Published in final edited form as:

Multimed Model. 2016 ; 9516: 738–751. doi:10.1007/978-3-319-27671-7_62.

Robust Object Tracking Using Valid Fragments Selection

Jin Zheng^{1,2}, Bo Li^{1,2}, Peng Tian¹, and Gang Luo³

Jin Zheng: JinZheng@buaa.edu.cn; Bo Li: boli@buaa.edu.cn; Peng Tian: TianPengDigMedia@buaa.edu.cn; Gang Luo: Gang_Luo@MEEI.HARVARD.EDU

¹Beijing Key Laboratory of Digital Media, School of Computer Science and Engineering, Beihang University, Beijing 100191, China

²State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China

³Schepens Eye Research Institute, Mass Eye and Ear, Harvard Medical School, Boston, MA 02114, USA

Abstract

Local features are widely used in visual tracking to improve robustness in cases of partial occlusion, deformation and rotation. This paper proposes a local fragment-based object tracking algorithm. Unlike many existing fragment-based algorithms that allocate the weights to each fragment, this method firstly defines discrimination and uniqueness for local fragment, and builds an automatic pre-selection of useful fragments for tracking. Then, a Harris-SIFT filter is used to choose the current valid fragments, excluding occluded or highly deformed fragments. Based on those valid fragments, fragment-based color histogram provides a structured and effective description for the object. Finally, the object is tracked using a valid fragment template combining the displacement constraint and similarity of each valid fragment. The object template is updated by fusing feature similarity and valid fragments, which is scale-adaptive and robust to partial occlusion. The experimental results show that the proposed algorithm is accurate and robust in challenging scenarios.

Keywords

Fragments-based tracking; Structured fragments; Valid fragment selection; Harris-SIFT filter; Template update

1 Introduction

Object tracking is one of the important areas in computer vision, and it has a wide range of applications in intelligent surveillance, activity analysis, content understanding, video compression, human-computer interaction etc. [1]. At present, object tracking still remains challenging for situations in which the tracked object undergoes large and unpredictable changes in its visual appearance due to occlusions, deformation, scales change, varying illumination, as well as the dynamic and cluttered environments.

Object tracking has been largely formulated in a match-and-search framework. In this framework, the process of tracking usually includes two steps: appearance modeling and motion searching. It is critical for robust tracking to handle appearance changes of objects [2]. In general, appearance features include global features and local features. For global features, the lacks of spatial constraints cause them sensitive to the changes in object itself and the surrounding background. When there are appearance changes in tracked objects, global features also undergo great variations and oftentimes are hard to determine where the appearance changes happen. On the contrary, local features may provide the spatial information, and are more robust to object deformation and partial occlusion. For example, ALIEN (Appearance Learning In Evidential Nuisance [3]) proposes a novel object representation based on the weakly aligned multi-instance local features, and resides on local features to detect and track. FoT (Flock of Trackers [4]) estimates the object motion using local trackers covering the object. Matrioska [5] also resides on local features and uses a voting procedure, and the detection module uses multiple key point-based methods (ORB, FREAK, BRISK, SURF, etc.) inside a fallback model to correctly localize the object. Similarly, Yi et al. [6] describes the object using feature points, and utilizes motion saliency and description discrimination to choose local feature for tracking. Goferman et al. [7] proposes a Context-Aware Saliency Detection (CASD) method, which is based on four principles observed in the psychological literature. The most popular one is multiple-patch or fragment-based tracking. In this type of methods, the tracked object is divided into several fragments and each fragment is tracked independently based on features matching. The whole object is tracked using linear weighting scheme, vote map, or maximum similarity of the fragment location.

The advantages of constructing multi-fragments to represent the object include: (1) Multi-fragments provide the spatial information of the object template based on the relative spatial arrangements of different object parts. This offers an important advantage over the conventional region-based trackers in which only the region of interest is modeled by a single histogram with the loss of spatial information. (2) The use of fragments makes tracking more robust to partial occlusion and deformation etc. For example, even if some of the fragments are lost due to occlusion, the object could be located using the un-occluded fragments. (3) The structured template composed of multi-fragments is similar to the voting mechanism, and it can effectively prevent wrong decisions due to a large change of an individual fragment. (4) The dimensions of fragments are much lower than the dimensions of global features, therefore, it can minimize dimensionality computation effectively.

