第 34 卷第 6 期
2017 年 6 月

控 制 理 论 与 应 用
Control Theory & Applications

Vol. 34 No. 6
Jun. 2017

DOI: 10.7641/CTA.2017.60539

多特征融合的实时人手跟踪算法

李 逢[1], 桑 农[1†], 王洪智[1], 颜 轶[1], 高常鑫[1], 刘乐元[2]

(1. 华中科技大学 自动化学院 图像信息处理与智能控制教育部重点实验室, 湖北 武汉 430074;
2. 华中师范大学 国家数字化学习工程技术研究中心, 湖北 武汉 430079)

摘要: 由于复杂背景、形变以及运动造成的模糊等因素, 导致在自然场景下的人手跟踪仍然是一个具有挑战性的问题. 本文中, 结合运动、颜色和Haar-like特征来构造一个具有鲁棒性的实时人手检测算法. 尽管不能运用于所有的情形, 但Haar-like特征成功地去除了类似肤色的运动背景区域. 利用三个特征构造三个弱分类器, 然后将其结合成一个强分类器. 如果一个分类器已经确定了人手的位置, 其他分类器将不会执行, 否则将会为下一个分类器提供一个可能的区域. 文中实现了提出的算法, 并且在几个具有挑战性的视频序列上进行了实验.

关键词: 人手跟踪; 多特征; 实时性; 人机交互

中图分类号: TP273　　**文献标识码:** A

Real-time and robust hand tracking using multiple features

LI Feng[1], SANG Nong[1†], WANG Hong-zhi[1], YAN Yi[1], GAO Chang-xin[1], LIU Le-yuan[2]

(1. National Key Laboratory of Science and Technology on Multispectral Information Processing, School of Automation,
Huazhong University of Science and Technology, Wuhan Hubei 430074, China;
2. National Engineering Research Center for E-Learning, Central China Normal University, Wuhan Hubei 430079, China)

Abstract: Hand tracking in unconstrained environments remains an extremely challenging problem due to several factors, such as background clutter, deformation, and motion blur. In this paper, we combine motion, color, and Haar-like features to construct a real-time and robust hand tracking system. Haar-like features successfully defeat moving skin-colored backgrounds, although they are unstable for the whole situation. Three weak trackers are built using each kind of feature and integrated in a boosted cascade. If one stage makes sure of the object position, no other stages is carried out. Otherwise it provides its own point of view to guide the next stage. We realize the proposed approach and demonstrate it on several challenging sequences.

Key words: hand tracking; multiple features; real-time; human computer interaction

1 Introduction

Hand tracking is important in many human computer interaction applications and has been intensely studied during the past decades[1–8]. In this paper, we focus on the scene captured by a motionless normal camera viewing possibly multiple people, as shown in Fig.1(a). It is common in the remote control based on gesture recognition for desktop, television, and some other terminals. Real-time and robust hand tracking in this scene remains an extremely challenging problem due to the following factors:

Background Clutter. The user probably wears a short sleeve T-shirt or moves his hand in front of the face. There could also be someone else sitting or walking behind the tracked hand. All the bare skin areas have the similar color as the object.

Deformation. The human hand is highly articulated with complex finger interactions and probably undergoes significant deformation during tracking.

Motion Blur. The hand region is occasionally blurred due to the lighting condition and the object motion. The features based on gradient do not perform well in this case, such as optical flow[9–10], edge[11], and orientation histograms[12].

Values in mapping images indicate probabilities of pixels being the object center and are shown by coloring. The color from blue to red represents the value from low to high.

A large number of excellent works have been made to overcome these difficulties. Freeman et al.[1] use the open hand templates based on the local orientation representation and perform the normalized correlation in moving regions extracted by background subtraction. They do not take into account the variety of hand shapes and the orientation representation is sensitive to motion blur. Argyros et al.[2] build a skin color model in the Bayesian framework and adopt the region growing operation to detect skin-colored blobs. The track-

Received 23 July 2016; accepted 21 December 2016.

[†] Corresponding author. E-mail: nsang@hust.edu.cn; Tel.: +86 13886014687.

