A ROBUST PART-BASED TRACKER

Wei Zhou, Liansheng Zhuang, Nenghai Yu

MOE-Microsoft Key Laboratory of Multimedia Computing and Communication, University of Science and Technology of China
zhouwei1@mail.ustc.edu.cn, lszhuang@ustc.edu.cn,ynh@ustc.edu.cn

ABSTRACT

In this paper, we propose a new method for modeling appearance variances in generic object tracking task. Although object tracking has been studied by many researchers for a long time, there are still many challenging problems, which is mainly due to the complex variances of object's appearance. While most of traditional methods using a global or pixel-wise approach, we proposed a part-based tracking framework. We divide an object region into several non-overlapping parts (note they are not semantic as limbs and head of a human), and then a local classifier is updated on-line for each part. We gain a global confidence map by applying these local classifiers to the next frame, and find the new location of target object, i.e. the peak of confidence map, using mean-shift. Our tracker runs real-time, and is robust to some kinds of appearance variance (e.g. change of illumination, occlusion, change of pose, deformation of shape, object/camera movement and so on). Experiments show that our method outperforms the other states of the art approaches, especially on dealing with occlusion.

Keywords— part-based tracking, appearance variance, on-line boosting, occlusion

1. INTRODUCTION

Visual object tracking is to locate the target object in each frame of a video sequence, given the initial location and scale. It is one of the challenging problems in the field of computer vision, and has many real valuable applications, such as video surveillance, driving assistant, human-computer interaction and motion based video analysis. Object tracking has been studied for a long time, and significant successes have been achieved in some specific domains, for example, in face and people tracking [1, 2, 3],however, there remain many challenging problems and visual object tracking is still far from solved.

The major problem is the flexible variability of object's appearance. It is due to many factors, both intrinsic and extrinsic. Extrinsic as changes of illumination, occlusion, and image noise, camera motion, changes of view point; intrinsic as changes of pose, deformation of shape, irregular movement and so on; besides of these, there are also other effects, e.g. clutter scene and similar objects. A robust tracker is claimed to deal with these variances.

Many appearance models had been proposed to handle these variances. Such as contour [4], template [5], subspace updating [6], mixture model [7], kernel based filters [8],classifier [9, 10, 11, 12, 13] and so on.Tracking is considered as a classification task in this paper too. We achieve robustness by two hands, on one hand we divide the object region into several spatial related parts, and update an online classifier to capture the variance of object appearance using on-line boosting [14] on each part; on the other hand we give each part a weight to estimate its reliability. While tracking on a new frame, each classifier does exhaustive search around the previous position of its part, then a confidence map is calculated from the output of all these classifiers, finally we locate the global target object by applying mean-shift analysis on the confidence map.

Our method is robust to object appearance variation. This is due to two reasons: first, our tracker inherits ability in capturing transformation of object appearance from on-line boosting, in this way, each local part classifier is updated by positive patches sampled from the corresponding parts of object region, and negative patches sampled around the object or from other parts of the object regions. Second, the robustness is ameliorated by dividing target object region into sub-regions. Because usually only some parts of object change significantly in a shot sequence, for instance partial occlusion or the head of a pedestrian, the tracker can make a decision mainly on the relatively stable parts and reduces the impact of the others. Another advantage of part-based method is that there are both competition and cooperation between parts, which result in the tracker overcoming drifting problem, the main drawback of on-line boosting tracker. Another contribution of this paper is a new method to measure tracking precision, which is needed but missing before.

This paper is organized as follows: the related works is presented in section 2, following with explanation for our method in detail, and in section 4, we show our experiments on challenging video sequences comparing to several state of the art trackers, the end is the conclusion.

This paper is supported by the National High Technology Research and Development Program of China (863)(No.2010ZX03004-003 & No.2008AA01Z117), National Natural Science Foundation of China (No.60933013),Research Fund for the Doctoral Program of Higher Education(No.20070358040), Science Research Fund of MOE-Microsoft Key Laboratory of MCC (No.07122809) and Science Research Fund of USTC for Young Scholars.

