# A Particle Swarm Optimization Algorithm with Local Sparse Representation for Visual Tracking

Xu Cheng
School of Information Science and Engineering, Southeast University, Nanjing, 210096, China
Email: xcheng@seu.edu.cn

Nijun Li, Tongchi Zhou, Lin Zhou and Zhenyang Wu
School of Information Science and Engineering, Southeast University, Nanjing, 210096, China
Email: {lnjleo, zhoutongchi, linzhou, zhenyang}@seu.edu.cn

***Abstract*—Handling appearance variations caused by the occlusion or abrupt motion is a challenging task for visual tracking. In this paper, we propose a novel tracking method that deals with the appearance changes based on sparse representation in a particle swarm optimization (PSO) framework. First, we divide each candidate state into multiple structural patches to cope with the partial occlusions of the object. Once the object is lost, we present an object's recovery scheme by the scale invariant feature transforms (SIFT) correspondence between two frames to reacquire the rough object position. Then the tracking state is searched in the vicinage of the rough object position using the PSO iteration. In addition, an online dictionary updating mechanism is presented to capture the object appearance variations. The object information from the initial frame is never updated in the tracking, while other templates in the dictionary are progressively updated based on the coefficients of templates. Compared with several conventional trackers, the experimental results demonstrate that our approach is more robust in dealing with the occlusions and abrupt motion variations.**

***Index Terms*—Visual tracking, Particle swarm optimization, Dictionary learning, Sparse representation, Scale invariant feature transform (SIFT)**

## I. Introduction

Visual tracking has been a hot topic in the field of computer vision, especially for application of surveillance, action recognition and human computer interface. Although a great deal of works has been proposed with the development of efficient schemes and fruitful applications, it still remains an object drifting problem which is caused by occlusions, background clutter, illumination and viewpoint changes.

Current tracking methods can be classified into either discriminative or generative approaches. Generative module tackles tracking as searching regions most similar to the learned appearance model. These methods, either templates based [1-4, 7, 8, 27] or subspace models based [5, 6], can work well when the size of training data is small. Adam et al. [3] introduce a part based tracker which splits the object state into multiple sub-regions to overcome the partial occlusions of the object. But the tracker ignores the problem of template updating. In [4], an effective tracking method which decomposes the observation and motion models into multiple basic corresponding models to capture a wide range object changes is presented. In [6], the low dimensional subspace learning scheme is explored to model the variations of object, which is robust to the illumination variations. PSO based tracker [7] that exploits the interactive swarms to track objects; the main drawback of this method is that subspace representation based method cannot handle the large object changes in neighboring frames. To overcome this problem, an efficient tracker [8] with the SIFT feature points correspondence and multiple fragments, which can handle the partial occlusions and large variations of the object motion, is proposed under the PSO framework.

Discriminative module formulates the tracking as a classification problem which aims to separate the object from the background region. Avidan [9] combines a set of weak classifiers into a strong to label a pixel as belonging to either the object or the background. Later in [10], an online boosting approach is used to update the discriminative features. However, they only use one positive sample (i.e. tracking result) and multiple negative samples to update the classifier. If the positive sample is imprecise, then an undesirable tracking result is added to the template set or training classifier, the entire tracking scheme will degrade, even failure finally. Babenko et al. [11] propose a multiple instance learning (MIL) scheme which puts the positive samples and negative ones into bags to handle the drifting problem. Zhang et al. [12] further extend MIL tracker, they consider the sample importance which is integrated into the online learning procedure, leading to a more robust estimation. Kalal et al. [13] propose P-N tracker using underlying structure of positive and negative samples to learn the classifier.

Although afore-mentioned approaches perform well in some scenes, it is sometimes easy to lose the object due to intrinsic and extrinsic factors. Recently, several attempts based on sparse representation have been made to address these issues. Mei et al. [14] treat the L1 tracking as finding a sparse representation in the template

Manuscript received January 26, 2014; revised February 10, 2014; accepted April 9, 2014.
Corresponding Author: Zhenyang Wu, zhenyang@seu.edu.cn


doi:10.4304/jcp.9.9.2230-2238

subspace. The representation is then exploited for visual tracking under the particle filter framework. The intensity feature in L1 tracker is explored to represent the object appearance since it is robust to occlusions and other tracking challenges. However, the drawback of the intensity feature is that it is sensitive to shape deformation of object and may fail when there is similar object clutter.

To overcome the above problem, we employ the SIFT descriptor as the appearance representation to describe the object more accurately. In this paper, our contribution is four-fold. (1) We propose a simple yet efficient tracking method based on the structural local sparse representation to cope with the complex challenges (e.g., appearance and abrupt motion variations). A candidate state is divided into multiple spatial patches and each patch denotes a fixed part of the object. During the tracking, we exploit the both partial information and spatial information of the object to continuously track the object especially in cases of background clutter with similar appearance model and partial occlusion. (2) SIFT feature point correspondence can recover the lost object from the tracking scene. The tracked object may fail due to large motion changes of the object between two consecutive frames. (3) The initial dictionary is constructed by the first ten frames of the tracked object. The object template from the first frame is never changed in the course of tracking. Other templates are selectively updated based on the sparse representation coefficient of the recent observation. (4) Compared with the particle filter based tracker, our approach can achieve the comparable tracking accuracy with much lower number of particles under the PSO framework.

