

中心对称纹理描述及其在目标跟踪中的应用

程 国

(商洛学院 数学与计算机应用学院, 陕西 商洛 726000)

摘要: 为解决目标跟踪过程中目标表示鲁棒性和实时性的问题, 构建了一种全新的纹理描述算子, 并以此完成目标表示进而实现鲁棒的目标跟踪。该算子通过计算邻域像素与中心像素相似的数量, 并充分利用中心像素的局部梯度信息, 构成一组 3 维向量完成纹理描述, 统计真实目标区域内像素的纹理信息完成目标表示, 同时采用一种简单的在线机制更新目标模板, 进而嵌入到粒子滤波框架完成目标跟踪。实验结果表明, 该方法在光照变化、姿态变化、遮挡、运动模糊等情况下均能完成鲁棒跟踪, 平均跟踪成功率为 87.9%, 表明该算子具有良好的旋转不变性和光照不变性, 对复杂纹理结构有较强的区分能力。

关键词: 目标跟踪; 纹理特征; 目标表示; 粒子滤波

中图分类号: TP391

文献标识码: A

文章编号: 1001-8891(2018)05-0424-07

Center Symmetric Texture Operator and Its Application in Object Tracking

CHENG Guo

(College of mathematics and computer application, Shangluo University, Shangluo 726000, China)

Abstract: Aiming at the robustness and efficiency of target representation in tracking process, a new texture description operator is constructed and then a robust tracking algorithm based on it is completed. This algorithm completes texture description in a group of three dimensional vectors by computing the number of pixels in the center similar to that in the neighborhood and making full use of local gradient information of center pixel, then the operator uses the texture information of the true pixels in the target to complete target representation. At the same time, a simple online mechanism to update the target template is adopted. Finally tracking is accomplished by embedding target representation into the framework of particle filter. Experimental results show that proposed algorithm can achieve a robust tracking under the situation of illumination changing, pose changing, occlusion and motion blur, the average tracking success rate is 87.9%, indicating that the operator has good rotation invariance, illumination invariance, and has the strong ability to distinguish complex texture structure.

Key words: target tracking, texture features, objective representation, particle filter

0 引言

运动目标跟踪在自动侦察、视觉导航、人机交互等领域的广泛应用已使其成了计算机视觉领域的一个研究热点^[1]。尽管大量算法已被提出, 但诸如光照变化、姿态变化、遮挡、运动模糊等因素的影响, 设计一个鲁棒的跟踪算法依然具有很大的挑战性。

假设目标在第一帧的状态已知, 跟踪可以建模成一个状态估计问题, 而目标表示则对观测模型的准确性起着至关重要的作用, 文献[2]采用颜色特征构建直

方图完成目标表示, 但易受光照变化影响, Tavakoli^[3]提出局部相似数量 (local similarity number, LSN) 这一纹理特征并结合颜色特征构建联合直方图减轻了光照变化的影响, 此外基于候选区域特征点及相互之间的结构关系也是当前目标表示的一个热点^[4-5], 但对特征点的数量要求较高, Babenko 等^[6]利用 Harr-like 特征并结合多样例学习机制完成目标表示, 在此基础上, 张开华等^[7]基于压缩感知理论提取 Harr-like 特征完成目标表示, 也取得了不错的跟踪效果, 近年来稀疏表示^[8-10]也成了跟踪领域的一个研究热点, 这类算

收稿日期: 2016-04-18; 修订日期: 2016-06-23.

作者简介: 程国 (1983-), 男, 硕士, 副教授, 研究方向: 图像处理与模式识别、数值代数。

基金项目: 陕西省教育厅自然科学研究计划项目 (17JK0240); 商洛学院科研项目 (16SKY008)。

法将目标表示成一系列模板的线性组合,但算法的计算复杂度较高影响了跟踪实时性,基于深度学习的目标跟踪^[11-13]通过大量图像集的线下训练得到判别性较高的特征来完成目标表示,但跟踪前的线下训练耗时较高,不利于目标跟踪的实时应用。

LBP (local binary patterns) 纹理特征由于不受光照变化影响,在目标跟踪得到了广泛应用^[14-15]。本文受文献[16]启发,利用以中心像素为对称点的局部梯度信息,提出了中心对称纹理描述 (center symmetric texture operator, CSTO) 这一新的局部相似度纹理描述算子,并通过生成 CSTO 掩膜提取真实目标像素完成目标表示,最后结合粒子滤波 (particle filter) 框架实现了一个鲁棒的目标跟踪算法。