978-1-4244-7493-6/10/$26.00

2. RELATED WORKS

Recent years, tracking is often considered as a binary classification problem. The aim of tracking is to generate the trajectory of an object over time by locating its position in every frame of the video, given the location and size of target object in the first frame, this task can be formulated as training a classifier to distinguish the object from the background. Under this framework, all well-studied classification algorithms can be used for tracking, such as support vector machine (SVM), AdaBoost, Bayesian network, Multiple Instance learning [15, 16] and so on.

S. Avidan [17, 18] used SVM to learn a classifier, which distinguishes object from background. Latter he combined an ensemble of weak classifiers into a strong classifier in the framework of AdaBoost, to cope with variance of object's appearance [10]. The ensemble was updated by adding new weak classifier and removing worst weak classifier.

H. Grabner et al. proposed an on-line boosting framework [14] and then applied it to object tracking [13]. They updated the ensemble of weak classifiers during tracking, thus are able to deal with appearance changes of the object. By using simple features such as Haar-like wavelets, orientation histograms and local binary patterns, LBPs, the algorithm runs very fast. But the on-line boosting tracker has a crucial problem that "each update of the tracker may introduce an error which, finally, can lead to tracking failure (drifting)" [11], moreover, the tracker's ability of accounting for appearance changes is limited. So the authors proposed semi-supervised on-line boosting [11] to alleviate the drifting problem. He trained a classifier previously to offer prior knowledge, however, it doesn't improve the robustness to variety and occlusion, and if the appearance have changed significantly over time, the pre-trained classifier becomes meaningless.

Multiple instance learning is a powerful tool which achieves superior performance on object detection with variety of appearance and noise [15, 16]. Boris Babenko et al. embedded MILBoost into on-line boosting frame-work and proposed on-line MILBoost [9], in which updating are done over bags of image patches rather than single examples. On-line MILBoost tracker is more robust to appearance changes than on-line Adaboost and has a certain ability to deal with occlusion, however, online MIL is much more complex and not very stable, the tracker often shift to front object if there is a relatively large occlusion.

Bo Wu and Ram Nevatia proposed a human tracker [3], they used responses of a set of body part detectors (such as legs, torso and head) to form a joint likelihood model. This method is not suitable for a general object tracking, in which we don't have the prior structure knowledge. Fragment based tracker [19] is also a part based model this algorithm used a static appearance model based on integral histograms. A. Adam et al. track target object by matching patches sampled from previously frames. Fragment based tracker is robust to occlusion, but it can't handle appearance variance well, and can't run real time. We have made a performance comparison in section 4.3, our method outperforms Fragment based tracker, and almost 30 times faster

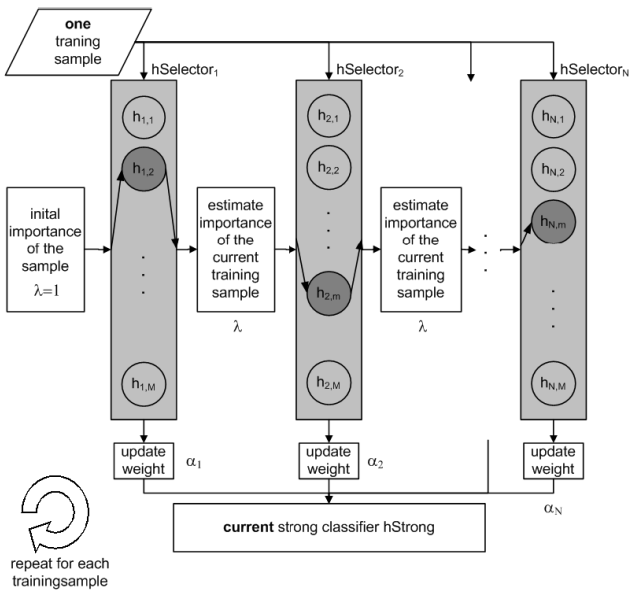


Fig. 1. principle of on-line boosting for feature selection [14].

while setting the search window half size to 15 pixels.