This paper is structured as follows. We begin by reviewing the related work in the next section. Section Ⅲ briefly reviews PSO algorithm and L1 tracking method. The proposed algorithm is presented in Section Ⅳ for more details. Section Ⅴ is devoted to conclusions.

## II. Related Works

Sparse representation for visual tracking has drawn much interest and been successfully applied in numerous vision applications. With sparsity constraints, a tracking candidate is sparsely represented in the form of linear combination of only a few atoms of the dictionary which is composed of object templates and trivial templates. The advantage of the sparse representation lies in the robustness to background clutter, occlusion and noises. Bao et al. [15] develop a new L1 tracker that runs in real time using accelerated proximal gradient (APG) algorithm. The authors add L2 norm regularization on the coefficients associated with the trivial templates to the L1 minimization model to improve the tracking performance. Liu et al. [16] propose a local sparse representation scheme for visual tracking. The object appearance is modeled by the histograms of sparse coefficients. The tracking results are decided via mean shift of voting maps. The main drawback of this approach is that it can not handle the similar object appearance well using a static local sparse dictionary. Object tracking with fixed template or dictionary learned from the first frame is likely to fail due to large appearance variation. Therefore, a time-variant object template during the tracking process is essential to capture the object appearance variations. In [17], authors propose a block-division based covariance feature multiple object tracking algorithm under the sparse representation framework. Chen et al. [18] propose a combination method of learned appearance model and sparse representation for robust tracking. The appearance of an object is modeled by multiple linear subspaces, so it is sensitive to the abrupt appearance variations. In [19], a collaborative tracker that combines the generative module and discriminative module is used to cope with the tracking challenging in the course of tracking. Compared to the afore-mentioned trackers that pursue the sparse representation independently, Zhang et al. [20] present a multi-task sparse coding tracking method that casts each particle as a single task and mines correlations among different tasks to achieve a more robust tracking and reduce the computational cost.

Although sparse representation has been applied to the real-world tracking by searching the best candidate with the smallest reconstruction error, most sparse representation based trackers only consider the holistic representation and do not make full use of the local information of the object. Hence these methods may fail with more possibility when occlusion or similar background clutter occurs.

Different from [14], the proposed scheme shows three improvements. First, our approach can handle the partial occlusion well due to using the patch information of object. When the partial occlusion occurs, we can make full use of the unoccluded parts of the object to continuously track it. Second, our algorithm recovers tracking by SIFT feature matching scheme between two frames if the object is lost. Matched SIFT feature points in current frame vote to the object's center using the relative position of each SIFT point and center position from the last frame to reacquire the object. Finally, an online dictionary updating scheme can adapt the appearance changes of the object with less possibility of drifting and reduce the influence of the occluded object template as well.

## III. Motivation

### A. Particle Swarm Optimization Tracking Framework

In this section, we briefly review the PSO tracking framework, and then formulate the visual tracking as the sparse representation problem.

Particle Swarm Optimization (PSO) [21] is a bio-inspired population based optimization algorithm which is used to search the global optimization by iteration. Each particle is a candidate solution and its movement depends on two important factors: the individual best state of each particle and the global best state among all the particles. Based on these two factors, each particle updates its velocity and position in the *n*th iteration as follows:

$$v^{i,n+1} = w^n v^{i,n} + c_1 u_1 (p^i - x^{i,n}) + c_2 u_2 (g - x^{i,n}) \quad (1)$$

$$x^{i,n+1} = x^{i,n} + v^{i,n+1} \quad (2)$$

where $i$ and $n$ denote the $i$th particle and the $n$th iteration, respectively; $v^{i,0}$ is its initial velocity; $w^n$ is the inertial weight; $p^i$ is individual best position in the previous iteration and $g$ is the best position of the swarm; $c_1$, $c_2$ are acceleration constants; and $u_1$, $u_2$ are uniformly distributed random numbers. It is clear that the movement of each particle relies on three components: inertial velocity, cognitive effect and social effect. The cognitive effect represents the evolution of particle based on its own best position, and social effect indicates the evolution of particle according to the cooperation among the swarm.

In the PSO algorithm, each particle $x^i$ has a corresponding fitness value $f(x^i)$. In each iteration, the individual best position ($p^i$) is updated by Eq.(3) if the fitness value of current state is greater than the previous best state, otherwise the previous best state will be kept. The global best ($g$) that has the highest fitness value among all individual best state is the global optimal solution. This can be formulated as follows:

$$p^i = \begin{cases} x^{i,n} & f(x^{i,n}) \geq f(p^i) \\ p^i & f(x^{i,n}) < f(p^i) \end{cases} \quad (3)$$

$$g = \arg\max_{p^i} f(p^i) \quad (4)$$

### B. Sparse Representation based Tracking Method

L1 tracker tackles tracking as the minimum error reconstruction via a regularized L1 minimization problem.

$$\min_{\alpha} \frac{1}{2} \left\| \mathbf{X} - \mathbf{D}\alpha \right\|_2^2 + \lambda \left\| \alpha \right\|_1 \quad \text{s.t.} \quad \alpha \geq 0 \quad (5)$$

where $\mathbf{D}=[\mathbf{T}, \mathbf{I}] \in \mathbb{R}^{d\times(n+d)}$ is composed of object templates $\mathbf{T}$ and trivial templates $\mathbf{I}$. $d$ is the dimension of dictionary and $n$ is the number of object templates. Each column in $\mathbf{T}$ is processed by reshaping pixels of a candidate object into a column vector; and $\mathbf{I} \in \mathbb{R}^{d\times d}$ denotes a unit matrix. An object candidate state $x \in \mathbb{R}^d$ can be approximated by a linear combination of $\mathbf{D}$. Finally, the candidate state with the minimum reconstruction error is regarded as the tracking result under the particle filter framework.

$$\hat{x} = \arg\min_{x} \frac{1}{C} \exp\{-\eta \left\| x - \mathbf{D}\alpha \right\|^2\} \quad (6)$$

where $\alpha$ is obtained by solving the L1 minimization; $C$ is a normalization factor and $\eta$ is a constant.