Some recent fragment-based tracking algorithms [8–16] are summarized. Almost all these algorithms derive features from color histogram, except [10] combines edge histogram and [13] combines gradient histogram. Using either overlapping [8–11, 14, 16] or non-overlapping [12, 13, 15] fragment partitions, the position of each fragment with respect to the center of the object is fixed and known in most of these algorithms [8, 10–16]. As a result, they may not deal with deformation and rotation well. Besides, each fragment is equally treated and just assigned a different weight based on its contribution. For estimating the contribution of each fragment, some algorithms combine the background [10, 12, 14], but still ignore the differences among these object fragments. Actually, similar object fragments of an object may cause confusion and therefore result in wrong tracking. Even

more, some algorithms locate the object just using one maximal similarity fragment [8, 10], but an individual fragment probably has a large drifting due to the unreliable similarity scores, especially in cluttered background. Even though a linear weighting location scheme of multi-fragments are used [9, 11–16], they may still fail as the number of unreliable fragments increases in complex environments, such as serious occlusion or deformation, cluttered background etc. In these situations, the color histogram similarity scores of incorrectly tracked fragments are unreliable, and the use of many such fragments would negatively affect tracking. Therefore, the selection of reliable patches, especially for the object enduring large deformation, is particularly important. It is worth mentioning that Kown [17] analyzes the landscape of local mode of the patch to estimate its robustness, and evolves the topology between local patches as the appearance of the object changes geometrically. Moreover, Kown emphasizes the robustness of patch, as well as the discrimination ability and the distribution of each patch [18].

As for motion searching, exhaustively searching [8, 11, 15], mean-shift [9, 10, 12, 14] and particle filter [13, 16] are commonly used. Exhaustively searching is time-consuming, and mean-shift probably falls into local optimization, and the original mean-shift formulation cannot handle orientation and scale. Particle filter [19] is a robust approach due to the ability to maintain multiple hypotheses of the object state. However, the complexity of object tracking based on particle filter can linearly increase along with the dimension of features and the number of particles. The feature selection and dimensionality reduction, as well as scale estimation are important for particle filter.

This paper proposes a fragment-based tracking algorithm using multiple validation measures to prevent erroneous tracking. The first validation uses a discrimination and a uniqueness measure of the local fragments, which describe the discriminative difference between the fragment and background, and the unique difference between a given fragment and the other fragments for the tracked object, respectively. This algorithm builds a pre-selection mechanism to determine discriminative and unique fragments (DUFrag), which can help reduce the fragment searching costs and overcome the problem of confusing fragments. In a further validation procedure, valid DUFrag (V_DUFrag) are selected using Harris-SIFT filter. Thus, V_DUFrag construct a valid structured object. The robustness of Harris-SIFT to scale changes, rotation and illumination changes can help improve the robustness of tracking, and the local characteristic of Harris-SIFT make the overall object tracking robust to partial occlusion. Importantly, Harris-SIFT allows us to make use of highly localized features in tracking. Thus, our method utilizes object features at 3 levels: (a) overall level – a group of structured fragments to describe the tracked object; (b) a medium level – each V_DUFrag is tracked based on features in local histogram; (c) a low level – Harris corners within each V_DUFrag. Furthermore, V_DUFrag are searched with certain spatial constraints. With these validation procedures, the flexible fragment grid does not necessarily result in object drifting.

2 The Proposed Algorithm

Assuming the object appearance is explicit and can be expressed in the detection phase, the algorithm selects DUFrag firstly. In the tracking process, it uses Harris-SIFT filter to

exclude the current invalid fragments, which may be due to occlusion or too large appearance change. The remaining V_DUFrgs are used for locating the object. The framework of the algorithm is illustrated in Fig. 1.

