of the object's location, current possible location was predicted by ELM, the iteration was started from the possible location instead of formal location, and the object's real center is calculated by mean shift algorithm. The simulation results show that proposed algorithm can track precisely target occluded, operation time and number of iteration are reduced so that efficiency and robustness are improved.

**Key words:** target tracking; occlusion; Mean Shift; ELM