## IV. Proposed Tracking Scheme

In this paper, we consider tracking under a new mechanism which is a combination of local search and global matching. The workflow of the proposed tracking algorithm is illustrated in Figure1. When the object moves smoothly over time, the PSO local searching based tracker can be exploited to determine the object position directly. However, when the tracker cannot perform well due to abrupt motions, occlusion, and illumination changes, the global matching scheme is performed to recover the object from the cluttered background. We hold the opinion that the tracker loses the object if the fitness value in current frame is less than a given threshold. In this case, we extract the SIFT descriptors from the current frame which causes a tracking failure and match SIFT keypoints with the object state of the last frame. The refined matching points, which are further obtained by the RANSAC scheme from the matching set, vote the object center using the memorized position between the each SIFT point and object's center position in the last frame. We can achieve the rough object position estimation by this means. Then the best particle state $x_t$ is selected to represent the current state of the object under the PSO framework. Finally, the dictionary is updated in each fixed length frame interval to capture the appearance variations.

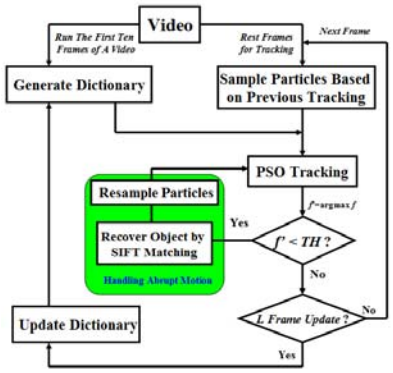



Figure 1. Overview of the proposed tracking algorithm.

### A. The Dictionary Generation

The initial dictionary $\mathbf{D}$ is constructed to model the object appearance. In our work, we adopt the first ten frames of a video to generate the initial dictionary. The initial object state is selected manually in the first frame, and then we run simple tracker (such as mean shift tracker [1]) in the first ten frames to obtain the rest object samples which contain rich object appearance information due to the motion of object in the scenes. We exploit overlapped patches inside each sample with a spatial layout and reshape the SIFT descriptor of each patch into a column vector. The patches altogether for all samples are regarded as the dictionary $\mathbf{D} \in \mathbb{R}^{d\times m}$, where $d$ denotes the dimension of dictionary; $m$ is the number of atoms in the dictionary. Our approach shares the same idea as the part based tracker that can handle the partial occlusion. The dictionary $\mathbf{D}= [\mathbf{T}_1,\mathbf{T}_2,\ldots,\mathbf{T}_N]$ is further divided into $N$ sub-matrices corresponding to $N$ patches of a candidate state ($N$=9 in our article). Each sub-matrix $\mathbf{T}_j$, which contains 10 atoms (columns), can be used to linearly represent the corresponding patch of a candidate state.

### B. PSO Local Searching

The scale invariant feature transform (SIFT) descriptor [22] has been proved to outperform other types of image

descriptor. We choose SIFT as the appearance descriptor of the object in our scheme and refer readers to a comprehensive survey for more details.

We apply an affine image warping to model the object motion between two consecutive frames [6]. When a new frame arrives, particles based on the object state of the last frame are randomly sampled according to Gaussian distribution at the beginning of PSO iteration, and each particle is regarded as an object candidate state. The state transition distribution is utilized by

$$x_{t,i} \sim N(x_{t-1}, \Sigma) \tag{7}$$

where $x_{t-1}$ is the state at the time $t$-1; the state $x_t$ is modeled with six parameters of the affine transformation; $\Sigma$ is the diagonal covariance matrix whose elements are variances of affine parameters. We then extract the SIFT feature from overlapped patches of each candidate state to represent the object and encode them with the dictionary. Let $\mathbf{X}=[x_1,\ldots,x_N]$ denotes the SIFT features extracted from all the patches of a candidate state, the sparse coefficient vector $\alpha$ for coding $\mathbf{X}$ can be calculated by Eq.(5). Each column of $\alpha$ denotes the coding result of one patch. Finally, all patches are combined within one candidate state for evaluating the fitness value. The score of one patch is obtained by

$$\varepsilon_{i,j} = \| x_{i,j} - \mathbf{T}_j \alpha \|_2, \quad j = 1,2,...,N \tag{8}$$

where $\varepsilon_{i,j}$ denotes the reconstruction error of the $j$th patch within the $i$th particle. In this paper, the fitness function, which is used to evaluate the global best state of particles, is defined as follows.

$$f(\mathrm{x}_{t,i}) = \exp\left\{-\beta \sum_{j=1}^{N} \varepsilon_{i,j}\right\} \tag{9}$$

$$\hat{\mathrm{x}}_t = \arg\max_i f(\mathrm{x}_{t,i}) \tag{10}$$

where $\mathrm{x}_{t,i}$ denotes the $i$th object candidate state at the time $t$; $\beta$ is a scaling factor ( $\beta$ =0.02 in this paper). The fitness function $f(\mathrm{x}_{t,i})$ will be larger if the particle (candidate state) is more similar to the template. The individual best state and global best state of particles are updated by evaluating a fitness function at PSO iteration. The larger fitness value of a candidate state is, the more possibility of global optimum is. Finally, we cast the global optimal candidate state as tracking result by Eq.(10).