1 局部相似度纹理描述算子

1.1 局部相似数量

Tavakoli^[3]在原始 LBP 的基础上,提出了一种纹理描述算子的简化模型,通过中心像素 g_c 及其以 R 为半径的 P 个邻域像素 g_i ($i=0, \dots, P-1$) 的差值大小建立模型,如果差值大小小于一定阈值 d ,则意味该邻域像素与中心像素相似,贴标签为 1,否则贴标签为 0,模型定义为 LSP (local similarity pattern):

$$\text{LSP} = \sum_{i=0}^{P-1} f(g_i - g_c, d) 2^i, \quad (1)$$

$$f(x, d) = \begin{cases} 1, & |x| \leq d \\ 0, & |x| > d \end{cases}$$

以 $P=8$ 为例, LSP 将产生 256 种模式, Tavakoli^[3]认为这些模式中包含的结构信息对于计算中心像素的显著程度并无帮助,因此计算与中心像素相似的邻域像素个数即可,用 2^0 代替式(1)中的 2^i ,模型简化为 LSN:

$$\text{LSN} = \sum_{i=0}^{P-1} f(g_i - g_c, d), \quad (2)$$

$$f(x, d) = \begin{cases} 1, & |x| \leq d \\ 0, & |x| > d \end{cases}$$

1.2 中心对称局部相似数量

从以上分析可知 LSN 极大提高了算子的效率,但同时也弱化了算子对复杂纹理结构的描述能力,尤其体现在中心像素的局部相似数量较低时(如边缘、角点等), LSN 将无法区分不同的纹理结构。而文献[16]利用了以中心像素为对称点的局部梯度信息,通过比较邻域像素对的大小来描述中心像素,提高了 LBP 算子的纹理描述能力,如图 1 所示。

在图 1 右侧框内可以看出 CS-LBP 算子主要是利用了以中心像素为对称点的 4 组像素点对的信息,为

此可以充分利用中心像素的局部梯度信息来提高算子对复杂纹理结构的描述能力,在 LSN 的基础上增加以中心像素为对称点的像素对的差值关系,来解决 LSN 算子虽然计算复杂度降低但复杂纹理结构区分度不够的问题,如图 2 所示。

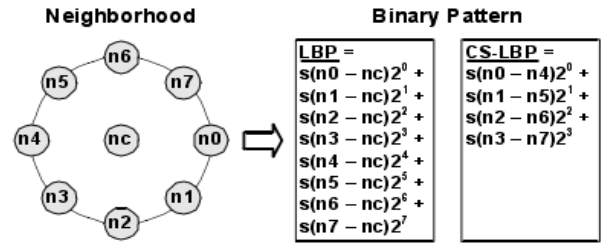


图 1 LBP 和 CS-LBP 算子 (以 8 邻域为例)

Fig.1 LBP and CS-LBP operators (Take 8 neighborhood as an example)

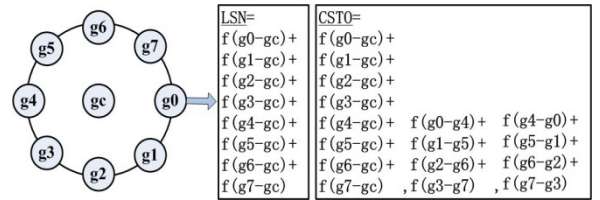


图 2 LSN 和 CSTO 算子 (以 8 邻域为例)

Fig.2 LSN and CSTO operators (Take 8 neighborhood as an example)

图 2 中的 CSTO 第 1、2、3 列的算子可表示成 $f_o(x, d)$ 、 $f_r(x, d)$ 、 $f_l(x, d)$ 。 $f_o(x, d)$ 描述中心像素与邻域像素的相似数量信息; $f_r(x, d)$ 、 $f_l(x, d)$ 描述以中心像素为对称点的邻域像素对的梯度信息,当像素对的差值大于阈值 d 时说明中心像素处于角点边缘等位置。因此本文定义 CSTO 为一组 3 维向量:

$$\text{CSTO} = \begin{pmatrix} \sum_{i=0}^{P-1} f_o(g_i - g_c, d) \\ \sum_{i=0}^{P/2-1} f_r(g_i - g_{i+P/2}, d) \\ \sum_{i=0}^{P/2-1} f_l(g_{i+P/2} - g_i, d) \end{pmatrix}^T \quad (3)$$

