Collaborative Tracking: Dynamically Fusing Short-Term Trackers and Long-Term Detector

Guibo Zhu, Jinqiao Wang, Changsheng Li, and Hanqing Lu

National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing 100190, China
{gbzhu,jqwang,csli,luhq}@nlpr.ia.ac.cn

Abstract. This paper addresses the problem of long-term tracking of unknown objects in a video stream given its location in the first frame and without any other information. It's very challenging because of the existence of several factors such as frame cuts, sudden appearance changes and long-lasting occlusions etc. We propose a novel collaborative tracking framework fusing short-term trackers and long-term object detector. The short-term trackers consist of a frame-to-frame tracker and a weakly supervised tracker which would be updated under the weakly supervised information and re-initialized by long-term detector while the trackers fail. Additionally, the short-term trackers would provide multiple instance samples on the object trajectory for training a long-term detector with the bag samples with P-N constraints. Comprehensive experiments and comparisons demonstrate that our approaches achieve better performance than the state-of-the-art methods.

Keywords: collaborative tracking, online learning, samples selection.

1 Introduction

Long-term tracking in unconstrained environments is a very active topic in computer vision due to its wide-ranging applications in video indexing, surveillance, human-computer interaction, augmented reality, etc. [1, 2]. A tracking system usually consists of three components: 1) an appearance model, used for evaluating the likelihood that the object of interest is at some particular location; 2) a motion model, which relates the locations of the object over time; 3) a search strategy for finding the most possible location in the current frame [3]. However, the problem and difficulty in a tracking system depend on several sources of varieties such as changes in appearance, varying lighting conditions, cluttered background, partial or complete occlusion, and frame-cuts.

Nowadays, various tracking algorithms have been proposed [13, 14, 17, 4, 6]. Template tracking [13, 14, 15] is the most straightforward approach that estimates the objects' motion between consecutive frames. Templates have limited modeling capability as they represent only a single appearance of the object. To deal with more appearance variations, the generative models [17, 18, 19, 20, 21] have been proposed. However, the generative trackers only model the appearance of the object and as such often fail in cluttered background. In order to alleviate this problem, training an

S. Li et al. (Eds.): MMM 2013, Part II, LNCS 7733, pp. 457–467, 2013.

adaptive discriminative classifier in an online manner to distinguishing the object from the background has shown promising results [3, 4, 5, 6]. The essential phase of adaptive discriminative trackers is the update: the close neighborhood of the current location is used to sample positive training examples, distant surrounding of the current location is used to sample negative examples, and these are used to update the classifier in each frame. It has been demonstrated that this updating strategy handles significant appearance changes, short-term occlusions, and cluttered background. However, these methods suffer from drift and failure if the object leaves the scene for a long time. To address the problems, the update of the tracking classifier has been constrained by an auxiliary classifier trained in the first frame [7] or by training a pair of independent classifiers [8, 9].

In this paper, we focus on the problem of long-term tracking an arbitrary object with no prior knowledge other than its location in the first frame. To develop a robust updating adaptive appearance models, we would like to handle partial occlusions or disappearance without significant drift through exploring the interrelationship between the short-term tracker and the long-term detector. Here, the adaptive short-term trackers consist of a frame-to-frame tracker and a weakly supervised tracker which would be updated under the weakly supervised information and re-initialized by long-term detector while the trackers fail. Simultaneously, the adaptive short-term trackers would provide multiple instance samples on the object trajectory for training a long-term detector. Unlike previous methods, we exploit the steady local information of object and develop the adaptive short-term trackers. Our algorithm dynamically fuses adaptive trackers and detector, which can deal with the appearance model and the motion model in a novel framework. Experimental results on the public available datasets demonstrate the effectiveness of our method.

The rest of the paper is organized as follows. In the next section, we introduce our tracking algorithm; in Section 2, we present qualitative and quantitative results of our tracker on a number of challenging image sequences. We draw the conclusion in Section 4.

