Received August 28, 2018, accepted September 21, 2018, date of publication September 28, 2018, date of current version December 27, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2872524

An Experimental Comparison of Swarm Optimization Based Abrupt Motion Tracking Methods

HUANLONG ZHANG[1], XIUJIAO ZHANG[1], YAN WANG[1], KUNFENG SHI[1], JIANWEI ZHANG[2], AND CHAO LI[3]

[1]College of Electric and Information Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002 , China
[2]Software Engineering College, Zhengzhou University of Light Industry, Zhengzhou 450002, China
[3]Shandong SNTON Optical Material Technology Co., Ltd., Dongying 257500, China

Corresponding author: Jianwei Zhang (ing@zzuli.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grants 61873246, 61503173, 61672471, and 61603347, and in part by the Doctor Fund Project of the Zhengzhou University of Light Industry under Grant 2016BSJJ002.

ABSTRACT In view of the problem that the traditional tracker does not adapt to the large displacement or abrupt motion well, the optimization method attracts more and more attention for a robust tracker. Considering population based has high feasibility for avoiding local optimal, the method of swarm optimization is introduced to visual tracking. These methods convert visual tracking to search the optimal solution in global. To show their merits, this paper reviews and evaluates three relatively classical swarm optimization-based tracking algorithms. These algorithms are ant lion optimization, cuckoo search, and particle swarm optimization. Their running results are compared with those of the probabilistic optimization algorithm, namely, simulated annealing. The experiments demonstrate the strength as well as the weakness of these methods. For illustrating their operational efficiency, run time is recorded and the convergence speed is analyzed. In addition, quantitative and qualitative analysis experiments are performed to interpret the accuracy of the tracking methods. In addition, the relation between parameters and tracking results is explained.

INDEX TERMS Experimental comparison, swarm optimization, abrupt motion, visual tracking.

I. INTRODUCTION

Visual tracking is a task that continuously locates the state of a specific target from an image sequence. At present, visual tracking plays a crucial role in many applications, especially for aerospace, autonomous navigation, medical diagnosis, surveillance, video compression, intelligent robot and human computer interaction by using technologies of artificial intelligence, pattern recognition, automatic control, and so on.

Over the past decades, although researchers have made significant progress, there still existing many challenging problems, such as motion uncertainty (inconsistent speed, camera switching, low frame-rate videos). On this occasion, trackers tend to deviate from the actual position even though the appearance model is flawless. This is may due to the inefficient samples extracted from the incorrect state space. Some methods have been proposed to overcome above problems. It includes detection based tracking methods [1]–[4], optimization based tracking method [5]–[8], motion model based tracking methods [9]–12] and random sampling based tracking methods [13]–[16], and so on.

In these methods, optimization based tracking methods have attracted more and more attention because of their superior global search performance. Many optimization methods have been applied to visual tracking for improving tracking performance particularly in the scene of motion uncertainty. Minami *et al*. [17] use Genetic Algorithm (GA) in a method called 1-step-GA for representing the error of the target in the images. Meanwhile, the target fish obtains a better identification and tracking result with the help of the PD-type controller. Zhang *et al*. [6] incorporate the temporal continuity information into PSO, multilayer sampling method based particle filter is formed. The tracker gets better performance especially when the target undergoes an arbitrary motion or obvious appearance changes. Chen *et al*. [18] propose Hybrid Quantum Particle Swarm Optimization (HQPSO) method, which overcomes the problem that the population diversity get easily lost during the latter period of evolution in PSO. Hao *et al*. [19] propose a particle filtering algorithm based on Ant Colony Optimization (ACO) to boost the property of particle filter with small sample set, which improves the tracking efficiency. Nguyen and Bhanu [20] present a modified Bacterial Foraging Optimization (BFO) algorithm, and designed a pedestrian tracking system based on the BFO to handle the some challenges. Gao *et al*. [21] propose a tracker is based on Firefly Algorithm (FA), which can robustly track an arbitrary target in various challenging conditions. Ljouad *et al*. [22] propose an extension version of CS algorithm combined with the well-known Kalman Filter, and design a visual tracking system based on the Hybrid Kalman-Cuckoo Search (HKCS) algorithm. The new tracker achieves very good tracking performances. In addition, Gao *et al*. [23], [24] present bat algorithm and flower pollination algorithm to solve tracking problems.