式中:

$$f_o(x, d) = \begin{cases} 1, & |x| \leq d \\ 0, & |x| > d \end{cases}$$

$$f_r(x, d) = \begin{cases} 1, & x \geq d \\ 0, & x < d \end{cases} \quad (4)$$

$$f_l(x, d) = \begin{cases} 1, & x \geq d \\ 0, & x < d \end{cases}$$

式(3)表明: CSTO 中第 1 维是原始 LSN 算子中心像素的局部相似数量,第 2、3 维向量提取了中心像

素的局部梯度信息。下面以 $P=8$ 、 $R=1$ 、 $d=2$ 为例（下同），说明提出的 CSTO 算子的对不同纹理结构的描述能力，如图 3 所示：示例 1 和示例 2 的中心像素具有不同的纹理结构，LSN 算子计算得到的局部相似数量均为 3，本文 CSTO 算子的 3 维向量描述分别为 $(3,2,2)$ 和 $(3,2,1)$ 。对于 8 邻域而言，CSTO 算子共产生 225 $(9 \times 5 \times 5)$ 种不同模式，较 LSN 算子的 9 种模式能将不同的纹理结构更好加以区分。

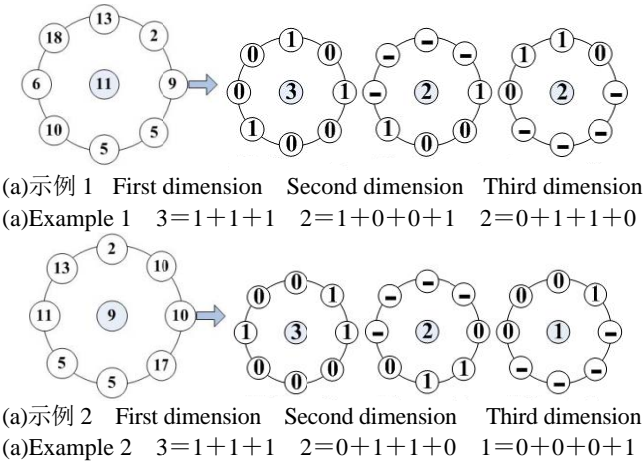


图 3 LSN 和 CSTO 算子生成过程

Fig.3 The generative process of LSN and CSTO operators

2 基于 CSTO 的目标表示

2.1 CSTO 掩膜生成

角点、边缘等区域像素较平坦区域能更好地表示目标，因此如果能保留角点、边缘信息的同时去除平坦区域信息，势必能更高效且准确地完成目标表示，为此本文定义 CSTO 掩膜为：

$$M_{\text{CSTO}} = \begin{cases} 1, & \sum_{i=0}^7 f_o(g_i - g_c, d) < 5 \text{ and } \left[\sum_{i=0}^3 f_c(g_i - g_{i+4}, d) > 0 \right. \\ & \left. \text{or } \sum_{i=0}^3 f_c(g_i - g_{i+4}, d) > 0 \right] \\ 0, & \text{otherwise} \end{cases} \quad (5)$$

以文献[14]提供的测试视频 *tabletennis.avi*（共 58 帧）为实验数据，用 CSTO 掩膜提取目标框中的真实目标像素，图 4 给出了掩膜提取目标像素的一个示例。式(5)中的阈值 d 刻画了像素之间的相似度，太大不具有区分性，实验表明不同阈值 d 将产生不同的提取效果，本文对 $d \in \{1, 2, 3, 4, 5\}$ 的提取效果进行评估，评估标准一是准确率即提取的像素中属于真实目标区域的比率），二是召回率（即真实目标区域中被掩膜提取的像素所占的比率），图 5 实验数据表明阈值取 2 或 3 时能得到较好的提取效果。从图 4 可以发现原始目标框中目标四周背景部分较为平坦，各像素点局部

相似度较高，而目标部分（人脸）则纹理结构相对较为复杂，各像素点局部相似度较低，通过本文的 CSTO 掩膜可以将局部相似度较高的像素位置设置为 0，不加入目标表示的计算，因此目标边缘、角点等信息得到了较好保留，且目标周围背景像素由于区域较为平坦而得到了很好地抑制，因此增加了目标表示模型的表征能力。