2 The Proposed Approach

We present details of the robust visual tracking framework by fusing adaptive short-term trackers and long-term detector, as shown in Fig.1.

The components of the framework are characterized as follows: the frame-to-frame tracker estimates the object's motion between consecutive frames. Adaptive short-term tracker estimates the object's location under the assumption that the object is visible or partial visible. If the object moves quickly or is occluded partially abruptly, the adaptive short-term tracker may recover when the frame-to-frame tracker is likely to fail and never recover by itself. The adaptive short-term trackers could provide multiple instance samples on the object trajectory for training a long-term detector.

The trained detector will scan full of the frame to localize all possible candidate patch that is similar to all appearances observed. Learner evaluates the performance of trackers and detector, estimates detector's errors and generates the credible templates

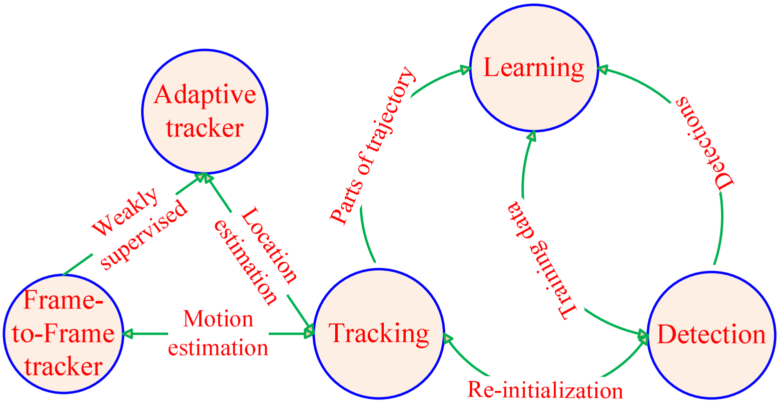

Fig. 1. The block diagram of our approach

and training data. The training data consists of bag samples to reinforce the detector's capability. For alleviating the effect from the condition that both the frame-to-frame tracker and the detector fail, we introduce adaptive short-term tracker and P-N constraints for bag samples selection to improve the detector's generalization capability. Additionally, because of the existence of object templates learned from the past, the learning strategy makes the detector have strong ability to discriminate the object against background.

2.1 Short-Term Trackers

Adaptive short-term trackers contain a frame-to-frame tracker and an approximate multiple instance learning tracker (MIL) [3]. Frame-to-frame tracker is used for exploring the motion of consecutive frames. We adopt the approach of Kalal et al. [21] for recursive tracking which bases on Lucas-Kanade tracker (KLT) [13].

The approach of KLT bases on three assumptions. The first assumption is referred to as brightness constancy [23] and is

$$I(X) = J(X + d) \tag{1}$$

Eq. (1) states that a pixel at the two-dimensional location X in an image I might change its location in the second image J but retains its brightness value. The vector d will be referred to as the displacement vector. The second assumption is referred to [22] as temporal persistence. It states that the displacement vector is small. Small in this case means that $J(X)$ can be approximated by

$$J(X) \approx I(X) + I'(X)d \tag{2}$$

where $I'(X)$ is the gradient of I at location X .

The third assumption, known as spatial coherence, alleviates this problem. It states that all the pixels within a window around a pixel move coherently. By incorporating this assumption, d is found by minimizing the term

$$\sum_{(x,y)\in W} (J(X)-I(X)-I'(X)d)^2 \tag{3}$$

which is the least-squares minimization of the stacked equations. The size of W defines the considered area around each pixel. Additional implementation details are in [22].