### C. Object's Recovery from the Failure

Since the range of sampling is limited around the previous object position. Most conventional trackers may lose the object easily and be difficult to recover tracking when the object undergoes a large shift or drastic illumination variations between two frames. In this case, we need to reinitialize the lost object using a global searching scheme. It is worth mentioning that the SIFT point matching plays an important role in the feature points based tracking methods. A frame-by-frame matching is obtained if the ratio of distances between the first and the second nearest neighbors is less than a predefined threshold. In Figure2, some matched keypoints are labeled in neighboring frames.

The following describes how to recover the lost object in our tracker. Our algorithm first detects the SIFT feature points in current frame which causes a tracking failure. Then the SIFT points, which are successful matched with the previous object state in the last frame, are preserved. We further resort to RANSAC scheme [23] to obtain the refined matching points set.

Furthermore, we apply the relative position between each SIFT point and object center from the last frame to estimate rough object position in current frame. Suppose that at frame $t$-1 the tracker memorizes the relative position of each SIFT keypoint and object center within the object tracking state. At frame $t$, each keypoint from the refined matching set votes the object's center position using the memorized positions of the last frame. These voted positions then are clustered into $K$ cluster centers. Finally, the average of these cluster centers is considered as the rough object position. Therefore, we can find the rough position of the object by this means.

Consequently, particles are resampled in the vicinage of rough position in current frame and tracking result that has the largest similarity to the template is reacquired by PSO iterations. The main steps of the proposed tracking approach are listed in Algorithm 1.

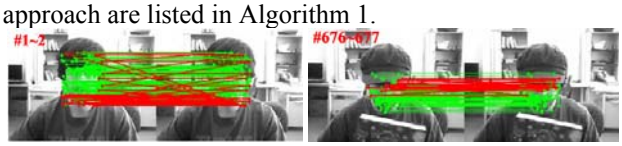



Figure2. SIFT feature matching between the consecutive frames on Faceocc2 video (green lines denote the correct matches; Red lines denote the outliers).

### D. Dictionary Update

The appearance of object during the tracking process may change drastically due to intrinsic (e.g., shape and pose variations) and extrinsic factors (e.g., occlusion and illumination changes). A time-invariant appearance template cannot capture appearance changes and may lead to drift. In most tracking applications, the tracker must simultaneously handle the changes of both the object and the environment. Therefore, it is necessary to update the dictionary once every $L$ frames ($L$=7 in all the experiments).

Similar to [8], if full occlusion is detected, the dictionary updating process will be paused to avoid introducing the background noise into object templates. However, the object information from the first frame is never changed in the tracking process to alleviate the problem of drifting.

Which sub-matrix one patch x corresponds to, satisfying

$$\min_{\mathbf{T}_j} \| x - \mathbf{T}_j \alpha \|_2 \quad 1 \le j \le N \tag{11}$$

In this paper, we update the dictionary according to the coefficients of the object atoms and use the recent observation with the smallest error in a fixed interval $L$ for updating the dictionary. For the $j$th patch $v_j$ of the observation, we find the atoms $t_{\hat{k}}$ with the smallest coefficient within the corresponding sub-matrix $\mathbf{T}_j$.

$$\hat{k} = \arg\min_{2\le k\le 10} \alpha_k \quad (12)$$

where $\alpha_k$ is the $k$th element of sparse coefficient $\alpha$. Then the atom $t_{\hat{k}}$ is replaced by the current patch $v_j$.

At the same time, the atoms $t_l$ with the largest coefficient is also replaced by

$$l = \arg\max_{2\le k\le 10} \alpha_k \quad (13)$$

$$\tilde{t}_l = t_l + \gamma v_j \quad (14)$$

where $\gamma$ is a learning rate (0.05 in the experiments); $\tilde{t}_l$ and $t_l$ indicate the updated atom and last updated atom, respectively.

### *E. Summary of the Pseudo-Algorithm*

To summarize the proposed tracking algorithm, described in above-mentioned section, the pseudo code is summarized as follows.