图 4 CSTO 掩膜提取目标像素示例

Fig.4 The example of extracting target pixels by CSTO mask

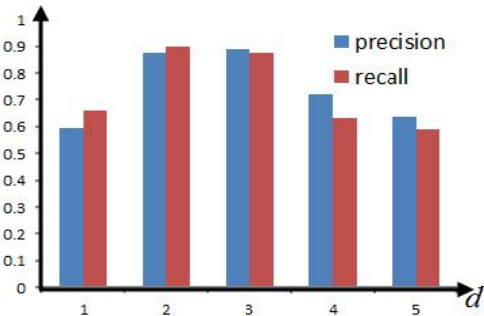


图 5 不同阈值 d 提取目标像素效果分析

Fig.5 Analysis of target pixel effects extracted under different threshold d

2.2 目标表示

在第一帧框选要跟踪的目标后，对 CSTO 掩膜提取的框选区域中的真实目标像素进行纹理信息统计，由于式(3)中 CSTO 算子是一组 3 维向量，每一维分别有 9、5、5 种模式，则目标特征空间由 $m=9 \times 5 \times 5$ 个分量构成，为了增强目标表示的可靠性，我们为离框选目标中心较近的像素点信息赋予较高的权重，采用 Epanechnikov 核函数^[2]：

$$K(x) = \begin{cases} 1 - \|x\|^2 & \|x\| \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

为框选区域内像素点与中心的距离。于是目标模型 $P = \{p(u)\}_{u=1,2,\dots,m} \in \mathbf{R}^m$ 表示为：

$$p(u) = C \sum_{i=1}^n k \left(\left\| \frac{x_i - x_0}{h} \right\| \right) \delta[b(x_i) - \mu]$$

$$C = 1 / \sum_{i=1}^n k \left(\left\| \frac{x_i - x_0}{h} \right\| \right) \quad (6)$$

$$h = \sqrt{(t_x / 2)^2 + (t_y / 2)^2}$$

式中: n 为框选区域中的真实目标像素个数; x_i 为各像素坐标; x_0 为目标中心坐标; t_x 和 t_y 分别是框选区域宽和高; $\delta[b(x_i) - \mu]$ 为 Kronecker delta 函数: 判断 x_i 处的纹理信息是否属于第 μ 个分量。为适应目标外观在跟踪过程中的不断变化, 目标模型 P 应该得到实时更新, 为此我们采用如下更新机制:

$$P_t = \lambda P_{t-1} + (1 - \lambda) Q_{t-1} \quad (7)$$

式中: P_t 是第 t 帧的目标模型; P_{t-1} 是第 $t-1$ 帧的目标模型; Q_{t-1} 是第 $t-1$ 帧的跟踪结果的目标表示; $\lambda \in (0.5, 1)$ 是学习率。这一在线更新机制不仅考虑到了目标外观的变化, 而且保留了之前帧稳定的目标表示。

3 粒子滤波跟踪框架

完成目标表示后, 文献[3]采用均值漂移 (mean shift) 搜索机制完成跟踪, 但 mean shift 容易导致局部收敛无法搜索到最佳跟踪效果, 因此本文跟踪算法建立在粒子滤波跟踪框架下, 给定前 t 帧的观测序列 $O_t = \{o_1, \dots, o_t\}$, 根据贝叶斯滤波理论, 跟踪可以看成是一个状态估计问题, 即完成后验概率的估计:

$$p(s_t | O_t) \propto p(o_t | s_t) \int p(s_t | s_{t-1}) p(s_{t-1} | O_{t-1}) ds_{t-1} \quad (8)$$

式中: $s_t = (x_t, y_t)$ 是当前目标状态; x_t 、 y_t 分别为沿 x 和 y 方向的转移量; $p(s_t | s_{t-1})$ 是运动模型: 根据前一帧目标状态 s_{t-1} 预测当前帧目标状态 s_t , 假设目标状态参数相互独立, 因此可以以高斯分布来建模: $p(s_t | s_{t-1}) = N(s_t | s_{t-1}, \Sigma)$, $\Sigma = \text{diag}(\sigma_x, \sigma_y)$; $p(o_t | s_t)$ 是观测模型: 目标状态 s_t 下的观测 o_t 与目标的相似程度。而粒子滤波器正是通过带有重要性权重的粒子集合 $\{s_t^i, \pi_t^i\}_{i=1}^N$ 来近似式(8)的积分式得到后验概率, 其中 N 个粒子的重要性权重 $\pi_t^i \propto p(o_t | s_t^i)$, 因此最优目标状态可以通过最大化下式得到:

$$\hat{s}_t = \arg \max_{\{s_t^i\}_{i=1}^N} p(o_t | s_t^i) p(s_t^i | \hat{s}_{t-1}) \quad (9)$$