According to the forward-backward error measure [21], Lucas-Kanade method is applied twice on points P_{b1} in the bounding box of the object and measured based on the similarity of the patches P_1 surrounding points P_{b1} and the patches P_2 surrounding the tracked points P_{b2} . Since the normalized correlation coefficient is invariant against uniform brightness variations [23], the similarity of these two patches P_1 and P_2 is calculated by the Normalized Correlation Coefficient (NCC) as

$$NCC(P_1,P_2)=\frac{1}{n-1}\sum_{x=1}^{n}\frac{(P_1(x)-\mu_1)(P_2(x)-\mu_2)}{\delta_1\delta_2} \tag{4}$$

where μ_1 , μ_2 , σ_1 and σ_2 are the means and standard deviations of P_1 and P_2 .

Under the three assumptions of Lucas-Kanade, this frame-to-frame tracker could provide samples for the long-term detector. If any of the three assumptions are not met, the frame-to-frame tracker would have a failure so that it couldn't provide the enough training samples for long-term detector, which has enormous influence on the tracked results. If the object is occluded quickly, the assumptions will be violated. For solving this problem, we introduce a weakly supervised tracker which could mine the discriminative local patch information and estimate the object effectively in short term, especially when the long-term detector isn't trained sufficiently.

In this paper, we use weakly supervised multiple instance learning tracking (WSMILT) as our weakly supervised tracker. Unlike MIL Track [3], WSMILT will use the weakly supervised information from frame-to-frame tracker. The basic flow of adaptive short-term tracker in this work is illustrated in Fig.1 and summarized in Algorithm 1. Like MIL Track [3], we extract a set of Haar-like features for each image patch [11, 24]. Then the appearance model is composed of a discriminative classifier which is able to return $p(y=1\mid x)$,where x is an image patch and y is a binary variable indicating the presence of the object of interest in that image patch. At every time step t , our weakly supervised tracker maintains the object location l_t^* . Let $l(x)$ denote the center location of image patch x. For each new frame, if the frame-to-frame tracker has tracked the object, we crop out a set of image patches $X_s=\{x:\| l(x)-l_{t-1}^* \|< s\}$ that are within some search radius s of the current tracker location, and compute $p(y=1\mid x)$ for all $x\in X^s$ .We then use a greedy strategy to update the tracker location:

$$l_t^* = l\left(\underset{x \in X^s}{\arg\max}\; p(y=1\,|\,x)\right)) \tag{5}$$

In other words, we don't maintain a distribution of the target's location at each frame, and our motion model assumes that the location of the tracker at time t is equally likely to appear within a radius s of the tracker location at time (t-1):

$$p(l_t^* \,|\, l_{t-1}^*) \propto \begin{cases} 1 & \mathit{if} \; \| l_t^* - l_{t-1}^* \| < s \\ 0 & \mathit{otherwise.} \end{cases} \tag{6}$$

```
Algorithm 1. Weakly Supervised Multiple Instance Learning Tracking
Input: Video frame number $k$
Method:
    1: Crop out a set of image patches, $X_s = \{x : \| l(x) - l_{t-1}^* \| < s\}$ and compute feature vectors.
    2: Use multiple instance learning classifiers to estimate the probability $p(y=1\,|\,x)$ for $x \in X^s$.
    3: Update the tracker location $l_t^* = l\left(\underset{x \in X^s}{\arg\max}\; p(y=1\,|\,x)\right))$.
    4: Crop out two sets of image patches $X^r = \{x : \| l(x) - l_t^* \| < r\}$ and $X^{r,\beta} = \{x : r < \| l(x) - l_t^* \| < \beta\}$, where $r < s < \beta$.
    5: If the frame-to-frame tracker has tracked the object, we update MIL appearance model with one positive bag $X^r$ and $| X^{r,\beta} |$ negative bags, each containing a single image patch from the set $X^{r,\beta}$.
Output: Object bounding box $A_t$
```

2.2 Long-Term Detector

Object detection enables us to re-initialize the frame-to-frame tracker since it doesn't maintain an object model and unable to recover from failure. While the frame-to- frame tracker depends on the location of the object in the previous frame, the object detection mechanism presented here employs an exhaustive search in order to find the lost object.