Although the above researches have made great progress, it still remains some problems: more iterative times, time-consuming and the optimal resolution relying on parameters setting greatly. Based on this, it is very important to review and evaluate the optimization based tracking methods in recent years for designing a new method with the specific tracking scenarios. However, as far as we know, there is scarcely researcher to perform the work.

In this paper, we described the tracking performance of ALO, CS and PSO. For showing their effectiveness, the tracking results are compared against those of SA. The experiments about tracking accuracy and efficiency are carried out. In order to account for the running speed of the algorithm, we record the running time and the convergence speed. For explaining the convergence speed, we performed the tracking on one frame using different number of iterations and population sizes. To verify the accuracy of tracking methods, we perform the quantitative and qualitative analysis. The result of quantitative analysis refers to average overlap rate and average error center rate. The result of qualitative analysis involves description and visualization in images. The influence of parameters to the speed and accuracy of the tracking algorithm is analyzed.

II. VISUAL TRACKING AND OPTIMIZATION

A. VISUAL TRACKING

Visual tracking problem is equivalent to the corresponding matching problem of the target position, speed, shape, texture, color and other related features [25]–[27] in the continuous image frames. Visual tracking consists of target initialization, appearance model, motion estimation and target location, as shown in Fig. 1.

1) THE INITIALIZATION OF TARGET STATE

The initialization of target is usually implemented by manual calibration or target detection. Manual calibration is the target that the user marks on the first frame using a rectangle or ellipse. Target detection generally adopts target detection algorithm to get initial target.

2) THE APPEARANCE MODEL

Appearance model generally consists of two components: feature representation and statistical learning.

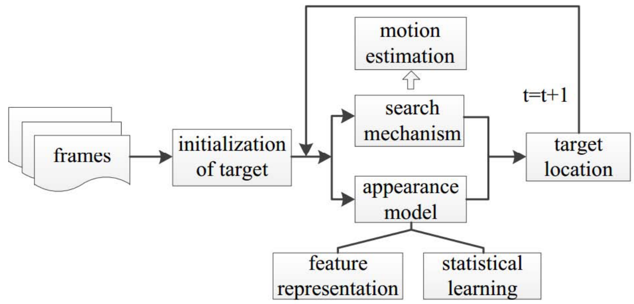


FIGURE 1. Flow chart of object tracking.

Feature representation focuses on how to construct robust object descriptors using different features. Statistical learning concentrates on how to build effective mathematical models for object identification using different machine learning methods.

Feature representation plays a vital role in the tracking process, which is the core challenge in object tracking. An object can be represented by either global representation or local representation. The global representation reflects the global statistical characteristics of object appearance, such as the covariance and histogram representation. In general, global representations are prominent in the tracking process of abrupt motion or fast motion. It's simple and computationally efficient in that case. However, the global representations are susceptible to the changes of global appearance because of the imposed global geometric constraints. Local feature-based representations describe the object information by detecting the interest points or saliency. The local representations can acquire the local texture of the target. Nevertheless, they are often puzzled by background clutter and noise disturbance.

Those feature vectors are then employed either in the generative method [28]–[30] or in the discriminative method [31]–[33] to detect and locate the target in the image sequences. In practice, powerful appearance models rely not only on effective feature representation but also robust statistical learning. Occlusions, cluttered backgrounds, illumination changes and other challenges are still existence in real world. Therefore, it is more beneficial to exploit target-specific representations through a learning process rather than using a fixed set of pre-defined features. Deep learning [34], [35], particle filter [36], [37], boosting [38] and support vector machine (SVM) [39] are learning methods commonly.

3) MOTION ESTIMATION

Motion estimation is formulated as a dynamic state estimation problem: $x_t = f(x_{t-1}, v_{t-1})$ and $z_t = h(x_t, w_t)$, where f is designed to evolution function, x_t refers to current state, v_{t-1} denotes the process noise, z_t involves to the current observation, h is the measurement function, and w_t is the measurement noise. The motion model is used to determine the search space which affects the tracking results in a large part. For example, when the target undergoes abrupt motion and may jump out of the prediction space, the tracker will fail.