**Algorithm 1 Tracking Algorithm**

**Input:** Image sequence $F_t$
**Output:** Object state $S_t$ in each frame
**Initialization:** Mark a bounding box for the object in the first frame and sample overlapped local image patches with fixed spatial layout within the marked object state, whilst extracting SIFT features of each patch and constructing the initial dictionary using the first ten frames of a video sequence.
**Tracking:**
**for** $t$ from 11 to the last frame do
**Local search:** 1. Get particles $\{x_{t,i}\}_{i=1}^{n}$ from the Eq.(7) based on the previous object state and extract SIFT descriptors from each candidate particle state.
2. Evaluate the fitness value of each candidate state by Eq.(3).
3. Update the individual best state for each particle and global best state among all the particles.
4. Carry out the PSO iteration using Eq.(1) and Eq.(2), until the convergence criteria are satisfied. The highest fitness value among all the particles is the global best position ($g$).
5. **if** $f(g) \ge TH$ ($TH$ which is a threshold to determine whether the object is lost is set to 0.25 in our experiments)
6. Obtain the object state $S_t$=$g$.
**else**
**Global matching:** 7. Match the SIFT feature detected by current frame $t$ with the previous object state in the last frame and further refine the matching set by RANSAC scheme.
8. Each SIFT point in the refined matching set votes to the object's center using the memorized position between the each SIFT point and center position from the last frame, and the rough object position estimation is determined by the average of the cluster of these voted centers.
9. Resample particles $\{x_{t,i}\}_{i=1}^{n}$ in the vicinage of rough object position.
10. Find the best state g' by the PSO iteration and obtain object estimation: $S_t$=$g$'.
**end if**
11. Update the dictionary once every $L$ frames.
**end for**

## V. EXPERIMENTS

### *A. Parameter Setting*

To evaluate the effectiveness of our approach which is carried out on MATLAB platform with Intel Core 2 Duo 2.93GHz CPU and 2.96GB RAM, we conduct experiments on several challenging video sequences which include heavy occlusions, abrupt motion and illumination variations. We resize the object image patch to $32\times32$ pixels and extract overlapped $16\times16$ local patches within the object region with 8 pixels as step length. The L1 minimization problem is solved by the SPAMS package [24]. The proposed tracker is compared against six other popular trackers including IVT [6], VTD [4], FragTrack [3], WMIL [12], L1 [15] and Lu [19]. All source codes or binaries are provided by the original authors for fair comparison, and the recommended parameters are set for initialization. Table I lists all the compared video sequences which contain the partial occlusions and illumination variations.

TABLE I.
EVALUATED IMAGE SEQUENCES IN OUR EXPERIMENTS

| Videos Clips | Number of Frame | Challenging Factors |
|---|---|---|
| Faceocc1 [3] | 898 | Partial occlusion |
| Faceocc2 [19] | 819 | Partial occlusion; Plane rotation |
| Woman [3] | 596 | Long-term partial occlusions; Similar appearance clutter |
| Deer [4] | 71 | Abrupt motion; background clutter |
| David [6] | 462 | Illumination and scale variations; plane rotation |
| CAVIAR [25] | 382 | Partial occlusion; Scale change |

### *B. Qualitative Evaluation*

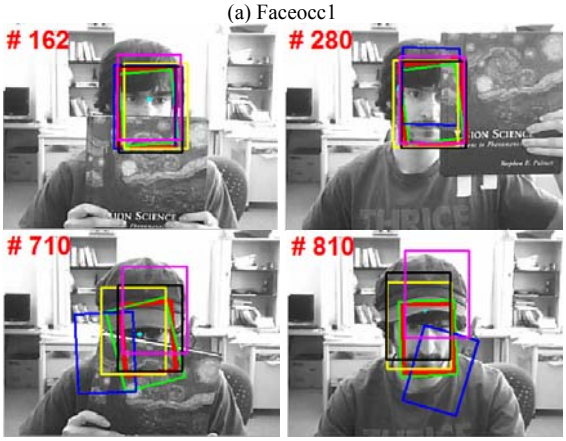
**Occlusions:** Occlusion is one of the most challenging problems during the tracking in Figure3 and Figure4. Several approaches have recently been developed to solve this issue. Faceocc1 sequence, which comes from the authors of [3], tracks face under different circumstances and challenges. In this sequence, most of trackers can keep up with the human face accurately due to never the illumination and motion variations. WMIL and L1 are able to encounter minor drift. The reason is that these trackers don't consider the partial information of object. However, we can see that some approaches for the Faceocc2 sequence are not easy to estimate the in-plane rotation when the object appearance changes much (e.g., when the face turns, the hat is put on the head or the face reappears) in Figure3 (b). FragTrack works poorly; IVT tracker tends to drift the region of object; WMIL cannot

adaptively adjust the object scale due to their design. The remaining trackers can track object through the whole sequence, but our approach is slightly better than Lu tracker.

Figure4 (a) shows the tracking results using CAVIAR sequence. The WMIL and FragTrack are able to adaptively adjust the object scale and perform poorly. The main drawback of WMIL is that Haar-like feature based object representation is less effective due to the similar appearance background clutter. IVT and L1 tracking algorithms exhibit severe drift and cannot handle the serious appearance changes well; while ours and Lu's achieve the best performance in terms of tracking accuracy.

"Walking Lady" sequence is also used to evaluate the performance of each tracker, which contains some long term partial occlusions when woman experiences behind the cars. The color of the woman clothes sometimes is similar to the occluded cars. In Figure4 (b), FragTrack cannot accurately catch up with the object; IVT and L1 lose object because of occlusions and similar background clutter; VTD and WMIL tend to include much of the background area into the bounding box when occlusion occurs; Lu method suffers from the inaccurate tracking result; while our method can perfectly overcome these problems and yield a more stable and accurate result than other trackers.