Due to the efficiency of randomized ferns classifier [27] which is widely used in object recognition [2, 25, 26], we employ it as long-term detector to find possible object location. Ferns classifier consists of a number of ferns which are evaluated in parallel on each patch and fast. Each leaf in a fern records the number of positive p and negative n examples using Binary Pattern features during training. For a test sample, its evaluation by calculating the binary pattern features leads to a leaf in the fern. After that, the posterior probability for that input testing sample in feature vector x_i to be labeled as an object $(y=1)$ by a fern j is computed as maximum likelihood estimator $\Pr_j(y=1 \mid x_i)=p/(p+n)$, or is set zero if the leaf is empty. The final probability is calculated by averaging the posterior probabilities given by all ferns:

$$\Pr(y=1 \mid x_i)=\sum_{j=1}^{T}\Pr_j(y=1 \mid x_i) \tag{7}$$

where T is the number of ferns. Short-term trackers controls the posterior by adding its positive and negative samples to the ferns according to P-N constraints as [2] and multiple instance bag [3]. The P-constraints force all samples close to the validated trajectory to have positive label, while N-constraints have all patches far from the validated trajectory labeled as negative. Differently from [2], we bring in multiple instance bags around the validated trajectory so as to avoid the following problems. Slight inaccuracies in the tracker can therefore lead to incorrectly labeled training examples, which will further lead to the classifier resolving the ambiguities by itself to yield robust tracking results.

2.3 Samples Selection

A good classifier needs to have high prediction accuracy and generalization capability. The training samples' quality is crucial, especially for the training of online classifiers. In this paper, we introduce the bag samples selection to enhance the robustness of P-N constraints, which is able to use both weakly labeled and unlabeled bags.

The P-N constraints explore the latent information that there are some spatial structure and temporal structure information among different patches in video sequences. The constraints assume that a single object appears in one location only and therefore its trajectory defines a curve in the spatial-temporal volume. The trajectory curve is not continuous and generated by adaptive Lucas-Kanade [13] tracker and evaluated by the patch selected in the first frame using NCC measure to evaluate the confidence. P-constraints require that all patches that are close to validated trajectory have positive label. N-constraints require all patches in surrounding of a validated trajectory have negative label. In this paper, to mine and use the latent information effectively, especially to improve the generation capability of long-term detector, we sample the positive bag based on the patches close to validated trajectory and training online detector with the instance of the positive bag with soft-label. For detail, in our weakly supervised multiple instance learning tracker, training data has the form $\{(X_1,y_1),...(X_n,y_n)\}$, where a bag $X_i=\{x_{i1},...,x_{im}\}$ and y_i is a bag label. The bag labels are defined as:

$$y_i=\max_j(y_{ij}) \tag{8}$$

where y_{ij} are the instance labels, which are not known during the training. Since we assume the patches in or very close to validated trajectory as positive instance, the bag which contains the patches is positive bag. The bag samples could be used for training the long-term detector.

2.4 Collaborative Training and Online Update

Frame-to-Frame tracker is used for motion estimation and collects the new templates which have high confidence with the old templates in the past validated trajectory of object appearance resized patches. It will be re-initialized by the final result fusing the trackers' and detector's result in the previous frame.

Adaptive weakly supervised tracker will be trained under the weakly supervised information coming from Frame-to-Frame tracker so that it could adapt to more cluttered background and prevent from drifting. Additionally, it could recommend more likely training samples for detector learning selection, especially in the case that detector hasn't been trained enough so as to fail to detect the possible candidates.

Learner will select the appropriate training data to train the long-term detector. For improving the detector's generalization capability, we generate multiple instance bags based on the predicted object location which is in the validated trajectory. For simplicity, we relax the condition of positive training examples and think that the instances' label is same to the bag's label:

$$y_{ij} = y_i \tag{9}$$

where y_{ij} is the label of the jth instance in the ith bag and y_i is the ith bag's label. Additionally, the instances in one same bag should be satisfied that:

$$X_i = \{x : \| l(x) - l_t^*(x) \| < s\} \tag{10}$$

where X_i is the ith bag, $l_t^*(x)$ is the predicted object location which is in the validated trajectory, $l(x)$ is the image patch's location, s is the bag's radius.