3. PART-BASED TRACKING

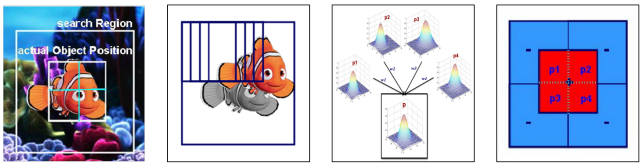
In this section, we first have a review of the on-line boosting algorithm, and then discuss our method in detail.

3.1. On-line boosting

Figure1 illustrates the principle of on-line boosting for feature selection proposed by H. Grabner. There are N selectors in this framework, each selector holds on M candidate weak classifiers, the weak classifiers of different selectors can be different but we usually use the same for simplicity. Once new sample arrives, each weak classifier of current selector is updated, and the weak classifier with least error rate is picked out and added into final strong classifier. Then the sample is passed to the next selector after weight updating. After passing all the selectors, the sample data is discarded, and the strong classifier has been updated.

3.2. Part-based tracking

Our tracker works in four main steps. Given the initial location and scale of target object, we first divide the whole object region into several non-overlapping parts, 22 for example. It is worth to remind that "part" is not defined semantic but only spatial. An on-line boosting classifier is trained for each part, they are updated independently using positive sample (the corresponding part region) and negative samples (image patches of the same size extract from neighborhood); then all 4 part classifiers are applied to the next frame and gain 4 local probability maps; thirdly, we calculate the confidence map from these local probability maps and locate target object by maximizing con-



(a) (b) (c) (d)

Fig. 2. An illustration of our tracking. Given the position at frame t, the object region is divided into 4 non-overlapping parts, 4 on-line classifiers updated independently (a). All 4 local classifiers detect target in the search region on frame $t+1$ (b) , then 4 local probability maps are combined into a global confidence map, then location of target is estimated (c), finally we update the weights of local classifiers and all 4 local classifiers are updated again(d).

fidence; finally the weights of each parts are updated based on their local probability maps. See Figure 2.

The global confidence map p is defined by:

$$P(i,j) = \sum_{k=1}^{K} w_k p_k(i + x_k, j + y_k) \tag{1}$$

$$p_k(i,j) = \frac{1}{1 + exp(-2C_k(i,j))} \tag{2}$$

where p_k is the local probability map, and w_k is the weight of part k. $C_k(i,j)$ is the output of the classifier of part k over the image patch located at position (i,j). $p_k(i + x_k, j + y_k)$ indicates the confidence that the target object locate at (i,j) with only part k being observed. $(x_k, y_k)_{k=1}^{K}$ denote the spatial bias of part k to the center of object region, which are evaluated at the first frame, and keep unchanged during tracking.

The location of target object, i.e. the peak of confidence map, can be estimated by maximizing confidence (in practice we use a soft maximizing by applying mean-shift to P in all of our experiments):

$$location = \underset{(i,j)}{argmax} P(i,j) \tag{3}$$

3.3. Updating

In our approach, we need to update weak classifiers and the weights of parts on-line to adopt the appearance changes of target object.

Update weak classifiers: we learn all weak classifiers the same as [1]. Each weak classifier h_k^j is composed of a Haar-like feature f^j and four parameters $(\mu_0, \sigma_0; \mu_1, \sigma_1)$, $h_k^j(x)$ is defined as:

$$h_k^j(x) = log(\frac{p(y=1|f^j(x))}{p(y=0|f^j(x))}) \tag{4}$$

Where $p(y=1|f^j(x)) \sim N(\mu_1, \sigma_1)$ and if the new positive data set is $\{x_i, y=1\}_{i=1}^{N}$, the updating rule is:

$$\mu_1 = \gamma\mu_1 + (1-\gamma)\frac{\sum_{i=1}^{N} f^j(x_i)}{N} \tag{5.1}$$

$$\sigma_1 = \gamma\sigma_1 + (1-\gamma)\sqrt{\frac{\sum_{i=1}^{N}(f^j(x_i) - \mu_1)^2}{N}} \tag{5.2}$$

It is similar to (μ_0, σ_0), here $p(y=1)$ and $p(y=0)$ are set equal to calculate $h_k^j(x)$ using Bayesian rule, and the parameter γ is set to 0.85 as [1] in all our experiments.