(a) Faceocc1

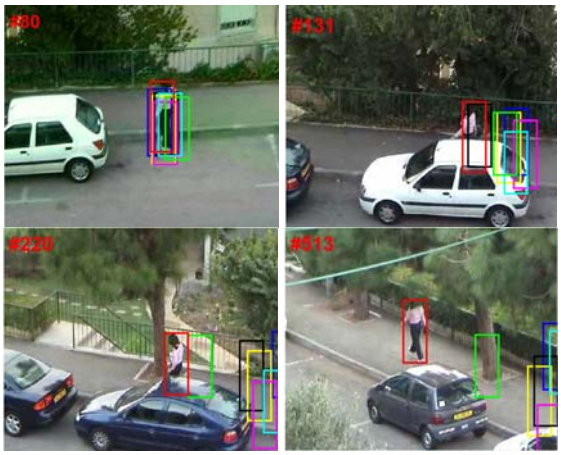



(b) Faceocc2

IVT WMIL FragTrack L1 VTD Lu Ours

Figure3. Object undergoes in-plane rotation and partial occlusions.


(a) CAVIAR



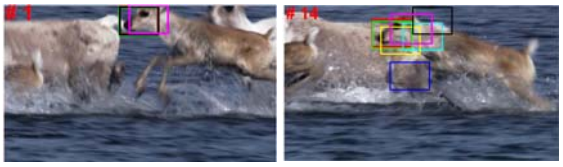

(b) Walking Lady

IVT WMIL FragTrack L1 VTD Lu Ours

Figure4. Tracking results when there is partial occlusion, pose change and similar background clutter.

**Illumination and Fast Motion:** Abrupt motion of the object will lead to blurred image appearance. Figure5 presents several representative frames using the Deer sequence. In this sequence, the object undergoes the drastically appearance changes, which is difficult to location the object. Most of trackers lose the object at the beginning of this sequence. WMIL is not easy to predict the object location. The reason is that most weak learners are not captured easily. Our approach can track the object of interest precisely.

In the David sequence, the appearance of the object may change when a man walks from a dark room into areas with spot light, and several key frames are shown in Figure6. FragTrack and WMIL do not adapt to the variations of scale and in-plane rotation; L1 tracking method drifts away the object gradually at frame 62 and recovers tracking at frame 198. The rest approaches can track the object for this sequence; while ours is slightly better than theirs.



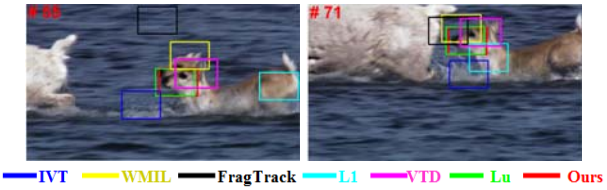



Figure5. Significant object appearance changes due to fast motion on Deer sequence.

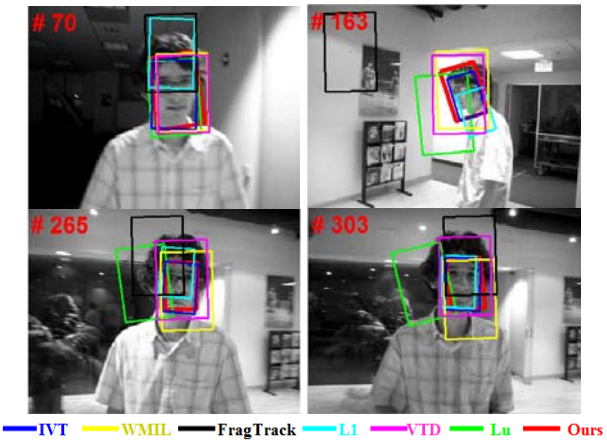



Figure6. Tracking results of different approaches for handling large variation of motion, pose and illumination on David sequence.

### C. Quantitative Evaluation

For all videos, the ground truth information is available in [19]. In this paper, we use two criteria to quantitatively evaluate all trackers performance. The first criterion is the average center location errors which are defined as the Euclidean distance from the tracking result to the ground truth at each frame. The pixel error in every frame is defined as follows:

$$error = \sqrt{(x_t - x_g)^2 + (y_t - y_g)^2} \qquad (15)$$

where $(x_t, y_t)$ represents the tracking position; $(x_g, y_g)$ is the ground truth. The results are summarized in TableII, it is clear that our tracker consistently produces a smaller error than others. In other words, the proposed approach can achieve the best or second best performance in most video sequences.

TABLE II.
CENTER LOCATION ERRORS (IN PIXELS) BOTH OUR TRACKER AND OTHER STATE-OF-THE-ART TRACKERS. THE RED BOLD FONTS INDICATE THE BEST PERFORMANCE, WHILE THE BLUE BOLD FONTS INDICATE THE SECOND BEST

| Videos | IVT | WMIL | Frag-Track | VTD | L1 | Lu | Ours |
|---|---|---|---|---|---|---|---|
| Faceocc1 | 8.8 | 20.9 | 4.6 | 15.1 | 6.5 | **3.7** | **3.5** |
| Faceocc2 | 31.9 | 13.3 | 16.7 | 10.4 | 11.1 | **5.5** | **5.1** |
| Woman | 167.5 | 118.5 | 113.6 | 136.6 | 131.6 | **105.9** | **22.7** |
| Deer | 130.8 | 75.3 | 94.4 | 15.7 | 140.9 | **9.7** | **11.2** |
| David | **3.1** | 8.3 | 105.5 | 13.6 | 7.7 | 33.4 | **6.7** |
| CAVIAR | 45.5 | 17.4 | **5.5** | 60.9 | 119.9 | 5.9 | **4.6** |
| Average | 64.6 | 42.3 | 56.7 | 42.1 | 69.6 | **27.4** | **9.0** |

The other measure is the overlap rate, which is defined by the PASCAL VOC [26].

$$score = \frac{area(R_T \cap R_G)}{area(R_T \cup R_G)} \qquad (16)$$

where $R_T$ and $R_G$ denote the tracking results and the corresponding ground truth, respectively. An object is successfully tracked when the score is above 0.5. TableIII summarizes the average overlap rate. We can see that the proposed approach performs favorably against other popular trackers.