2.5 Result Fusion of Trackers and Detector

To fusing the results of the frame-to-frame tracker F_t, the weakly supervised tracker A_t and the confident detections D_t into a final result B_t is given. The decision is based on the number of detections, the detector' confidence values $P_{D_t}^+$ and the confidence of the tracking results $P_{R_t}^+$, $P_{A_t}^+$. The latter is obtained by running the template matching method on the tracking results. If the detector yields exactly one result with a confidence higher than the result from the trackers, then the response of the detector is assigned to the final result. The frame-to-frame tracker will be re-initialized by the final result. If the frame-to-frame tracker produced the most confident result, the result will be assigned to the final result. If their confidents are all high, we combine them by median selection. If $P_{R_t}^+$ and $P_{D_t}^+$ is low, we choose to believe the adaptive tracker. If $P_{A_t}^+$ is bigger than a threshold, the A_t is assigned to the final result. In other cases the final result remains empty, which suggests that the object is not found in the current frame.

3 Experimental Results

In order to evaluate the performance of the proposed tracking approach, we test our system in C++ on several challenging image sequences. Nine videos (David, Jumping, Animal, Shaking, Cliffbar, Faceocc, Faceocc2, Surfer, Sylv) [10, 20, 3] are collected from the public dataset. The challenges of these videos include illumination variation, partial occlusion, pose variation, background clutter and scale change. For cross-validation, the center position error is compared with that of current state-of-the-art methods (FT[12], L1[29], MIL[3], and TLD[10]). We implemented these trackers using publicly available source code or binaries provided by the authors. They were initialized using their default parameters.

Table 1. Average center location error (pixels). The best performance is in bold, the second best is in underlined.

Sequence	#Frames	FT[12]	L1[29]	MIL[3]	TLD[10]	OURS
David	761	90	51.9	39.9	14.9	**5**
Jumping	313	58.2	50.8	12.6	5.6	**4.7**
Animal	71	91.2	160.5	27.9	86.6	**12.1**
Shaking	365	61.7	117.7	51.4	231.8	**23.3**
Cliffbar	328	17.7	43.3	**13.8**	50.7	16.5
Faceocc	886	**5.7**	6.6	35.3	11.3	13.7
Faceocc2	812	15.5	30.4	12.2	14.8	**6.8**
Surfer	376	139	37.7	16.1	18.1	**15.9**
Sylv	1344	13.3	34.5	14.7	**9.4**	13.3

The performance of visual trackers is evaluated according to the average per-frame distance (in pixels) between the center of the tracking result and that of ground truth. Clearly, this instance should be small. In Fig.2, we can see that our tracker consistently produce s a smaller distance than other trackers. This implies that our method can accurately track the target despite illumination changes.

At the same time, performance evaluation on public datasets is measured by Precision/Recall [2]. The results are displayed in Table 2.

The *David* sequence has large illumination changes. The initialized box makes many generative models fail in several frames. TLD and our method add the motion estimation information so as to prevent the target from missing. It's also very important for one appearance model updating. In the *Shaking* sequence, the tracked object is subject to changes in illumination and pose. TLD will fail in frame *58* because of abrupt powerful light. Our method will work because of the short-term tracker. When abrupt motion and large appearance changes simultaneously, our algorithm may fail.