Weights updating: the weights of parts are set equal in initial. Then we update them based on the performance of classifiers on new frame. Once we have estimated the location of target object at frame t, the local score maps $\{I_k | I_k(i,j) = C_k(i,j)\}_{k=1}^{K}$ are used to update weights. We label patches sampled from the region around the right position in radius r as positive and patches out of neighborhood with radius R are labeled negative, then we fit distribution $N(\mu_1, \sigma_1)$ to positive scores and $N(\mu_0, \sigma_0)$ to negative scores, a threshold is defined by:

$$T_k = \frac{\sigma_0\mu_1 + \sigma_1\mu_0}{\sigma_0 + \sigma_1} \tag{6}$$

We calculate binary classification error rate e_k on these examples using T_k, then $\lambda_k = log((1-e_k)/e_k)$ is computed and normalized, the updating rule of weights is:

$$w_k = \alpha w_k + (1-\alpha)\lambda_k \tag{7}$$

According to the description above, the weights reflect the degree of variance of relevant parts. If a part of target object is stable, such as the head of a pedestrian, the classifier trained on previous frame will be discriminative on unseen frames, and then the weight of this part will increase. In the opposition, if the appearance of a part changes obviously, e.g. there is occlusion or other variance, the classifier learned has lower generalization, and would perform worse on coming frames, which will lead to weight decrease. In this way, we can focus attention on local parts that are credible, see Figure 3, we have a test on the video named occluded face (provided by author of [19], sequence and ground truth are both available at [20]), in which there is typical partial occlusion. In order to achieve a smooth curve, the parameter α is set to 0.9, and the curve is the average of 5 runs. The radius r and R are set to 3 and 5 respectively in all of our experiments.

4. EXPERIMENTS

In this section, we apply our tracker to 11 publicly available Video sequences, also one video captured by ourselves. We first examine the robustness of our tracker, and then make a comparison to other states of the art trackers.

4.1. Robustness to variance

We use area overlap rate (AOR) to measure the tracking performance. AOR is defined the acreage of overlapped region between tracker's prediction and ground truth divided the acreage of ground truth region. AOR ranges of zero to one, and one indicates accurate location and zero means lost. AOR is better than location error in pixels that we can see status of tracker from

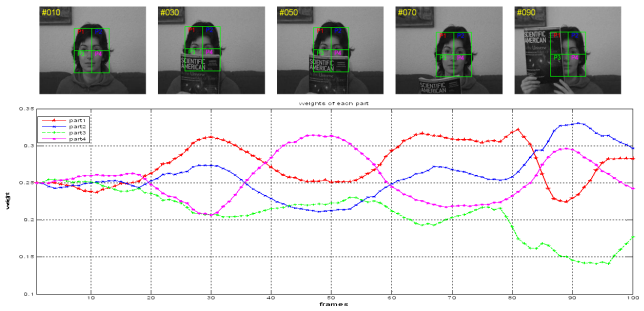


Fig. 3. The weights of parts change according to occlusion. The top row is image samples, the frame number and each part are labeled out, the bottom row is the weight-to-frame curves(red:top-left, blue:top-right, green:bottom-left, magenta:bottom-right). We can see that weights of steady parts increase while others decrease. And our tracker is robust to occlusion (the rectangles on images).