式(9)中的观测模型 $p(o_t | s_t^i)$ 在跟踪任务中起着至关重要的作用, 依据 2.2 的目标表示构建为:

$$p(o_t | s_t^i) \propto e^{-|1 - \rho(P_t, Q_t^i)|} \quad (10)$$

式中: P_t 是第 t 帧的目标模型; $Q_t^i = \{q_t^i(u)\}_{u=1,2,\dots,m} \in \mathbf{R}^m$ 是第 t 帧中第 i 个粒子采样样本的目标表示;

$\rho(P_t, Q_t^i) = \sum_{u=1}^m \sqrt{p_t(u) q_t^i(u)}$ 表示二者之间的巴氏系数, 越大表示二者越相似。

4 实验与结果分析

选用文献[17-18]提供的具有不同挑战性的标准测试视频序列 (共 2847 帧) 来评估跟踪算法的性能, 序列所具特性如表 1 所示。

表 1 实验视频的不同特性

Video name	Frame number	Key characteristic
Car	365	Similar goals, motion blur
Suv	946	Partial occlusion, complete occlusion
David	251	Light, perspective, posture
Faceocc2	812	Partial occlusion, attitude, angle of view
Crossing	120	Light, scale, posture
Tiger1	353	Partial occlusion, angle of view, motion blur

为验证 CSTO 算子的目标表征能力及跟踪算法的鲁棒性, 分别将采用 CSTO 算子和 particle filter 框架的跟踪 (CSTO+PF)、采用 CSTO 算子和 mean shift 框架的跟踪 (CSTO+MS)、采用 LSN 算子和 mean shift 框架的跟踪^[3] (LSN+MS) 以及当前比较火热的基于压缩感知的跟踪^[7] (CT) 和基于多示例学习的跟踪^[6] (MIL) 进行对比分析。实验平台采用 Intel Core (TM) 2 Duo CPU 主频 2.5 GHz 的 PC 机, 利用 OpenCV2.4.9 在 VS2010 上进行调试。

4.1 参数设置

CSTO 算子: 采用半径 $R=8$ 的 $P=8$ 个邻域像素, 像素相似阈值 $d=2$; 目标模型中学习率: Car、Suv、Crossing 三个视频中目标在跟踪过程姿态视角等变化不大故设定为 $\lambda=0.9$, David、Faceocc2、Tiger1 三个视频中目标在跟踪过程姿态视角等变化较大故设定为 $\lambda=0.7$; particle filter 跟踪框架中目标状态中标准差设为 $\sigma_x=4$ 、 $\sigma_y=4$, 粒子数 $N=200$; mean shift 框架中最大迭代次数设为 50 (参照文献[13])。

4.2 结果分析

实验采用常见的两个标准^[7]对算法性能进行评估, 一是跟踪成功率 (success rate, SR), 定义 $sr = \frac{R_T \cap R_G}{R_T \cup R_G}$ 代表每一帧跟踪结果的得分, 其中 R_T 代表跟踪结果区域, R_G 代表目标真实区域, 若 $sr > 0.5$ 则认为跟踪

结果正确；二是中心位置误差（center location error, CLE），定义为：

$$E_{cl}=\sqrt{(x_T-x_G)^2+(y_T-y_G)^2}$$

式中：(x_T,y_T)是跟踪结果的目标位置；(x_G,y_G)是目标真实位置。

表2给出了算法在实验视频中的成功率及平均跟踪速度（average frames per second, FPS），加粗代表最优性能，加下划线代表次优性能。可见本文采用的CSTO+PF与其它算法相比基本获得了最优或次优性能，除了在Faceocc2序列中成功率为89.3%，低于CT的100%和MIL的99.0%，而本文CSTO+MS得益于CSTO的纹理描述能力平均跟踪成功率较文献[3]的LSN+MS提高了31.7%，在所有对比算法中排名第2，仅次于本文CSTO+PF的87.9%。表2最后一行显示本文CSTO+PF的平均跟踪速度为15.1 Frame/s，逊于CSTO+MS的25.7 Frame/s，但平均跟踪成功率提高了4.7%，原因在于particle filter框架比mean shift框架耗时更多，但避免了mean shift局部收敛的问题，因此获得了最佳的跟踪效果。