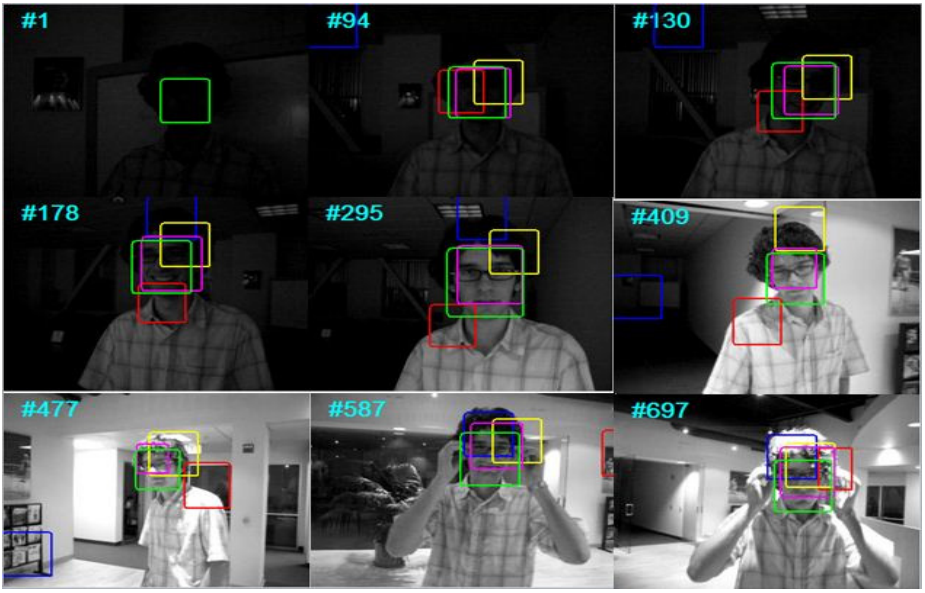

Fig. 2. Representative frames on sequences David under illumination changes. Blue, red, yellow, magenta and green bounding boxes were generated by FT, L1, MIL, TLD, and ours, respectively.

Table 2. Performance evaluation on public dataset measured by Precision/Recall. Bold numbers indicate the best score. The dataset is same as Table 1.

Sequence	FT[12]	L1[29]	MIL[3]	TLD[10]	OURS
David	0.158/0.158	0.309/0.309	0.143/0.143	0.999/0.999	**1.000/1.000**
Jumping	0.204/0.204	0.179/0.179	0.978/0.978	1.000/0.997	**1.000/1.000**
Animal	0.042/0.042	0.056/0.056	0.887/0.887	0.981/0.746	**1.000/1.000**
Shaking	0.397/0.397	0.063/0.063	0.825/0.825	**1.000**/0.156	0.893/**0.893**
Cliffbar	0.393/0.393	0.305/0.305	0.909/0.909	**0.942**/0.591	0.893/**0.893**
Faceocc	**1.000/1.000**	1.000/1.000	0.997/0.997	1.000/1.000	1.000/1.000
Faceocc2	**1.000/1.000**	0.702/1.000	**1.000/1.000**	**1.000/1.000**	0.974/0.974
Surfer	0.221/0.221	0.093/0.093	0.646/0.646	0.774/0.774	**0.787/0.787**
Sylv	0.885/0.885	0.467/0.467	0.858/0.858	0.949/0.949	**0.955/0.955**

The whole quantitative comparisons are shown in Table 1 and Table 2. From the tables, we can see that our tracking algorithm is better than the others in most cases.

4 Conclusion

In this paper, we propose a novel framework exploring their mutual relationship of adaptive trackers and detector and fusing them to act on visual tracking. Our method combines the flexibility of multiple instance learning on where to select positive updates, the effectiveness of frame-to-frame tracking on object motion estimation and the robustness of detector towards partial occlusion and disappearance. In order to alleviate the drift of adaptive multiple instance tracker, we use the weakly supervised information coming from the frame-to-frame tracker. For improving the detector's generation capability, P-N constraints for bag samples selection are introduced to train the detector. Experimental results show the superiority of our approach over state-of-the art methods.

Acknowledgments. This work was supported by 973 Program (2010CB327905) and National Natural Science Foundation of China (61070104，61273034，61202325, 60905008).