Recommended by Associate Editor YU Zhu-liang.

er is robust to rotation, deformation, and motion blur, but struggles in the background where there are something of similar color. Krahnstoever et al.[4] combine motion and color cues to remove the unmoving skin-colored distractors. Stenger et al.[5] start with template matching based on the normalized cross-correlation and in case of failure use [4] as a fall-back strategy. The tracker performs well even when a face moves behind the hand, but could only track the frontal fist. Spruyt et al.[13] select local binary pattern, gradient orientation, and three color based features, combined with random forest classifier which is trained off-line to obtain a hough forest detector, so the algorithm performance depends much on the selected features and training data. In addition, there are many tracking algorithms for arbitrary objects[14–21]. Grabner et al.[16] propose an on-line AdaBoost framework to select the most discriminating features adaptively, which allows to update features of the classifier during tracking. Sam Hare et al.[18] uses a kernelized structured output support vector machine, which is learned online to provide adaptive tracking. Zdenek Kalal et al.[19] propose a novel tracking framework (TLD) that explicitly decomposes the long-term tracking task into tracking, learning and detection and the detector localizes all appearances that have been observed so far and corrects the tracker if necessary. João et al.[21] adopt discrete fourier transform to reduce both storage and computation and derive a new kernelized correlation filter which has the exact same complexity as its linear counterpart. They actually use Haar-like features[22], orientation histograms, and local binary patterns[23] to achieve inspiring performance on their test sequences. Nevertheless it is not effective enough for robust hand tracking in unconstrained environments.

Tracking success or failure essentially depends on the difference between the object and background in the feature space. Different parts of background have different types of difference from the object and should be distinguished using different types of feature. Motion is a valuable feature which has been used in hand pose recognition[24], as it is robust under different conditions. A strength image based on motion is produced using optical flow, as shown in Fig.1(b). Color is also an applicable feature for hand tracking as the hand is uniform in color, and color has proved useful for hand tracking and hand pose estimation in previous work[25–27]. A strength image based on color is produced using Gaussian mixture model (GMM), as shown in Fig.1(c). The remaining question is searching for another feature to defeat moving skin-colored backgrounds. We find that Haar-like features are quite qualified for this task in our experiments. A strength image based on Haar-like features is produced using the on-line AdaBoost algorithm, as shown in Fig.1(d). It is hard to locate the tracked hand according to any one of the three strength images. We suppose that the three kinds of features are independent and compute a fused strength image using the product of them, as shown in Fig.1(e). The position of the tracked hand is easy to be determined in this image, which demonstrates the efficacy of fusing these three kinds of features. In this paper, we combine motion, color, and Haar-like features to build a real-time and robust hand tracking system. Details are given in Section 2. Experimental results in comparison with the state-of-the-art algorithms are given in Section 3. Finally we report some conclusions in Section 4.

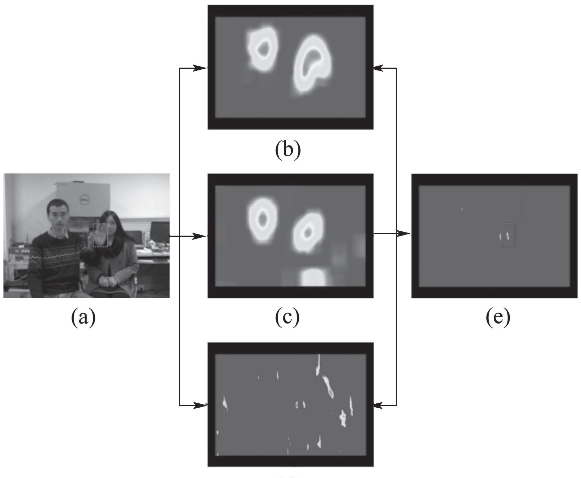

Fig. 1 (a) A frame in the test sequences. (b), (c) and (d) Three mapping images of hand-likelihood based on motion, color, and Haar-like features. (e) A fused mapping image using the product of (b), (c) and (d).

2 Approach

This section introduces how to combine motion, color, and Haar-like features to construct a real-time and robust hand tracking system. We use each kind of feature to build three weak trackers and then integrate them in a boosted cascade.

2.1 Motion

In general, there are three kinds of methods for motion detection, such as frame difference, background subtraction, and optical flow. Frame difference has the minimum computational complexity, but hardly gets the whole moving region without missing pieces. The key problem of background subtraction is background modeling. However it is difficult to maintain an effective model during tracking, especially there are large-scale changes in the background. Chan et al.[28] propose an adaptive model based on a generalization of the Stauffer-Grimson background model for backgrounds containing motion. Pham et al.[29] detect dynamic background with swaying movements from motions features. These approaches are not suitable for background modeling in hand tracking, as the background is still most of time. Farnebäck[9] estimates the optical flow velocities between two frames based on polynomial expansion. This algorithm could not only detect moving pixels but also calculate their displacements. It is found to be a good compromise between speed and accuracy in our experiments. Hence we adopt it to extract motion information in our proposed approach.