2.1 DUFrgs Selection Based on Discrimination and Uniqueness

Given an tracked object, firstly we divide the bounding box of the object template into several fragments. Our algorithm adopts adjacent non-overlapping partition. The fragment must be large enough to provide a reliable histogram. On the other hand, the number of fragments must be large enough to provide sufficient spatial information. Thus, an object is empirically divided into squares of approximately 20-pixel wide. HSV color histogram with 85 bins [20] is adopted, and Bhattacharyya distance is used to calculate the color histogram similarity.

Assume that the object template T is divided into N fragments, which is represented as positive samples set $\Omega_{Pos} = \{F_1^t, F_2^t, \dots, F_N^t\}$, and there are negative samples set $\Omega_{Neg} = \{\tilde{F}_1^t, \tilde{F}_2^t, \dots, \tilde{F}_M^t\}$ existing in the neighboring background. N is the number of positive samples, M is the number of negative samples, and t is the time. Discrimination is defined as D_i^t as formula (1).

$$D_i^t = \max(\text{sim}(F_i^t, \tilde{F}_j^t)) \quad i=1 \dots N, j=1 \dots M \quad (1)$$

Here, F_i^t denotes the i th object fragment at the t th frame, and $\tilde{F}_j^t$ denotes the j th background fragment at the t th frame. $\text{sim}(\cdot)$ is the similarity measure function. Searching the most similar negative fragment with the object fragment F_i^t , and the maximal similarity is denoted by D_i^t . If D_i^t is big enough, it means that the object fragment F_i^t is not discriminative from the background.

Uniqueness describes whether the estimated fragment could be distinguished from other object fragments. Hence, uniqueness U_i^t is measured by the maximum difference between the estimated object fragment and other object fragments.

$$U_i^t = \sum_{j=1}^N g(F_i^t, F_j^t, \tau) \quad i \neq j, i=1 \dots N \quad (2)$$

$$g(F_i^t, F_j^t, \tau) = \begin{cases} 1 & \text{sim}(F_i^t, F_j^t) \geq \tau \\ 0 & \text{else} \end{cases} \quad (3)$$

When U_i^t increases, it means that there are many object fragments having the similar features with the estimated object fragment, so the estimated object fragment is not unique, and it is ambiguous for locating the object.

According to D_i^t and U_i^t , the pre-selected fragments are decided by $\{F_i: D_i^t < \tau_1, U_i^t < \tau_2\}$, and they are DUFRags. The samples set of DUFRags is marked as $\Omega_{DU} = \{F_1, F_2, \dots, F_{N_{DU}}\}$, and the samples number of Ω_{DU} is N_{DU} . τ_1 is set according to the average discrimination capability in the worst selection at the initialization frame, and τ_2 is related with the number of positive fragments.

$$\tau_1 = \alpha_1 \cdot \frac{1}{N} \sum_{i=1}^N D_i^0, \quad \tau_2 = \alpha_2 \cdot N, \quad \alpha_1, \alpha_2 \in [0, 1] \quad (4)$$

For convenience, time t is omitted in the following.

Fragment selection is actually a process of features selection and dimension reduction. The obtained DUFRags allow the proposed algorithm to better handle distractive environments and interference from similar objects. It also could focus its computational resources on more informative regions.

2.2 V_DUFRags Selection Based on Harris-SIFT Filter and Spatial Constraint

In the tracking process, particle filter is used to get the candidate fragments at each frame, and the searching area is restrained in a small range. If the highest color-histogram similarity is used to locate the object (Fig. 2(a)), it is obvious that some of them are basically correct (such as fragments 1–5), the others are mismatched obviously (such as fragments 6–10). Such deformed fragments or noisy fragments sometimes drift off the object completely, and this could lead to completely wrong localization of the object. How to exclude the mismatching in flexible fragment grid is important. Actually, applying histogram calculation to validate fragments and then using those valid fragments with histogram similarity weight to track is analogous to circular reasoning, if validity is not clearly defined. This is the essential limitation for the existing weighted fragment-based tracking methods only using color histogram.