4) SEARCH MECHANISM

Tracking methods can be divided into two categories according to their search mechanisms, they are deterministic and stochastic. Deterministic methods [40]–[42] track target by iteratively searching the candidate space and maximizing the fitness function between candidate patches and the target in each frame. Probabilistic methods [43]–[45] regard the tracking process as a state solving problem under the Bayesian framework, modeling uncertainty and propagating the conditional densities through the tracking process.

5) TARGET LOCATION

Finally, based on the apparent modeling and motion estimation, some strategies are used to obtain the position of the target.

B. RELATIONSHIP BETWEEN OPTIMIZATION AND VISUAL TRACKING

Optimization is the process of searching best solution from all available candidate solutions. Most of the optimization problems with explicit targets can be presented as nonlinearly constrained as following generic form

$$Target = optimize\, f\,(x)\,, x \in D \tag{1}$$

where D is the search space, f represents fitness function or objective function. Each feasible solution x consists of optimization variables $x = [x_1, x_2, \ldots, x_n]$.

In the tracking, the result of deterministic methods can be obtained by minimizing or maximizing an objective function $f(x)$ based on distance, similarity or classification measure. For stochastic methods, some stochastic factors are introduced into the searching process to create a higher probability of reaching the global optimal solution in the search space D.

III. ALGORITHM REVISIT

A. ANT LION OPTIMIZATION

The ALO algorithm imitates the hunting behavior of ant lions in nature [46]. Ants are search agents which wander over the search space and ant lions build pits to trap and consume the ants. The fitter function is optimized to reflect the hunting ability of ant lion. The mainly elements of ALO algorithm can be explained as below:

- *Random* walks of ants: In nature, ants move randomly when they are searching for food. The random movement of ants can be modeled as follows:

$$X(t) = [0, cumsum(2r(t_1) - 1), cumsum(2r(t_2) - 1), \ldots, cumsum(2r(t_T) - 1)] \tag{2}$$

where *cumsum* calculates the cumulative sum, T is the maximum number of iteration, t shows the step of random walk (iteration in this study), and $r(t)$ is a stochastic function. In order to limit the updating position of ants inside the boundary, the values are normalized using the following equation:

$$X_i^t = \frac{\left(X_i^t - a_i\right) \times \left(d_i^t - c_i^t\right)}{(b_i - a_i)} + c_i^t \tag{3}$$

Where a_i is the minimum of random walk of $i-th$ variable, b_i is the maximum of random walk in $i-th$ variable, c_i^t is the minimum of $i-th$ variable at $t-th$ iteration, and d_i^t indicates the maximum of $i-th$ variable at $t-th$ iteration.

- *Trapping* in antlions' traps: The effect of ant lions on the random walk of ants are shown in the antlions' trap.

$$c_i^t = Antlion_j^t + c^t \tag{4}$$
$$d_i^t = Antlion_j^t + d^t \tag{5}$$

where c_i^t and d_i^t have been defined earlier, c^t is the minimum of all variables at $t-th$ iteration, and d^t indicates the maximum of all variables at $t-th$ iteration, and $Antlion_j^t$ shows the position of the selected $j-th$ antlion at $t-th$ literation

- *Building* traps: In order to model the antlions' hunting capability, a roulette wheel is employed. The ALO algorithm is required to utilize a roulette wheel operator for selecting antlions based on their fitness during optimization. Antlions shoot sands outwards the center of the pit once they realize that an ant is in the trap. The mathematical model describes the way how the trapped ant slides down towards antlion is given as follows:

$$c^t = \frac{c^t}{I} \tag{6}$$
$$d^t = \frac{d^t}{I} \tag{7}$$

where $I = 10^w \frac{t}{T}$ is a ratio. T is the maximum number of iterations, and w is a constant and the value be computed by $t > 0.1T$, w=2; $t > 0.5T$, w=3; $t > 0.75T$, w=4; $t > 0.9T$, w=5; $t > 0.95T$, w=6.