AOR-frame curve. In this experiment, we apply our tracker on 4 videos of typical appearance variance, they are: fish (illumination and camera motion) [21], david indoor (pose and illumination) [21, 22], occluded face 2 (pose and occlusion) [22] and walking woman (motion and occlusion) [20]. The ground truths of occluded face 2 and david indoor are available on the website, and we manually labeled fish and walking woman the center of target object for every 5 frames. Our tracker doesn't take scale into account and the size of target object is fixed during tracking. Figure 4 shows the average AOR-frame curves of several tracking methods on these videos. From these curves, we can see that our tracker is robust to all these appearance variance, including changes of pose and illumination, motion and occlusion. Though the appearance changes significantly, our tracker has never lost the target, while indeed others do. Our tracker achieves the highest AOR over all these 4 videos.

4.2. Comparison

We also compare our tracker to on-line MIL tracker (MILTracker) [9], Fragment based tracker (FragTracker) [19] and On-line Adaboost tracker (OABTracker) [13], which are states-of-the-art trackers. The parameter of FragTracker is the same as [19]. The number of candidate weak classifiers of both MILTracker and OABTracker is 250, and the number of chosen weak classifiers (i.e. the number of selectors in on-line boosting framework) is 50. To make a fair comparison, these numbers used in our method is divided by the number of parts, i.e. there are only 13 chosen weak classifiers and 63 candidates for each part if there are 4 parts, and 6 chosen weak classifiers and 28 candidates for each part if the target object region is divided to 9 parts. The parameter is set to 0.8 experimentally for weight updating.[]

It's important to declare that we divide target object region into 4 parts (14) for walking woman, caviar occlusion; 4 parts (22) for tiger1 and tiger2; 6 parts (23) for coke can and squeezer, and others are divided into 9 parts (33). The size of each part

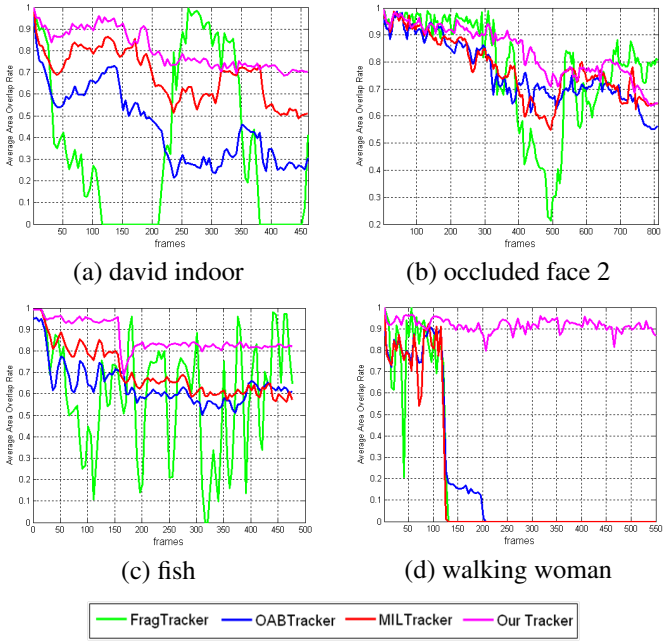


Fig. 4. Some tracking results measured in AOR vs frame curves, these curves are average of 5 runs except FragTracker.

should not be too small to make sure there is enough features to distinguish target region from surrounding. To our experience, the size of part is better to be larger than 1616 pixels.

We apply these algorithms to video sequences publicly available and popularly used in tracking literatures. They are:

Occluded face & walking woman: provided by A. Adam [20] with ground truth. There is only typical partial occlusion in the former, and in the latter, a woman was walking along a street with occlusion. Because the ground truth of the latter is not complete, we manually labeled it for every 5 frames from the first frame.

Tiger 1, Tiger 2, Coke can & Occluded face 2: provided by B. Babenko [22] with ground truth, all these videos contain frequent occlusions, the first three also contain fast motion and there are many different poses and out of plane rotations in the Tiger 1 & 2 sequences. The Coke Can sequence contains aspecular object, which adds some difficulty. Occluded face 2 contains changes of pose and rotation.