TABLE III.
TRACKING SUCCESS RATES. THE RED BOLD FONTS INDICATE THE BEST PERFORMANCE; THE BLUE BOLD FONTS INDICATE THE SECOND BEST

| Videos | IVT | WMIL | FragTrack | VTD | L1 | Lu | Ours |
|---|---|---|---|---|---|---|---|
| Faceocc1 | 0.86 | 0.67 | 0.83 | 0.77 | 0.88 | **0.91** | **0.93** |
| Faceocc2 | 0.39 | 0.62 | 0.60 | 0.59 | 0.67 | **0.78** | **0.80** |
| Woman | 0.19 | 0.19 | **0.20** | 0.15 | 0.18 | 0.11 | **0.60** |
| Deer | 0.09 | 0.14 | 0.08 | 0.51 | 0.07 | **0.60** | **0.62** |
| David | **0.74** | 0.50 | 0.08 | 0.53 | 0.63 | 0.31 | **0.76** |
| CAVIAR | 0.27 | 0.52 | 0.69 | 0.20 | 0.28 | **0.71** | **0.74** |
| Average | 0.42 | 0.44 | 0.41 | 0.46 | 0.45 | **0.57** | **0.74** |

### D. Computational Cost

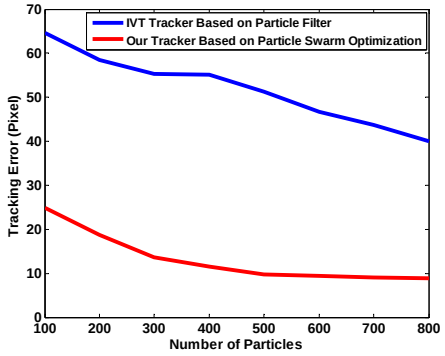
In this subsection, we will evaluate the computational cost of both the proposed tracker and particle filter based tracker. We carry out both trackers with different number of particles and the runtime cost increases dramatically with the number of particles in Figure7. It is clear that the performance of tracking for the particle filter based tracker depends more heavily on the number of particles. But the number of particles in our tracker does not have a significant impact on the tracking errors when it reaches a certain amount. The reason is that our approach is based on PSO iteration which is a multi-layer sampling process. The particles are moved forwards the region where the fitness value of observation is the most similar to the object template via the PSO iterations.

The computational complexity of each frame is dominated via PSO iterations. The number of iterations for different frames is of difference. Figue7 (a) shows the tracking accuracy of both trackers with different number of particles. We can see that our approach is able to achieve the better performance with about 400 particles than IVT method with the same number of particles and takes about average 2.6s for each frame with a tracking error of 11.6.

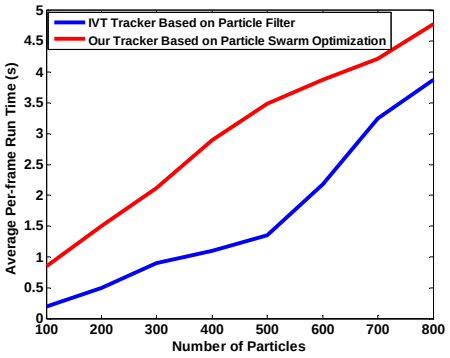
## VI. CONCLUSION

In this paper, we propose an effective PSO based tracking method which can solve some challenges (such as appearance occlusions and abrupt motion variations) in visual tracking applications using spatial structure information and SIFT feature points correspondence. Structural information of the object and SIFT feature matching scheme can handle partial occlusions and reacquire the lost object from the scene, respectively. In addition, an online dictionary learning strategy is used to account for the appearance variations. The dictionary is further segmented into *N* sub-matrices corresponding to *N* spatial patches of an object state. The atoms from the first frame are never changed during the tracking to alleviate the drifting problem, and other atoms in one sub-matrix are selectively updated based on the corresponding coefficients of object templates. The experimental results

show that our approach can cope with the problems of large motion changes and partial occlusions.




(a) Tracking errors with different number of particles




(b) Average runtime for each frame in all sequences

Figure7. Comparison of both IVT tracker and our tracker with the different number of particles in all test sequences.

## ACKNOWLEDGMENT

The authors would like to thank the anonymous reviewers for useful and constructive comments that help improve the quality of this paper. This work is supported by National Nature Science Foundation of China (NSFC) under Grant (No. 60971098, 61302152, 61201345) and the Beijing Key Laboratory of Advanced Information Science and Network Technology (No. XDXX1308).

## REFERENCES

[1] J. Cao, W. Li, D. Wu, "Multi-feature fusion tracking based on a new particle filter," Journal of Computers, vol.7, no.12, pp. 2939-2947, 2012.

[2] J. Chen, S. Zhang, G. An, et al, "A generalized mean shift tracking algorithm," Science China Information Sciences, vol. 54, no. 11, pp. 2373–2385, 2011.