Thus, Harris-SIFT filter and spatial constraint are proposed for excluding the mismatching and achieving accurate tracking. For each DUFRags, the top-rank N_P locations which have higher color histogram similarities are selected. We believe that color histogram similarity is appropriate for confirming the candidates, but cannot locate the unique correct location accurately. For N_{DU} fragments in Ω_{DU} , there are $N_{DU} \times N_P$ candidate fragments marked as $\Phi = \{\bar{F}_1^1, \bar{F}_1^2, \dots, \bar{F}_1^{N_P}, \bar{F}_2^1, \bar{F}_2^2, \dots, \bar{F}_2^{N_P}, \dots, \bar{F}_{N_{DU}}^1, \bar{F}_{N_{DU}}^2, \dots, \bar{F}_{N_{DU}}^{N_P}\}$. Then we use Harris-Sift filter to identify the valid DUFRags. A Harris-SIFT feature is defined as a Harris corner and its SIFT feature vector, and it uses the efficiency of Harris corners, as well as the robustness of SIFT features. In detail, Harris corners in Ω_{DU} and Φ are firstly extracted respectively, and their SIFT feature vectors are computed. Then, the Euclidean distances of

SIFT feature vectors between Ω_{DU} and Φ are calculated for matching. Random Sample Consensus (RANSAC) is used [21] to exclude the mismatched points. To realize the spatial constraint, it is required that the valid candidate fragments are 8-neighbor connected with template fragment in Ω_{DU} .

As can be seen in Fig. 2(b), for each DUFragment, 10 candidate fragments having top-rank color histogram similarities are found using particle filter. Then, Harris-SIFT are computed for these candidate fragments, and five matched pairs are obtained (Fig. 2(c)). The five fragments in Ω_{DU} are marked by red, red, blue, yellow and orange squares. According to Harris-SIFT, the corresponding matched fragments in Φ are red; red and blue; blue and yellow; yellow and blue; orange and violet ones (Fig. 2(d)). Take the orange DUFragment on the person's knee as an example. Its matched points at the current frame are covered by one orange fragments and one violet fragment. Because the two candidate fragments are all 8-neighbor connected with the orange fragment in the template, they are valid fragments.

Thus, a candidate fragment in Φ is considered valid when there exists SIFT matching-pair and connectivity relationship for this fragment and its corresponding object template fragment in Ω_{DU} . All the valid fragments in Φ are marked as Φ_{V_DU} , which keeps the stable and structured parts for the object.

2.3 Object Location Based on V_DUFragment Fusion

After Harris-SIFT filtering, all the fragments in Φ_{V_DU} are fused to determine the object location. Suppose the matched set is $\Phi_{V_DU} = \{\bar{F}'_1, \bar{F}'_2, \dots, \bar{F}'_L\}$ at the current frame, the object tracking likelihood function using multi-fragment is define as

$$\hat{P}(T) = \sum_{j=1}^L P(T|\bar{F}'_j) P(\bar{F}'_j) \quad (5)$$

The mode of $P(T|\bar{F}'_j)$ denotes the location result using $\bar{F}'_j$, and $P(\bar{F}'_j)$ denotes the similarity belief of $\bar{F}'_j$. Similarity belief uses color histogram similarity measurement. As a voting process, all the fragments in Φ_{V_DU} participate to form a joint likelihood function using color similarity confidences and fragment displacements.

Because one matched Harris corner can be covered by several different candidate fragments, thus several $\bar{F}'_i$ probably have a substantial spatial overlap in images (such as Fig. 2(d)), their beliefs may be correlated for conquering non-overlapping fragments partition.

2.4 Feature Fusion Update

Object drift is largely due to imperfect template updating. It is a common updating strategy depending on the feature similarity between the best-matched candidate and the template. However, oftentimes the similarity measure is not reliable, for instance, histogram similarity

is prone to error due to occlusion, similar background and noises etc. Moreover, the threshold of similarity for making updating decision is not easy to determine.