Fish, David indoor, David in trellis & Sylvester: provided by D. Ross [21], the first three contain challenging illumination changes and motion, the video about David also contain pose changes and there is fast motion and occlusion in Sylvester. The ground truth for David indoor and Sylvester is available at [22], and we manually labeled the others.

Girl, Caviar occlusion & Squeezer: the first comes from authors of [2], and available at [22] with ground truth, it contains significantly appearance variance. The second is from CAVIAR database [23], which contains occlusion between two people. The third is our own, the target is a card, and this sequence contains challenging shape deformation and occlusion.

The performance of tracking methods is listed in Table 1

Table 1. Comparison of different trackers. "Error" is the average object center location error measured in integer pixels and AOR is the average area overlap rate. All these data is the mean of 5 runs on each video clip. The red indicates the best performance and blue indicates the second best.

Data	MIL Tracker		OAB Tracker		Frag. Thacker		Our Tracker	
	Error	AOR	Error	AOR	Error	AOR	Error	AOR
fish	22	0.70	27	0.64	32	0.63	10	0.86
David in trellis	66	0.30	90	0.23	40	0.46	40	0.58
David indoor	23	0.69	49	0.45	69	0.36	13	0.81
Sylvester	12	0.73	23	0.60	24	0.65	20	0.64
Girl	32	0.67	48	0.53	25	0.76	21	0.78
Occluded face	27	0.76	44	0.63	7	0.93	14	0.87
Occluded face 2	20	0.77	22	0.76	20	0.78	14	0.84
Tiger 1	15	0.64	35	0.38	40	0.31	20	0.60
Tiger 2	17	0.62	34	0.36	40	0.21	25	0.47
Squeezer	42	0.40	24	0.61	25	0.55	10	0.84
Coke can	21	0.42	25	0.29	64	0.09	14	0.54
Caviar occlusion	26	0.67	42	0.55	8	0.83	8	0.84
Walking woman	124	0.18	100	0.21	128	0.19	25	0.77

and Figure 5. We can see that our method outperforms others on most video sequences. On Occluded face, FragTracker is more accurate, that due to invariance of woman face, so patches match very well. While on other challenging sequences with variance of target object, such as Walking woman, Sequeezer and Occluded face, FragTracker works worse. MILTracker performs better than our part-based tracker on tiger1, tiger2 and sylvester. There are mainly two reasons, first, the texture of these plush toys is too simple, and the size of target object is too small, the classifiers learned from each part are not so discriminative; second, there is very severely motion and appearance change in these videos, the on-line MIL is more powerful to capture appearance changes than on-line Adaboost. Even so, our part-based tracker still works better than OABTracker and FragTracker. Generally speaking, our part-based tracker works the best.

Our algorithm runs 17 fps (set search radius to 25, and 32 fps if set search radius to 15, which is usually broad enough) on a PC with Pentium? dual core CPU 2.0G. It works much faster than MILTracker and FragTracker.

5. CONCLUSION

In this paper, we proposed a novel part-based framework for general object tracking. Our tracker is composed of several part trackers. We update our trackers based on on-line boosting learning, and other classification based tracking methods can be easily adopted into our framework. It not only boosts up the robustness to variation, but also overcomes drifting problem. Experiments show that our tracker is more robust to challenging appearance variance, and outperforms other states of the art trackers.

6. REFERENCES

[1] L.M. Fuentes and S.A. Velastin, "People tracking in surveillance applications," *Image and Vision Computing*, vol. 24, no. 11, pp. 1165–1171, 2006.

[2] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*. Citeseer, 1998, pp. 232–237.

[3] B. Wu and R. Nevatia, "Detection and tracking of multiple, partially occluded humans by Bayesian combination of edgelet based part detectors," *International Journal of Computer Vision*, vol. 75, no. 2, pp. 247–266, 2007.