The object rectangle in the previous frame is available and then a concentric expanded rectangle region is defined as the search range. The optical flow velocities at each pixel in this range are computed when the new frame comes. We count the number of nonzero velocities only in the previous object rectangle. If they do not take the dominant role, the object is considered to be motionless. Otherwise the object displacement could be estimated using an average of them. Unfortunately it is unstable and easily causes tracking drift in most cases. Instead of this estimation, we compute a binary motion mask indicating whether the pixel is moving or not in this paper. For viewing convenience, a motion mask computed in the whole image is filtered by the average template with the same size as the object rectangle, as shown by coloring in Fig.1(b).

2.2 Color

Skin color is a significant feature which can used to easily distinguish between the hand and most of other backgrounds, especially the hand is uniform in color. Hence color has been widely used for hand tracking. In general, there are three kinds of skin color models, such as color histogram, Gaussian model, and GMM. It is difficult to establish adequate color histograms with a balance between efficiency and accuracy in practice. The GMM is better than the Gaussian model since the skin color distribution could be multimodal under the influence of illumination. After the object is located, the GMM for the object $F_{\rm O}$ is established based on the pixels in the object rectangle. And the GMM for the background $F_{\rm B}$ is established based on the pixels in the remaining region within the search range. For each new frame, a binary color mask is computed using the G-MMs as

$$\text{Mas}k_{\text{color}}(x) = \begin{cases} 1, \ p(y\,|F_{\rm O}, x) > p(y\,|F_{\rm B}, x), \\ 0, \ \text{otherwise}, \end{cases} \tag{1}$$

where y is the color value at the pixel x. $p(\cdot)$ is the Gaussian mixture distribution as

$$p(y|F) = \sum_{i=1}^{k} \frac{\pi_i}{\sqrt{\varSigma_i}} \exp\{-\frac{1}{2}(y-\mu_i)^{\rm T}\varSigma_i^{-1}(y-\mu_i)\}, \tag{2}$$

where k is the number of components, π_i is the weighting coefficient of the ith component, μ_i is the mean and $\varSigma_i$ is the covariance matrix. For viewing convenience, a color mask computed in the whole image is filtered by the average template having the same size as the object rectangle, as shown by coloring in Fig.1(c).

2.3 Haar-like

Now that we select motion and color as features for hand tracking, the background is accordingly classified into four kinds. The first kind is not moving and skin-colored. The second kind is moving, but not skin-colored. The third kind is not moving, but skin-colored. The fourth kind is moving and skin-colored. Motion combined with color could distinguish the hand from the first three kinds of background. The remaining question is searching for another feature to defeat moving skin-colored distractors. We construct three online boosting trackers[16] respectively using Haar-like features, orientation histograms, and local binary patterns. Although none of them perform well on the test sequences, Haar-like features succeed to get a low similarity between the tracked hand and skin-colored backgrounds. A strength image based on Haar-like features produced in the tracker is shown by coloring in Fig.1(d). We can see that values corresponding to skin-colored backgrounds (e.g. faces and other hands) are smaller than the value corresponding to the tracked hand.

2.4 Cascaded tracker

Motion, color, and Haar-like features are different types of feature. We suppose that the strength images based on each kind of feature are independent. A fused strength image is computed using the product of them, as shown in color in Fig.1(e). In this image, it's easy to locate the tracked hand only depending on the maximum value, which demonstrates the effectiveness of feature fusion. Note that all the strength images just need to be computed within the search range in practice. These features can be integrated roughly in parallel to build a hand tracker. Three individual strength images are computed for each new frame and then multiplied together to obtain a fused strength image. The maximum rule is used to locate the tracked hand in this image. The tracker is expected to perform better than those relying on single kind of feature, but there are two obvious disadvantages. Firstly, it only needs one or two kinds of features to track the hand in most cases. The redundant feature processing increases the execution time. Secondly, the tracker does not benefit from a newly added feature all the time. The newly added feature needs to introduce no negative influence.