This paper sets a rule that the template should not be updated in cases of occlusion or large deformation, otherwise it can be updated in cases of zoom in/out or rotation. Because Harris-SIFT is sensitive to occlusion and can provide some information about rotation, scale change, we propose an object template update strategy based on Harris-SIFT and color histogram measures, which is described as follow.

Update condition—If there is SIFT matched feature point pair in the fragment, or its color histogram similarity is higher than the average color histogram similarity of the fragment computed based on the previous frames, the fragment is regarded as in updating state. If all the fragments of the object are in updating state, the object update condition is satisfied.

Scale estimation—The object size estimated by particle filter is not accurate sometimes, especially when the object has irregular scale change. Therefore, our algorithm estimates scale change based on readily available SIFT matching-pairs. Define the bounding boxes of the matched SIFT points in the object template and the current frame as C_1 and C_2 , and the bounding box of object template as R_1 . Then calculate the distance between C_1 and R_1 , which is marked as $Gap_1 = [top, bottom, left, right]$. Then,

$$Gap_2 = \sqrt{\frac{Area(C_2)}{Area(C_1)}} \times Gap_1 \quad (6)$$

Add Gap_2 to C_2 , and the bounding box of the object in the current frame is gotten, which is denoted as R_2 .

For each tracking frame, once the update condition is satisfied, scale estimation is processed, and the object template is updated. Afterwards, DUFRags are chosen again.

3 Experimental Results

We first qualitatively tested the selection of DUFRags and V_DUFRags to demonstrate the validity of this procedure for object tracking. And then, tracking results with qualitative and quantitative evaluation are presented. Finally, computational cost and limitation are also analyzed in this section.

3.1 DUFRags and V_DUFRags Selection

Detecting discriminative and unique fragments, as well as occluded or deformed fragments of the object are the important steps for the proposed tracking algorithm. Figure 3 shows some results of DUFRags and V_DUFRags selection.

Figure 3(a) show the object and its neighboring background, and Fig. 3(c) show DUFRags found by our pre-selection method (labeled by yellow rectangles). As most existing

fragment-based methods [8–16] do not include pre-selection mechanism, we illustrate Context-Aware Saliency Detection (CASD) results [7] for reference. Similar to our method, CASD doesn't need prior knowledge or training database, and the detected saliency regions (shown as bright areas in Fig. 3(b)) is based on local low-level consideration, global considerations, visual organizational rules, and high-level factors. The experimental results show our DUFRags and CASD are well coincide with each other not only for high discrimination regions, such as eye, nose and mouth, but also for low uniqueness regions, such as hair and cheek in face images. Especially, when the background is simple, such as plane and dollar images, CASD and DUFRags both behave the similar results. However, when the background is complex, such as walker images, CASD is not exclusive to the object, for instance, the leg is darker than some neighboring background. On the contrary, DUFRags regard all the object fragments as discriminative and unique fragments. In case of lack of local feature, such as the vehicle roof and window in the minibus image, CASD may ignore the regions lacking small scale features, while DUFRags appear to be tracking relevant.

For V_DUFRags selection, the fragment selection method in Robust Fragments-based Tracking (RFT) [8] is used for comparison because it is similar to our algorithm. In RFT, the object is divided into 36 overlapping fragments and those with top 25 % ranked similarity are selected to represent the current candidate. The color histogram is used in similarity measure. As can be seen in Fig. 3(e), some occluded fragments or background fragments are selected incorrectly by RFT, especially when the occlusions are large or have the similar color as the object, while some useful fragments are ignored. On the contrary, the proposed Harris-SIFT filter could filter out occluded or highly deformed fragments. A typical example is shown by the dollar images. The tracked object undergoes great deformation and disturbance by a similar object. Our method is able to capture the corrected fragments of the tracked dollar bill. These examples demonstrate that overall V_DUFRags are robust and reliable fragments to describe the tracked objects.