- *Catching* preys and rebuilding the traps: Catching preys occurred when ants become fitter than its corresponding predator. Then, antlion will update its latest location of the hunted ant to improve its opportunity of catching new prey, which this mechanism can be modeled as below:

$$Antlion_j^t = Ant_i^t \;\; if\; f\left(Ant_i^t\right) > f\left(Antlion_j^t\right) \tag{8}$$

where Ant_i^t indicates the position of $i-th$ ant at $t-th$ iteration.

- *Elitism* and rebuilding pits: the best antlion in each iteration is considered to be elite. It means that every ant randomly walks around selected antlion by the roulette wheel and the elite simultaneously as follows:

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \tag{9}$$

whereR_A^t is the random walk around the antlion selected by the roulette wheel at $t-th$ iteration, R_E^t is the random walk around the elite antlion $t-th$ iteration.

B. CUCKOO SEARCH

The basic idea of cuckoo search algorithm is based on the obligate brood parasitic behavior of cuckoo specie combination with the *Lévy* flight behavior of fruit flies. The cuckoos aggressively reproduce and lay their eggs in the nests of other host bird species. Some host birds throw away the alien eggs or build new nests elsewhere upon discovering cuckoo's eggs. In addition, some works have testified that the flight behaviors of fruit flies have the typical characteristics of the *Lévy* flights. Considering these propagation and flight behaviors, a CS algorithm proposed by Yang and Deb [47]. It can be described by using three idealized rules as following:

1) A bird chooses a nest randomly then lays one egg at a time.

2) The highest quality of nest will be reserved to the next generation.

3) The number of host bird nests is fixed. The probability of discovering the cuckoo's eggs by the host bird is set at a constant value $pa \in [0,\ 1]$.

The mainly elements of CS algorithm can be explained as below:

- *Lévy* flight: The mathematical representation of cuckoo random walk is performed using *Lévy* flight, the *Lévy* flight based walk is given in

$$X_i^{t+1} = X_i^t + \alpha \oplus L(\lambda) \tag{10}$$

  Where t represents the current iteration, $\alpha > 0$ shows the step size. The product $\oplus$ is the entrywise multiplications. The *Lévy* flight random walks based on *Lévy* distribution, as given in

$$Levy(\lambda) = t^{-\lambda}, (1 < \lambda \leq 3) \tag{11}$$

- *The* discovering probability *Pa*: *Pa* provides the percentage of nest rebuilding required. The *Pa* value is used to deserted the worst nests and replace with new nests. If $rand < Pa$, keep the original nest. Otherwise, host birds build a completely new nest using Eq.(12)

$$nest = x_{\min} + (x_{\max} - x_{\min})\, rand \tag{12}$$

  where x_{min} and x_{max} are the lower and upper boundary values. *rand* is random number.

C. PARTICLE SWARM OPTIMIZATION

PSO is inspired by the social behavior of bird flocking which use of group collaboration and information sharing to find the optimal location of food [48]. In PSO algorithm, each of bird represents a particle.Each particle is updated by the individual optimal *pbest* and the global optimal *gbest*. In every iteration, individual optimal *pbest* is the best solution that individual has achieved so far, and the global optimal *gbest* is the best solution obtained so far by all particles.

The mainly elements of PSO algorithm can be explained as below:

- *Updated* speed: In D dimensional space, suppose the population size is N, and the flight speed of each particle can be expressed as $v_i = (v_{i1}, v_{i2}, \ldots, v_{id})^T$, the position of the particle is expressed as $x_i = (x_{i1}, x_{i2}, \ldots, x_{id})^T$, the *i-th* particle can update speed and position according to formula (13):

$$v_{id}^{t+1} = w \bullet v_{id}^t + c_1 \bullet r_1 \bullet \left(pbest_{id}^t - x_{id}^t\right) + c_2 \bullet r_2 \bullet \left(gbest_{gd}^t - x_{id}^t\right) \tag{13}$$

  Where $i = 1, 2, \ldots, N$; $d = 1, 2, \ldots, D$. w is called the inertia weight which range is [0.4, 0.9], and the positive constants c_1 and c_2 are, respectively, cognitive and social parameters. r_1 and r_2 are the random between [0,1]. v_{id}^t indicates the flying speed of *i-th* particle at $t - th$ iteration. v_{id}^{t+1} indicates the flying speed of *i-th* particle at $(t+1) - th$ iteration. x_{id}^t indicates the position of *i-th* particle at $t - th$ iteration.