表3给出了算法在实验视频中的跟踪误差对比，其中Max代表中心位置误差最大值、Mean代表中心位置误差平均值，Std代表中心位置误差标准差，加粗代表最优性能，斜体加下划线代表次优性能。由于Suv序列中目标在被全局遮挡后，LSN+MS、CT、MIL三种方法完全跟丢目标，故不再统计算法中心位置误差。同样可见，在中心位置误差最大值（Max）下CSTO+PF全部获得了最优或次优性能，在中心位置误差平均值（Mean）下CSTO+PF在除了David序列外基本获得了最优或次优性能，而且在中心位置误差标准差（Std）下CSTO+PF除了David序列都获得了最优性能，这意味着CSTO+PF能获得准确且更稳定的性能。

图6给出了本文算法的部分跟踪结果（左上角数

字代表帧序数），下面针对视频不同特性对本文算法的鲁棒性进行分析。相似目标和运动模糊：Car序列是一段航拍视频，航拍平台的剧烈运动导致运动模糊的出现，跟踪目标为中间白色小车，前后各有相似目标且目标与背景不易区分，Tiger1序列中第100~150帧左右目标的快速运动同样导致了运动模糊，算法的全程鲁棒跟踪体现了CSTO算子的纹理描述能力；遮挡：Suv、Faceocc2和Tiger1序列均出现了不同程度的遮挡，Suv序列第206帧左右目标被完全遮挡，当目标重新出现在场景中时，算法依然能重新定位到目标，Faceocc2序列第686帧当人脸被纹理复杂的书本局部遮挡时，算法的跟踪误差偏大，这里可以结合其他目标信息（如色度）加以改进；光照变化：David、Crossing序列中出现了不同程度的光照变化，当视频中目标逐渐从暗处移动到亮处是，算法基本能够很好定位到目标（David序列第31帧左右出现了较大偏差），这得益于CSTO算子采用局部相似程度来描述具有良好的光照不变性；视角和姿态变化：David、Faceocc2、Crossing和Tiger1序列中目标姿态出现了不同程度的变化，如David序列第169帧左右、Faceocc2序列第339帧左右人脸均发生了旋转，Crossing、Tiger1序列中目标姿态一直处于不断变化过程中，算法能鲁棒完成跟踪任务是因为CSTO算子采用圆形区域采样具有良好的旋转不变性，但在Crossing序列中随着目标远离场景过程中尺度逐渐变小，算法的跟踪误差逐渐偏大以致丢失目标（如第119帧），下一步拟结合现有的尺度更新机制解决此类问题。

5 结论

从提高目标表示能力出发，提出一种新的纹理描述算子，并代入粒子滤波框架完成跟踪。在具有不同挑战性的视频上的实验表明，本文方法在光照变化、姿态变化、遮挡、运动模糊等情况下较其它算法获得了准确且更稳定的跟踪性能：平均跟踪成功率达到

表2 跟踪成功率（SR）和平均跟踪速度（FPS）对比

Table 2 Comparison between success rate (SR) and average frames per second (FPS)					
Sequence	CSTO+PF	CSTO+MS	LSN+MS	CT	MIL
Car	100	100	<u>90.1</u>	100	88.9
Suv	98.9	<u>97.2</u>	37.5	35.3	33.2
David	<u>86.6</u>	80.3	50.2	89.1	68.2
Faceocc2	89.3	81.1	56.9	100	<u>99.0</u>
Crossing	88.7	<u>83.5</u>	50.8	81	67.2
Tiger1	<u>64.1</u>	57.4	22.3	78.1	39.3
Average SR	87.9	<u>83.2</u>	51.5	80.6	66.0
Average FPS	15.1	25.7	<u>30.3</u>	35.2	10.5