3.2 Tracking Results

For evaluating the tracking performances, eight challenging video sequences used in previous publications [8, 19] were tested. These videos include partial occlusion, non-rigid deformation, scale change, background disturbances and motion blur. For each sequence, the location of the tracked object is manually labeled at the first frame. We evaluate our tracking algorithm against five state-of-the-art algorithms, including RFT [8], DFT [22], LOT [23], RCT [24] and LSST [25]. We used the source codes provided by the authors and ran them using optimal parameters for each algorithm.

From the exemplar results shown in Fig. 4, it can be observed that the proposed algorithm was among the best methods, as it could track objects with the highest accuracy for most of the sequences. Especially in case of very large scale change, such as Highway2. According to the quantitative evaluation in Table 1 (the average center location error) and Table 2 (the average overlap rate), LSST tracker gave better results than ours for DavidOutdoor, DavidIndoor and Face sequences. In these sequences, the appearances of the objects vary too

much due to non-rigid activity. Due to limitations of valid fragment updating, our method could not adapt to too severe deformation.

The tracking results on those challenging sequences show that the proposed algorithm was among the top 2 and mostly the top 1 in terms of average center location error and the average overlap rate.

3.3 Computational Cost

The proposed tracking algorithm was implemented in Matlab 2013 code, and was tested on a PC with a AMD Athlon (tm) II X4 635 Processor (2.9 GHz) and 4 GB RAM. For the aforementioned video sequences, the processing time of each sequence is listed in Table 3. The processing time of the algorithm mainly spent on color similarity computation in particle filter framework, SIFT features computation and matching. Apparently, the processing time was different depending on the size and texture of the object. In our experiments, the number of positive samples were determined by the size of the object, and the number of negative $M = 200$. In addition, the color histogram is computed using d -dimensional feature vectors, $d = 85$, and the size of fragment is $h \times w$, $w = h = 20$, $\alpha_1 = 0.3$, $\alpha_2 = 0.2$. The particle number for each DUFragment was $M_p = 100$, and $N_p = 10$.

It should be noted that, since each patch is tracked individually, parallel hardwares (such as multiple/multi-core processors or graphics programmable units) can be explored to further increase running speed.

3.4 Limitations and Future Work

Even if the tracked object has a large area of homogeneous color, as long as the object is distinct from the background, the proposed algorithm can use the peripheral discriminative fragments for tracking, such as the person in Caviar1 sequence. In rare cases where our algorithm may fail, the reasons could be that, all fragments are too homogeneous and do not contain unique fragments, or the object is blurry and does not contain discriminative fragments from the background, or Harris-SIFT tracking failed. Another limitation is that small objects (less than 20×20 pixels) are not suitable for fragment-based tracking or fragment selection. How to design more efficient local features, especially for homogeneous and blurry object, as well as small object, is our future work.

More experimental works should be implemented on benchmark sequences (e.g., OTB 2013, VOT 2013, VOT 2014), as well as comparing with more state-of-the-art tracking algorithms, to examine the algorithm's adaptive capability to severe, non-rigid deformation. This is also our future work.

4 Conclusion

The conventional features selection usually directly use the local features selected from previous frames to locate object while ignoring their validity, and this can easily lead to feature degradation and impair tracking robustness and accuracy. Unlike many existing algorithms, this paper proposes a robust fragments-based object tracking algorithm that uses discriminative and unique fragments pre-selection mechanism to determine DUFrags, and

then uses Harris-SIFT filter and spatial constraint to further identify the current valid fragments (VDUFrags). This process helps to exclude the occlusion or transformed fragments. Finally, the object is localized using a structured fragment template combining the displacement and similarity of each valid fragment.

We don't consider the fragments tracking directly determined by SIFT matching. The reason is mainly that our method is possible to use the multiple overlapped candidate fragments, which is coincident with multiple hypotheses of particle filter. In addition, in our approach SIFT deal with Harris corners at a low level, a fragment describes an object at a medium level, and the combination of multiple structured fragments represents the object at a high level. Such a multi-level framework can be more robust than methods that only rely on single level. The experimental results show that the proposed algorithm is accurate and robust in challenging scenarios that include severe occlusions and large scale changes.