[3] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," IEEE Conference on Computer Vision and Pattern Recognition, pp. 798-805, 2006.

[4] J. Kwon and K. M. Lee, "Visual tracking decomposition," IEEE Conference on Computer Vision and Pattern Recognition, pp. 1269-1276, 2010.

[5] K. Yu, G. Xu, and X. Shen, "Position location and trajectory tracking for MUWT based kinematics approach," Journal of Computers, vol. 8, no. 4, pp. 1020-1026, 2013.

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

[7] M. Thida, H. Eng, D.N. Monekosso, P. Remagnino, "A particle swarm optimization algorithm with interactive swarms for tracking," Applied Soft Computing, vol. 13, no. 6, pp. 3106–3117, 2013.

[8] X. Cheng, N. Li, S. Zhang, Z. Wu, "Robust visual tracking with SIFT features and fragments based on particle swarm optimization," Circuits, Systems, and Signal Processing, pp. 1-20, 2013, in press.

[9] S. Avidan, "Ensemble tracking," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 29, no. 2, pp. 261-271, 2007.

[10] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," British Machine Vision Conference, pp. 47-56, 2006.

[11] B. Babenko, M. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 33, no. 8, pp. 1619-1632, 2010.

[12] K. Zhang, and H. Song, "Real-time visual tracking via online weighted multiple instance learning," Pattern Recognition, vol. 46, no. 1, pp. 397-411, 2013.

[13] Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: Bootstrapping binary classifiers by structural constraints," IEEE Conference on Computer Vision and Pattern Recognition, pp. 49-56, 2010.

[14] X. Mei, and H. Ling, "Robust visual tracking using L1 minimization," IEEE International Conference on Computer Vision, pp. 1436–1443, 2009.

[15] C. Bao, Y. Wu, H. Ling and H. Ji, "Real time robust L1 tracker using accelerated proximal gradient approach," IEEE Conference on Computer Vision and Pattern Recognition, pp. 1830-1837, 2012.

[16] B. Liu, J. Huang, L. Yang, and C. Kulikowsk, "Robust tracking using local sparse appearance model and k-selection," IEEE Conference on Computer Vision and Pattern Recognition, pp. 1313-1320, 2011.

[17] X. Zhang, W. Li, "Block covariance based l1 tracker with a subtle template dictionary," Pattern Recognition, vol. 46, no. 7, pp. 1750-1761, 2013.

[18] F. Chen, Q. Wang, S. Wang, W. Zhang, and W. Xu, "Object tracking via appearance modeling and sparse representation," Image and Vision Computing, vol. 29, pp. 787-796, 2011.

[19] W. Zhong, H. Lu, and M. Yang, "Robust object tracking via sparsity-based collaborative model," IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8, 2012.

[20] T. Zhang, B. Ghanem, S. Liu, et al, "Robust visual tracking via structured multi-task sparse learning," International Journal of Computer Vision, vol. 101, no. 2, pp. 367-383, 2013.

[21] K. James, R. Eberhart, "Particle swarm optimization," in Proceedings of IEEE International Conference on Neural Networks, pp. 1942–1948, 1995.

[22] D. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, vol. 60, no. 2, pp. 91–110, 2004.

[23] M.A. Fischer, R.C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," Communications of the ACM, vol. 24, no. 6, pp. 381–395, 1981.

[24] J. Mairal, F. Bach, J. Ponce and G. Sapiro, "Online learning for matrix factorization and sparse coding," Journal of Machine Learning Research, vol. 11, pp. 19-60, 2010.

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

[26] M. Everingham, L. V. Gool, C. Williams, et al, "The pascal visual object classes (voc) challenge," International Journal of Computer Vision, vol. 88, no. 2, pp. 303-338, 2010.
[27] Z. Feng, B. Yang, Y. Zheng, et al, "Initialization of 3D Human Hand Model and Its Applications in Human Hand Tracking," Journal of Computers, vol.7, no.2, pp. 419-426, 2012.

**Xu Cheng** is currently a Ph.D. student at Southeast University, Nanjing, China. He received his B.S. and M.S. degree in information engineering from Taiyuan University of Technology, in 2007 and 2010, respectively. His research interests include image/video processing, computer vision and pattern recognition.

**Nijun Li** received his B.S. degree from the Southeast University in 2010. He is currently a Ph.D. student at Southeast University. His research interests include action recognition and object classification.

**Tongchi Zhou** received his M.S. degree from the Xinjiang University in 2011. He is currently a Ph.D. student at Southeast University. His research interests include action recognition and object classification.

**Lin Zhou** received the Ph.D. degree in signal and information processing from Southeast University, Nanjing, China in 2005. She is currently an Associate Professor of Signal and Information Processing. Her research interests include speech processing and spatial hearing.

**Zhenyang Wu** received the M.S. degree in electrical engineering from Southeast University, Nanjing, China in 1982. During 2000–2008, he was the vice dean of the School of Information Science and Engineering, Southeast University. He is currently a Professor of Signal and Information Processing. His research interests include speech and audio processing, image and multimedia processing, and sensor array signal processing. He has published more than 100 international journal and conference paper. From 1990s, he has been an invited reviewer of several famous scientific journals. He is a number of IEEE. Prof. Wu served as a member of technical program committee for the 13th IEEE International Symposium on Consumer Electronics (ISCE2009). In 2008, he received the National Award for Distinguished Teacher.