- *Updated* position: The position of the particle is updated as follows:

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \tag{14}$$

  Where x_{id}^{t+1} indicates the position of *i-th* particle at $(t + 1) - th$ iteration.

D. SIMULATED ANNEALING

The simulated annealing algorithm is an annealing process [49] that simulates solid matter which requires an initial temperature T_0, final temperature T_m and cooling rate α. At each temperature, some disturbances are performed to generate new sampling points and the energy of new sampling points will be measured. Then, the energy of the new sampling point is compared with the current energy. If the energy value $f\left(x'\right)$ of the new sampling point is higher than the energy value $f(x)$ of the current sampling point then it will be accepted as current sampling point. If the energy value $f\left(x'\right)$ of the new sampling point is inferior to the energy value $f(x)$ of the current sampling point then it may still be accepted based on the acceptance probability Pr.

The mainly elements of SA algorithm can be explained as below:

- *The* acceptance probability: the acceptance probability Pr is,

$$pr = \exp\left[\frac{f(x) - f\left(x'\right)}{T(x)}\right] \tag{15}$$

  Where $T(x)$ is the temperature at sampling point x. If $rand < pr$, the new sampling point will be accepted.

- *Cooling* rate: The SA algorithm requires a static or adaptive cooling schedule. A common geometric cooling schedule is used,

$$T\left(x'\right) = \alpha T(x) \tag{16}$$

  Where, $T\left(x'\right)$ is the new temperature.