Acknowledgments

This work is supported by the National Science Foundation of China (No. 61370124), China 863 Program (Project No. 2014AA015104), the National Science Foundation of China for Distinguished Young Scholars (No. 61125206), China Scholarship Foundation (No. 201303070205), and NIH/NIA grant AG041974.

References

1. Li X, Hu W, Shen C, et al. A survey of appearance models in visual object tracking. *ACM Trans Intell Syst Technol (TIST)*. 2013; 4(4):58.
2. Salti S, Cavallaro A, Di Stefano L. Adaptive appearance modeling for video tracking: survey and evaluation. *IEEE Trans Image Process*. 2012; 21(10):4334–4348. [PubMed: 22759454]
3. Pernici F, Del Bimbo A. Object tracking by oversampling local features. *IEEE Trans Pattern Anal Mach Intell*. 2013; 99:1. PrePrints.
4. Vojir, T.; Matas, J. Robustifying the flock of trackers. *Computer Vision Winter Workshop; IEEE*; 2011. p. 91-97.
5. Maresca, ME.; Petrosino, A. MATRIOSKA: a multi-level approach to fast tracking by learning. In: Petrosino, A., editor. *ICIAP 2013, Part II. LNCS. Vol. 8157*. Springer; Heidelberg: 2013. p. 419-428.
6. Yi, KM.; Jeong, H., et al. Initialization-insensitive visual tracking through voting with salient local features. *IEEE International Conference on Computer Vision (ICCV); IEEE*; 2013. p. 2912-2919.
7. Goferman S, Zelnik-Manor L, Tal A. Context-aware saliency detection. *IEEE Trans Pattern Anal Mach Intell*. 2012; 34(10):1915–1926. [PubMed: 22201056]
8. Adam, A.; Rivlin, E.; Shimshoni, I. Robust fragments-based tracking using the integral histogram. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition; IEEE*; 2006. p. 798-805.
9. Srikrishnan V, Nagaraj T, Chaudhuri S. Fragment based tracking for scale and orientation adaptation. *Comput Vision Graph Image Process*. 2008;328–335.
10. Jeyakar J, Babu RV, Ramakrishnan KR. Robust object tracking with background-weighted local kernels. *Comput Vision Image Underst*. 2008; 112(3):296–309.
11. Naik, N.; Patil, S.; Joshi, M. A fragment based scale adaptive tracker with partial occlusion handling. *TENCON 2009–2009 IEEE Region 10 Conference; IEEE*; 2009. p. 1-6.
12. Wang F, Yu S, Yang J. Robust and efficient fragments-based tracking using mean shift. *AEU-Int J Electron Commun*. 2010; 64(7):614–623.
13. Nigam C, Babu RV, Raja SK, et al. Fragmented particles-based robust object tracking with feature fusion. *Int J Image Graph*. 2010; 10(1):93–112.