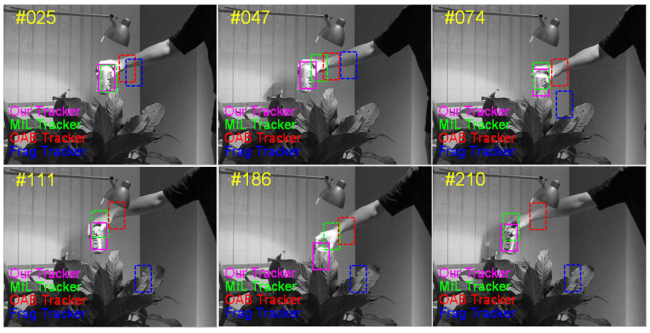
[4] M. Isard and A. Blake, "Condensation: conditional density propagation for visual tracking," *International journal of computer vision*, vol. 29, no. 1, pp. 5–28, 1998.

[5] I. Matthews, T. Ishikawa, and S. Baker, "The template update problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26.

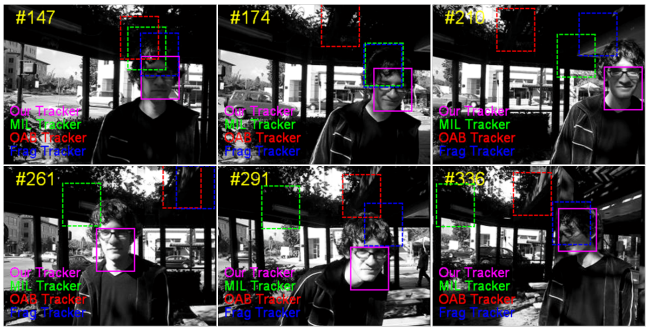
[6] D. Ross, J. Lim, R.S. Lin, and M.H. Yang, "Incremental learning for robust visual tracking," *International Journal of Computer Vision*, vol. 77, no. 1, pp. 125–141, 2008.

[7] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE transactions on pattern analysis and machine intelligence*, vol. 25, no. 10, pp. 1296–1311, 2003.

[8] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, 2003.



(a) coke can



(t) david in trellis

(c) squeezer

(d) walking woman

Fig. 5. Screenshots of tracking results. The rectangles of different color indicate different tracking methods, the solid magenta, i.e. our part-based tracker, has the highest veracity.

[9] B. Babenko, M.H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2009, pp. 983–990.

[10] S. Avidan, "Ensemble tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 261–271, 2007.

[11] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," *Proc. of the 10th European Conf. on Computer Vision*, pp. 234–247, 2008.

[12] F. Tang, S. Brennan, Q. Zhao, and H. Tao, "Co-tracking using semi-supervised support vector machines," in *Proc. of IEEE 11th Intl. Conf. on Computer Vision*, 2007, pp. 1–8.

[13] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *Proceedings of BMVC*, 2006, vol. 1, pp. 47–56.

[14] H. Grabner and H. Bischof, "On-line boosting and vision," in *Proc. of IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2006, pp. 260–267.

[15] S. Vijayanarasimhan and K. Grauman, "Keywords to visual categories: Multiple-instance learning forweakly supervised object categorization," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, 2008, pp. 1–8.

[16] P. Viola, J. C. Platt, and C. Zhang, "Multiple instance boosting for object detection," *Advances in neural information processing systems*, vol. 18, pp. 1417–1426, 2006.

[17] S. Avidan, M.E.V. Technol, and I. Jerusalem, "Support Vector Tracking," in *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2001, pp. 184–191.

[18] S. Avidan, M.E.V.T. LTD, and I. Jerusalem, "Support vector tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1064–1072, 2004.

[19] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, 2006, pp. 798–805.

[20] *http://www.cs.technion.ac.il/∼amita/fragtrack/fragtrack.htm*.

[21] *http://www.cs.toronto.edu/∼dross/ivt/*.

[22] *http://vision.ucsd.edu/∼bbabenko/project_miltrack.shtml*.

[23] *http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/*.