In this paper, we use each kind of feature to build three weak trackers and then integrate them in a boosted cascade. The block diagram of this approach is shown in Fig.2. The color tracker computes a color mask based on the method as described in Section 2.2 and extracts the connected regions. The region which is smaller than half of the object rectangle is removed after holes are filled. If there are no regions left, the object is considered to be occluded by something else or disappear from the camera view. If only one region which is smaller than the object rectangle remains, it is considered to be the tracked hand. We calculate the average coordinate in this region as the center of the object rectangle. Otherwise, it means the color tracker is unable to locate the object. The motion tracker first detects whether the object is moving or not based on the method as described in Section 2.1. If the object is moving, the motion mask and the color mask from the color tracker are multiplied together to get a fused mask. It extracts the connected regions using this fused mask and attempts to locate the object as the same as the color tracker. The color and motion trackers are enough to distinguish the tracked hand from most of backgrounds except for moving skin-colored distractors. In order to overcome this limitation, we train a Haar-like classifier during tracking using the on-line Adaboost algorithm. When the first two trackers fail to locate the object, the Haar-like classifier detects each pixel whose value is nonzero in the fused mask. If all the strength values produced by this classifier are negative, the object is occluded by something else or disappears from the camera view. Otherwise, the pixel with the maximum strength value is considered to be the center of the object rectangle.

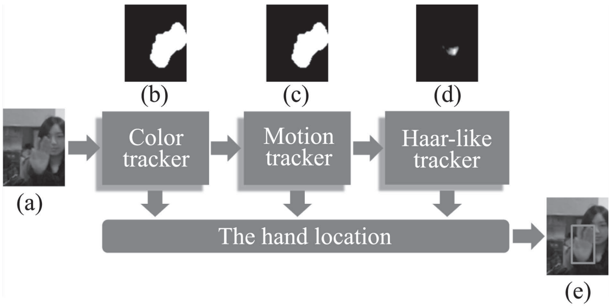

Fig. 2 The block diagram of the proposed approach. (a) The image patch within the search window. (b) The color mask produced in the color tracker. (c) The fused mask produced in the motion tracker. (d) The fused mapping image of hand-likelihood produced in the Haar-like tracker. (e) The object rectangle obtained by proposed approach is drawn on the image patch.

2.5 Related issues

There are several related issues that need to be solved when we construct a real hand tracking system.

Tracking initialization. Although tracking initialization is beyond the scope of this paper, it is important for a real tracking system. In all of our experiments, a bounding box around the object is drawn by hand to initialize the tracker when the object appears in each sequence. In practice, object tracking could be initialized by automatically detecting the moving object. We train an open hand detector offline as in [22] and introduce constraints to improve the accuracy, such as the face position, motion, and skin color.

Update. The color tracker is updated only when the first two trackers succeed to locate the object. The object GMM is established based on the pixels whose values are nonzero in the mask. The background GMM is established based on the pixels in the remaining region within the search window. Positive and negative samples are produced based on the object position to update the Haar-like classifier as in [16].

Handling occlusion. The tracked hand could be occluded when someone walks in front of the user. If the object disappears in the image within a short time, the tracker does not update the object rectangle until the object reappears. Otherwise, the tracker is finished and waits to be reinitialized. Note that the tracker cannot distinguish whether the object is occluded or moves out of the camera view. But the two cases can be handled by the same solution.

3 Experiments

The proposed approach was implemented using Visual C++ and validated on a variety of challenging sequences. As no standard dataset is available in literature to evaluate hand tracking, we created several challenging real-world video sequences, due to space limitation, experimental results on three representative sequences are presented in this section. The three sequences were captured by a motionless normal camera viewing possibly multiple people in our laboratory. A small number of frames in each sequence are shown in Fig.3.