表3 中心位置误差 (CLE) 对比 Table 3 Center location error (CLE) comparison

Sequence	CSTO+PF			CSTO+MS			LSN+MS			CT			MIL		
	Max	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max	Mean	Std	Max	Mean	Std
Car	1.6	1.2	0.6	<u>1.9</u>	<u>1.6</u>	0.9	10.1	5.3	4.4	<u>1.9</u>	1.2	<u>0.7</u>	2.7	1.9	1.5
Suv	15.4	7.8	2.4	<u>20.4</u>	<u>10.2</u>	<u>5.0</u>	—	—	—	—	—	—	—	—	—
David	<u>20.2</u>	16.7	6.1	22.5	<u>16.5</u>	10.1	64.8	56.3	25.9	18.1	16.2	4.0	41.3	18.5	<u>5.2</u>
Faceocc2	<u>17.4</u>	<u>10.5</u>	3.3	23.3	18.7	9.0	39.9	23.5	11.8	15.4	10.0	<u>4.4</u>	36.2	17.1	7.2
Crossing	22.4	7.9	5.5	<u>29.1</u>	12.3	<u>7.4</u>	54.7	29.6	21.2	42.3	<u>9.0</u>	11.2	51.3	11.3	19.7
Tiger1	19.2	<u>7.4</u>	7.2	29	10.2	7.8	70.2	20.6	23.1	20.1	6.0	<u>5.2</u>	116.2	27.1	28.4