14. Li, G.; Wu, H. Robust object tracking using kernel-based weighted fragments. 2011 International Conference on Multimedia Technology (ICMT); IEEE; 2011. p. 3643-3646.
15. Dihl, L.; Jung, CR.; Bins, J. Robust adaptive patch-based object tracking using weighted vector median filters. 24th SIBGRAPI Conference on Graphics, Patterns and Images (Sibgrapi); IEEE; 2011. p. 149-156.
16. Erdem E, Dubuisson S, Bloch I. Fragments based tracking with adaptive cue integration. Comput Vision Image Underst. 2012; 116(7):827–841.
17. Kwon, J.; Lee, KM. Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping monte carlo sampling. Proceedings of IEEE Conference on Computer Vision and Pattern Recognition; 2009.
18. Kwon J, Lee KM. Highly nonrigid object tracking via patch-based dynamic appearance modeling. IEEE Trans Pattern Anal Mach Intell. 2013; 35(10):2427–2441. [PubMed: 23969387]
19. Isard M, Blake A. Condensation—conditional density propagation for visual tracking. Int J Comput Vision. 1999; 29(1):5–28.
20. Yang J, Wang J, Liu R. Color histogram image retrieval based on spatial and neighboring information. Comput Eng Appl. 2007; 43(27):158–160.
21. Chen, HY.; Lin, YY.; Chen, BY. Robust feature matching with alternate hough and inverted hough transforms. IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE; 2013. p. 2762-2769.
22. Sevilla-Lara, L.; Learned-Miller, E. Distribution fields for tracking. IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE; 2012. p. 1910-1917.
23. Oron, S.; Bar-Hillel, A.; Levi, D., et al. Locally orderless tracking. IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE; 2012. p. 1940-1947.
24. Zhang, K.; Zhang, L.; Yang, M-H. Real-Time compressive tracking. In: Fitzgibbon, A.; Lazebnik, S.; Perona, P.; Sato, Y.; Schmid, C., editors. ECCV 2012, Part III. LNCS. Vol. 7574. Springer; Heidelberg: 2012. p. 864-877.
25. Wang, D.; Lu, H.; Yang, MH. Least soft-threshold squares tracking. IEEE Conference on Computer Vision and Pattern Recognition (CVPR); IEEE; 2013. p. 2371-2378.

Fig. 1.
The framework of the proposed algorithm

Fig. 2.
Float fragments with structure constraint (Color figure online)

Fig. 3. DUFRags & V_DUFRags selection (a) object and background, (b) CASD, (c) DUFRags, (d) occluded/deformed candidate, (e) RFT, (f) V_DUFRags (Coloe figure online)

Fig. 4.
Sample tracking results for challenging sequences

Table 1

Average center location error (in pixels)

Sequence	RFT	DFT	LOT	RCT	LSST	Ours
OcclusionI	6.2	22.4	22.0	21.5	7.1	1.5
DavidOutdoor	65.3	58.6	64.6	103.3	5.1	5.5
CaviarI	5.2	9.9	2.5	100.4	2.5	2.5
Highway	49.4	8.8	44.7	11.0	13.4	3.2
DavidIndoor	52.1	17.2	84.7	12.5	4.8	7.9
Deer	91.0	255.9	89.9	201.4	8.9	5.2
Face	49.6	135.7	34.1	158.8	4.3	8.5
Dollar	11.6	3.7	68.9	15.4	3.0	1.2
Avg.	41.3	64.0	51.4	78.0	6.1	4.4

Table 2

Average overlap rate

Sequence	RFT	DFT	LOT	RCT	LSST	Ours
OcclusionI	0.89	0.73	0.51	0.70	0.88	0.96
DavidOutdoor	0.41	0.57	0.50	0.24	0.77	0.78
CaviarI	0.70	0.66	0.85	0.23	0.83	0.85
Highway	0.16	0.34	0.30	0.35	0.32	0.57
DavidIndoor	0.25	0.44	0.15	0.51	0.75	0.52
Deer	0.11	0.07	0.12	0.07	0.59	0.64
Face	0.37	0.36	0.48	0.16	0.89	0.79
Dollar	0.69	0.88	0.20	0.66	0.89	0.95
Avg.	0.45	0.51	0.39	0.37	0.74	0.76

Table 3

Processing time

Sequence	Image size (pixels)	Object size (pixels)	Length	Average processing time (s/f)
Occlusion1	352 × 288	116 × 146	898	0.6674
DavidOutdoor	640 × 480	39 × 131	252	0.3340
Caviar1	384 × 288	31 × 80	382	0.0798
Highway	320 × 240	43 × 63	45	0.1433
DavidIndoor	320 × 240	60 × 93	462	0.2891
Deer	704 × 400	101 × 71	71	0.0884
Face	640 × 480	94 × 110	492	0.4949
Dollar	320 × 240	58 × 90	327	0.3892