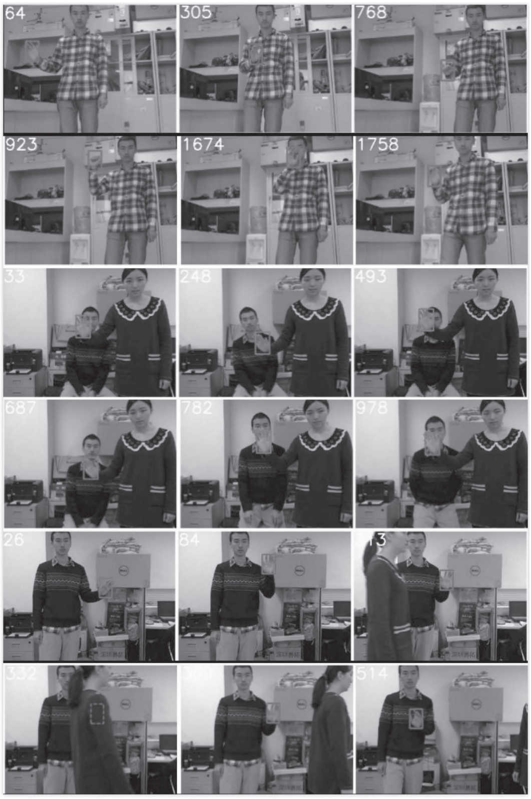

Fig. 3 Three sequences captured by a motionless normal camera viewing possibly multiple people in our laboratory. The object rectangle in each frame obtained by proposed approach is drawn on the image.

No parameters were changed from one experiment to the next. An almost saturated performance is achieved by the proposed approach on all these sequences. The tracked object rectangle in each frame is drawn on the image, as shown in Fig.3. In addition, we selected three other challenging sequences in [13] to evaluate the generality of our approach. In this paper, the proposed tracker is used for single hand tracking, therefore the other five sequences in [13] contain much movement with two hands were not evaluated here. The tracked object are marked by rectangle the same as before, as shown in Fig.4.

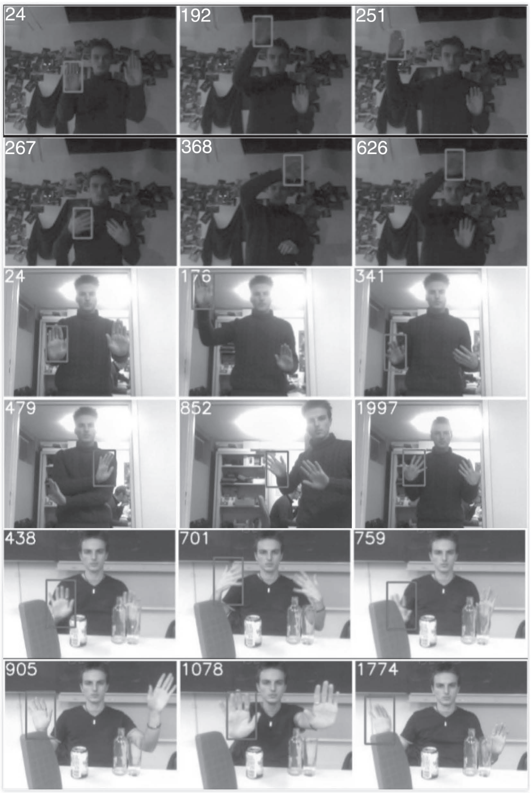

Fig. 4 Three representative challenging sequences used in [13], which captured at different conditions including illumination variation, scale variation, deformation, motion blur, fast motion and background clutters. The object rectangle in each frame obtained by proposed approach is drawn on the image.

For evaluation purposes, we calculated the Pascal VOC score

$$S(x,y)=\frac{\mathrm{Area}\left(\mathrm{box}_{\mathrm{tracker}}\cap\mathrm{box}_{\mathrm{groundtruth}}\right)}{\mathrm{Area}\left(\mathrm{box}_{\mathrm{tracker}}\cup\mathrm{box}_{\mathrm{groundtruth}}\right)}$$

of each sequence. A score lower than 0.5 could be considered a tracker error, while a score of 0 means that the tracker completely lost the target. Table 1 shows the average score for each of the algorithms evaluated on each of the sequences. The first column represents different sequences, the first three are shown in Fig.3 and the last are shown in Fig.4. The four tracking approaches' source code are available on the Internet and the default parameters were used. Although we have tried to adjust the parameters of these algorithms to achieve a better performance, there was no significant improvement in our experiments. From the table, we can see that our approach achieved the best performance on all these sequences, obviously outperforming other trackers.