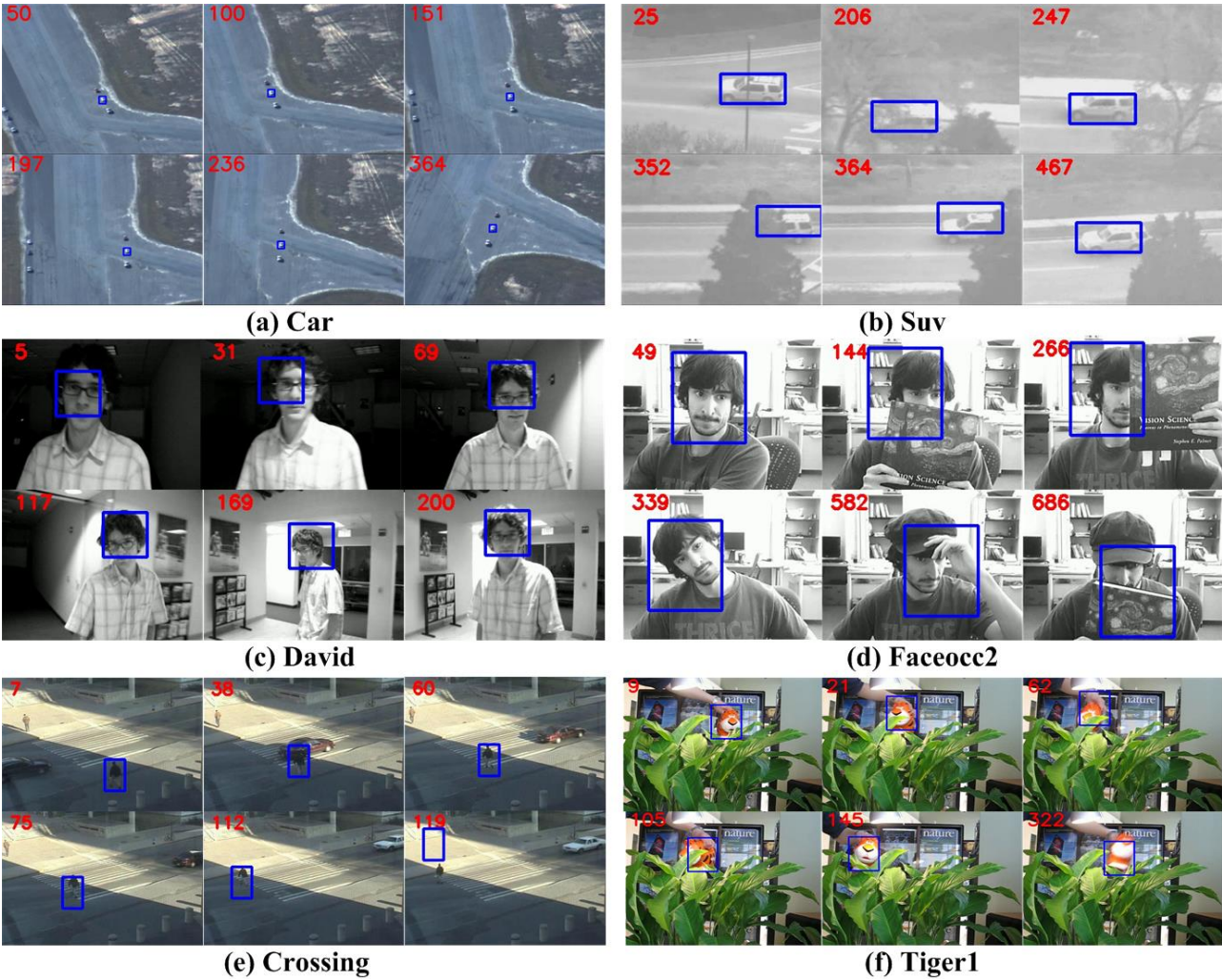


图6 本文算法的部分跟踪结果 Fig.6 Partial tracking results of the proposed algorithm

87.9%，平均跟踪速度为 15.1 Frame/s。下一步拟打算优化粒子滤波框架以进一步提高算法实时性，并将尺度更新和遮挡判断机制嵌入到本文算法框架中进一步提高算法的鲁棒性。

参考文献:

[1] Emilio M. *Video Tracking: Theory and Practice*[M]. London: Queen Mary University of London, UK, 2012: 15-24.
[2] Comamiciu D, Ramesh V, Meer P. Kernel-Based object tracking[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2003, **25**(5): 564-577.

[3] Tavakoli H R, Moin M S, Heikkilä J. Local similarity number and its application to object tracking[J]. *International Journal of Advance Robotic Systems*, 2013, **10**(1): 184-190.
[4] Georg Nebehay, Roman Pflugfelder. Consensus-based matching and tracking of keypoints for object tracking[C]//*IEEE Winter Conference on Applications of Computer Vision*, 2014: 862-869.
[5] Sam Hare, Amir Saffari, Philip H S Torr. Efficient online structured output learning for keypoint-based object tracking[C]//*IEEE Conference on Computer Vision and Pattern Recognition*, 2012: 1894-1901.
[6] Babenko B, Yang M H, Belongie S. Robust object tracking with online multiple instance learning[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, **33**(8): 1619-1632.

- [7] ZHANG K H, ZHANG L, YANG M H. Real-time compressive tracking[C]// *European Conference on Computer Vision*, 2012: 864-877.
- [8] MEI X, LING H. Robust visual tracking and vehicle classification via sparse representation[J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2011, **33**: 2259-2272.
- [9] BAO C L, WU Y, LING H B, et al. Real time robust L1 tracker using accelerated proximal gradient approach[C]// *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, **157**(10): 1830-1837.
- [10] ZHONG W, LU H C, YANG M H. Robust object tracking via sparsity-based collaborative model[C]// *IEEE Conference on Computer Vision and Pattern Recognition*, 2012, **157**(10): 1838-1845.
- [11] GAO J, LING H, HU W, et al. Transfer learning based visual tracking with Gaussian processes regression[C]// *European Conference on Computer Vision*, 2014, **8691**: 188-203.
- [12] WANG N, YEUNG D Y. Learning a deep compact image representation for visual tracking[J]. *Neural Information Processing Systems*, 2013: 809-817.
- [13] WANG Q, CHEN F, YANG J, et al. Transferring visual prior for online object tracking[J]. *IEEE Transactions on Image Processing*, 2012, **21**(7): 3296-3305.
- [14] NING J F, ZHANG L, ZHANG D, et al. Robust object tracking using joint color-texture histogram[J]. *International Journal of Pattern Recognition and Artificial Intelligence*, 2009, **23**(7): 1245-1263.
- [15] 张炯, 宁继峰, 颜永丰, 等. 对立色 LBP 模型的目标跟踪[J]. *中国图象图形学报*, 2012, **17**(11): 1418-1424.
ZHANG Jiong, NING Jifeng, YAN Yongfeng, et al. Object tracking with opponent color LBP model[J]. *Journal of Image and Graphics*, 2012, **17**(11): 1418-1424.
- [16] Heikkila M, Pietikainen M, Schmid C. Description of interest regions with local binary patterns[J]. *Pattern Recognition*, 2009, **42**(3): 425-436.
- [17] WU Y, LIM J, YANG M H. Online object tracking: A benchmark[C]// *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, **9**(4): 2411-2418.
- [18] 杨智雄, 余春超, 严敏, 等. 基于特征融合的粒子滤波红外目标跟踪算法[J]. *红外技术*, 2016, **38**(3): 211-217.
YANG Zhixiong, YU Chunchao, YAN min, et al. Particle filter infrared target tracking algorithm based on feature fusion[J]. *Infrared Technology*, 2016, **38**(3): 211-217.