References

1. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A Survey. ACM Computing Surveys 38(4) (2006)
2. Kalal, Z., Matas, J., Mikolajczyk, K.: P-N Learning: Bootstrapping Binary Classifiers by Structural Constraints. In: Conference on Computer Vision and Pattern Recoginition (2010)
3. Babenko, B., Yang, M.H., Belongie, S.: Visual Tracking with Online Multiple Instance Learning. In: Proc. CVPR (2009)
4. Grabner, H., Bischof, H.: On-line boosting and vision. In: CVPR (2006)
5. Collins, R., Liu, Y., Leordeanu, M.: Online Selection of Discriminative Tracking Features. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(10), 1631–1643 (2005)
6. Avidan, S.: Ensemble Tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(2), 261–271 (2007)
7. Grabner, H., Leistner, C., Bischof, H.: Semi-supervised On-Line Boosting for Robust Tracking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part I. LNCS, vol. 5302, pp. 234–247. Springer, Heidelberg (2008)
8. Tang, F., Brennan, S., Zhao, Q., Tao, H., Santa Cruz, U.C.: Co-tracking using semi-supervised support vector machines. In: ICCV (2007)
9. Yu, Q., Dinh, T.B., Medioni, G.G.: Online Tracking and Reacquisition Using Co-trained Generative and Discriminative Trackers. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 678–691. Springer, Heidelberg (2008)
10. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-Learning-Detection. IEEE Transaction on Pattern Analysis and Machine Intelligence 6(1) (2010)
11. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: CVPR (2001)
12. Adam, A., Rivlin, E., Shimshoni, I.: Robust fragments-based tracking using the integral histogram. In: CVPR (2006)
13. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: International Joint Conference on Artificial Intelligence, vol. 81 (1981)
14. Shi, J., Tomasi, C.: Good features to track. In: CVPR (1994)
15. Matthew, I., Ishikawa, T., Baker, S.: The Template Update Problem. IEEE TPAMI (2004)
16. Black, M.J., Jepson, A.D.: Eigentracking: Robust matching and tracking of articulated objects using a view-based representation. IJCV (1998)
17. Ross, D., Lim, J., Lin, R., Yang, M.: Incremental Learning for Robust Visual Tracking. IJCV (2007)

18. Wang, S., Lu, H., Yang, F., Yang, M.-H.: Superpixel Tracking. In: CVPR (2009)
19. Kwon, J., Lee, K.M.: Visual Tracking Decomposition. In: CVPR (2010)
20. Liu, B., Huang, J., Yang, L., Kulikowsk, C.: Robust Tracking Using Local Sparse Appearance Model and K-Selection. In: CVPR (2011)
21. Kalal, Z., Mikolajczyk, K., Matas, J.: Forward-Backward Error: Automatic Detection of Tracking Failures. In: ICCV (2010)
22. Bradski, G., Kaehler, A.: Learning OpenCV: Computer Vision with the OpenCV Library, 1st edn. O'Reilly Media (2008)
23. Lewis, J.P.: Fast normalized cross-correlation. In: Vision Interface. In: Canadian Image Processing and Pattern Recognition Society (1995)
24. Dollar, P., Tu, Z., Tao, H., Belongie, S.: Feature Mining for Image Classification. In: Proc. IEEE Conf. CVPR (2007)
25. Bosch, A., Zisserman, A., Muoz, X.: Image classification using random forests and ferns. In: ICCV (2007)
26. Dinh, T.B., Vo, N., Medioni, G.: Context Tracker: Exploring Supporters and Distracter in Unconstrained Environments. In: CVPR (2011)
27. Ozuysal, M., Fua, P., Lepetit, V.: Fast keypoint recognition in ten lines of code. In: CVPR (2007)
28. Viola, P., Platt, J.C., Zhang, C.: Multiple Instance Boosting for Object Detection. In: Proc. Neural Information Processing Systems (2005)
29. Mei, X., Ling, H.: Robust Visual Tracking using L1 Minimization. In: ICCV (2009)