Table 1 Average VOC score for each sequence

Seq.	OAB[16]	Struck[18]	TLD[19]	KCF[21]	Ours
1	0.0970	0.4214	0.1651	**0.5366**	**0.7454**
2	0.0641	0.3085	0.199	**0.2409**	**0.5972**
3	0.0381	**0.2137**	0.1021	0.196	**0.6954**
4	0.0192	0.0823	0.0865	**0.1465**	**0.4982**
5	0.0382	0.0737	0.2409	**0.344**	**0.5816**
6	0.0593	0.1321	**0.2885**	0.2691	**0.5490**

Experimental results demonstrate that our approach is able to track a hand in unconstrained videos captured by a motionless normal camera viewing possibly multiple people. Feature type is crucial in OAB, however, it's difficult to select the optimal feature type during tracking and thus it may lead to model drift and tracking failure. The TLD and Struck train a classifier online, but they may get trapped in a background which exhibit a similar appearance compared to hand. Since the KCF uses a fixed template size in kernel correlation filter, it may result in tracking failure when the scale of the objects changes. In our proposed tracker, we use color and motion feature which are scale-adaptive and robust to deformations and partial occlusions. What's more, with on-line AdaBoost algorithm, the tracker can detect the hand from background clutter.

4 Conclusions

In this paper, we proposed a new approach that combines motion, color, and Haar-like features to construct a real-time and robust hand tracking system (about 22 fps using a PC with Intel Core2 CPU). There are three key benefits of our approach. The basis is that we select three kinds of features according to different types of difference between the hand and background. Although multiple features are also used in [16, 19], they are not carefully designed for hand tracking. Furthermore we build three week trackers for each kind of feature and integrate them into a boosted cascade. Once the object is located, no other trackers will be executed. However, if the tracker fails to locate the object, it could also reduce the search window. Hence it reduces the computation while tracking hand, and the system can also performance well at a high tracking speed. This cascaded structure is crucial especially when no one kind of feature is enough for tracking the object, and the weak tracker can benefit from the integration. Finally, unlike the algorithm in [13], the classifier has to be trained off-line, we adopt on-line AdaBoost algorithm to update the Haar-like features of hand during tracking and thus is able to cope with appearance changes

of the hand and background. In addition, experimental results compared with several state-of-the-art methods on challenging sequences demonstrate the effectiveness and robustness of the proposed algorithm.

References:

[1] FREEMAN W T, WEISSMAN C D. Television control by hand gestures [J] //*International Workshop on Automatic Face and Gesture Recognition*, 1995: 179 – 183.

[2] ARGYROS A A, LOURAKIS M I A. Real-time tracking of multiple skin-colored objects with a possibly moving camera [C] //*European Conference on Computer Vision*. Berlin, Heidelberg: Springer, 2004: 368 – 379.

[3] WANG H, SANG N, YAN Y. Real-time tracking combined with object segmentation [C] //*International Conference on Pattern Recognition*. Stockholm, Sweden: IEEE, 2014: 4098 – 4103.

[4] KRAHNSTOEVER N, SCHAPIRA E, KETTEBEKOV S, et al. Multimodal human-computer interaction for crisis management systems [C] //*IEEE Workshop on Applications of Computer Vision*. Flordia Orlando: IEEE, 2002: 203 – 207.

[5] STENGER B, WOODLEY T, KIM T, et al. AIDIA—Adaptive interface for display interaction [C] //*British Machine Vision Conference*. Leeds: [s.n.], 2008: 1 – 10.

[6] OIKONOMIDIS I, KYRIAZIS N, ARGYROS A A. Efficient model based 3D tracking of hand articulations using Kinect [C] //*Proceedings of British Machine Vision Conference*. Scotland: [s.n.], 2011, 1(2): 3.

[7] SRIDHAR S, OULASVIRTA A, THEOBALT C. Interactive markerless articulated hand motion tracking using RGB and depth data [C] //*IEEE International Conference on Computer Vision*. Sydney: IEEE, 2013: 2456 – 2463.

[8] QIAN C, SUN X, WEI Y, et al. Realtime and robust hand tracking from depth [C] //*IEEE Conference on Computer Vision and Pattern Recognition*. Columbus: IEEE, 2014: 1106 – 1113.

[9] FARNEBÄCK G. Two-frame motion estimation based on polynomial expansion [C] //*Scandinavian Conference on Image Analysis*. Berlin, Heidelberg: Springer, 2003: 363 – 370.

[10] BARRON J L, FLEET D J, BEAUCHEMIN S S. System and experiment performance of optical flow techniques [J]. *International Journal of Computer Vision*, 1994, 32(2): 72 – 4.

[11] CANNY J. A computational approach to edge detection [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1986, 8(6): 679 – 698.

[12] DALAL N, TRIGGS B. Histograms of oriented gradients for human detection [C] //*Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. San Diego: IEEE, 2005, 1: 886 – 893.

[13] SPRUYT V, LEDDA A, PHILIPS W. Real-time hand tracking by invariant hough forest detection [C] //*The 19th IEEE International Conference on Image Processing (ICIP)*. Flordia Orlando: IEEE, 2012: 149 – 152.

[14] COLLINS R T, LIU Y, LEORDEANU M. Online selection of discriminative tracking features [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2005, 27(10): 1631 – 1643.

[15] AVIDAN S. Ensemble tracking [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2007, 29(2): 261 – 271.

[16] GRABNER H, GRABNER M, BISCHOF H. Real-time tracking via on-line boosting [C] //*British Machine Vision Conference*. Edinburgh, Uk: [s.n.], 2006, 1(5): 6.

[17] KWON J, LEE K M. Visual tracking decomposition [C] //*IEEE Conference on Computer Vision and Pattern Recognition*. San Francisco: IEEE, 2010: 1269 – 1276.

[18] HARE S, GOLODETZ S, SAFFARI A, et al. Struck: Structured output tracking with kernels [J]. *IEEE transactions on pattern analysis and machine intelligence*, 2016, 38(10): 2096 – 2109.

[19] KALAL Z, MIKOLAJCZYK K, MATAS J. Tracking-learning-detection [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2012, 34(7): 1409 – 1422.

[20] WU Y, LIM J, YANG M H. Online object tracking: a benchmark [C] //*2013, IEEE Conference on Computer Vision and Pattern Recognition*. [s.l.]: IEEE, 2013: 2411 – 2418

[21] HENRIQUES J F, CASEIRO R, MARTINS P, et al. High-speed tracking with kernelized correlation filters [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2015, 37(3): 583 – 596.

[22] VIOLA P, JONES M. Rapid object detection using a boosted cascade of simple features [C] //*IEEE Conference on Computer Vision and Pattern Recognition*. Kauai, Hawaii: IEEE. 2003, 1: 511.

[23] OJALA T, PIETIKÄINEN M, MÄENPÄÄ T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2002, 24(7): 971 – 987.

[24] STENGER B. Template-based hand pose recognition using multiple cues [M] //*Computer Vision—ACCV 2006*. Berlin, Heidelberg: Springer, 2006: 551 – 560.

[25] ZHOU H, HUANG T S. Tracking articulated hand motion with eigen dynamics analysis [C] //*Proceedings of the 9th IEEE International Conference on Computer Vision*. Nice, France: IEEE, 2003: 1102 – 1109.

[26] STENGER B, THAYANANTHAN A, TORR P H S, et al. Model-based hand tracking using a hierarchical bayesian filter [J]. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006, 28(9): 1372 – 1384.

[27] STENGER B, THAYANANTHAN A, TORR P H S, et al. Hand pose estimation using hierarchical detection [M] //*Computer Vision in Human-Computer Interaction*. Berlin, Heidelberg: Springer, 2004: 105 – 116.

[28] CHAN A B, MAHADEVAN V, VASCONCELOS N. Generalized Stauffer – Grimson background subtraction for dynamic scenes [J]. *Machine Vision and Applications*, 2011, 22(5): 751 – 766.

[29] PHAM D S, ARANDJELOVIC O, VENKATESH S. Detection of dynamic background due to swaying movements from motion features [J]. *IEEE Transactions on Image Processing*, 2015, 24(1): 332 – 344.

作者简介:

李 逢 (1992–), 男, 硕士, 目前研究方向为目标检测与跟踪, E-mail: hustwinds@hust.edu.cn;

桑 农 (1968–), 男, 博士, 教授, 目前研究方向为生物视觉感知模型(目标识别、运动感知、注意等)及其在计算机视觉中的应用、视觉认知、基于统计学习的图像分析与目标识别、医学图像处理与分析、遥感影像解译、智能视频监控, E-mail: nsang@hust.edu.cn;

王洪智 (1990–), 男, 硕士, 目前研究方向为目标检测与跟踪, E-mail: hzwang@hust.edu.cn;

颜 轶 (1990–), 女, 硕士, 目前研究方向为目标检测与跟踪, E-mail: yanyi@hust.edu.cn;

高常鑫 (1983–), 男, 博士, 副教授, 计算机视觉与模式识别、生物视觉感知模型、图像处理与分析等, E-mail: cgao@hust.edu.cn;

刘乐元 (1982–), 男, 博士, 讲师, 目前研究方向为计算机视觉、多模态人机交互, E-mail: lyliu@mail.ccnu.edu.cn.