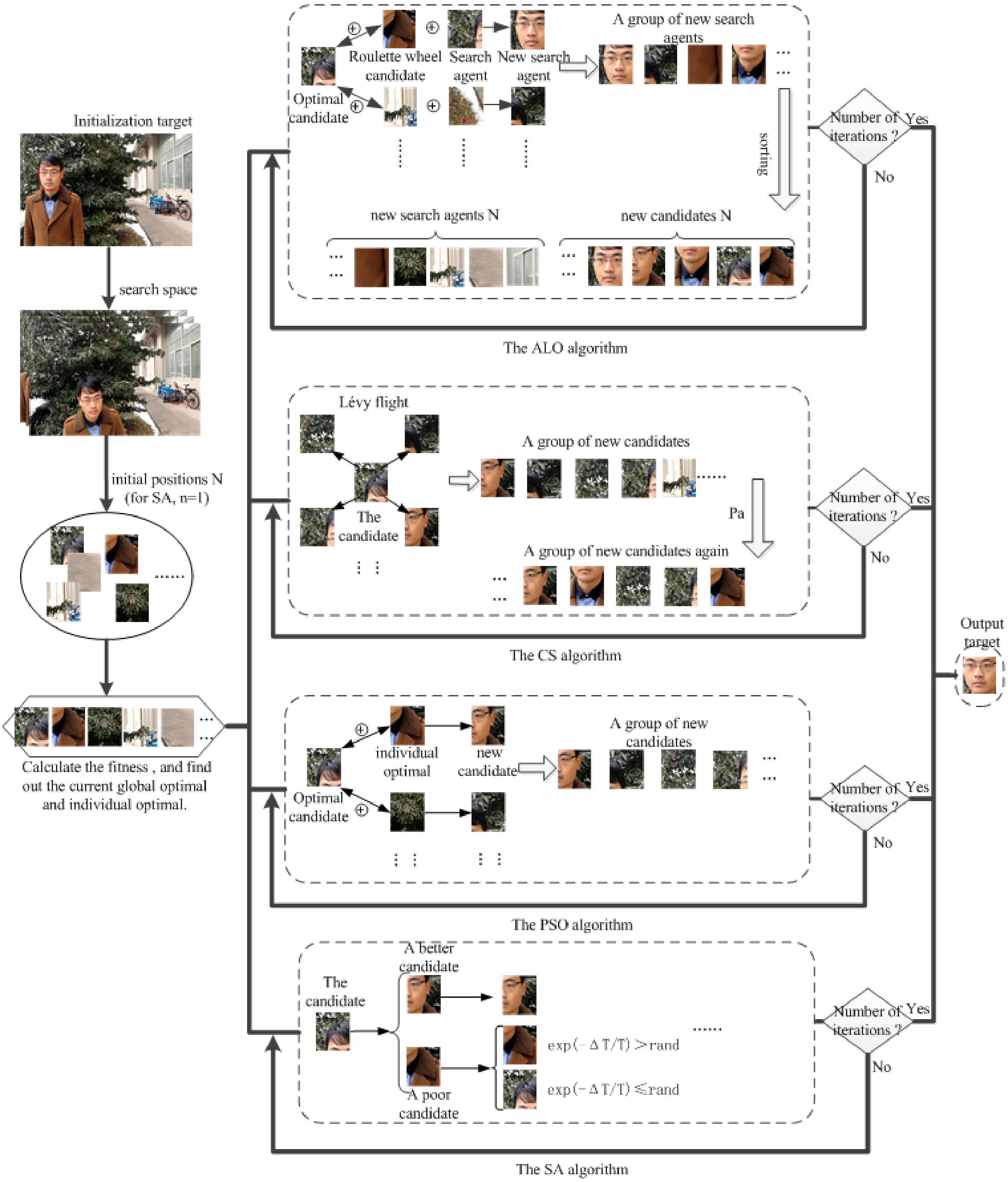


FIGURE 2. Flow chart of optimization based tracker.

IV. VISUAL TRACKING BASED ON OPTIMIZATION ALGORITHM

In the tracking framework, we adopt the way of manual calibration to initialize the target state. Histogram of Oriented Gradient (HOG) feature is extracted for feature description, and the correlation coefficient is used to measure the similarity. The ALO algorithm, CS algorithm, PSO algorithm and SA algorithm are used as the search strategy respectively. The tracking flowchart is shown in Fig. 2. In the tracking, the four algorithms can be described as follows:

- *Suppose* a ground-truth in the image (state space), which corresponding to the target (best elite antlion) and a set of target candidate images (antlions and ants) are randomly generated. In a sense, the ants are actual search agents that explore and move in the searching space, whereas antlions can save the best position of ants that are obtained so far. Meanwhile, the antlions guide the next searching of ants. This interaction mechanism allows ants to update their location to find better solutions and elite antlion to maintain their current best position. The ALO-based tracker (ALOT) can be described as a global optimization process for finding the best elite antlion in all antlions and ants.

- *As* with ALO-based tracking process, a hypothesis that ground-truth exists in the image corresponding to the target is proposed. The target refers to best nest. And a set of target candidates nests are randomly generated. The purpose of the CS-based tracker (CST) is to search the best nest in the all candidate nests by using the CS algorithm. Based on the purport, a CS-based tracking architecture is designed.
- *A* same hypothesis is proposed in the PSO-based tracking process. The target corresponds to the food. And a number of birds are randomly generated as the candidates to search the food. The aim of the PSO-based tracker is to find food using the PSO algorithm. The framework of PSO-based tracker (PSOT) is designed for robust and efficient visual tracking.
- *In* the SA-based tracking process, the target corresponding to the point of minimum energy will be located. A sampling point is randomly generated as candidate at initial temperature. The algorithm performs several perturbations in the sampling point to generate new sampling point and the temperature gradually decreases simultaneously. Iterate through the execution until the internal energy is minimized. The SA-based tracker (SAT) is also shown in fig2.

V. TESTING AND EVALUATION

To evaluate different trackers' performance, we test four optimal algorithms on 8 challenging image sequences. For BOY, COUPON, DEER, HUMAN7 and SHAKING sequences, they are available on the website http://visualtracking.net. ZXJ, FACE2 and FHC are our own. They have 118, 310 and 123 frames, respectively, and their sizes are 568*320, 640*480, 1920*1080. These sequences main challenge large displacement motion or abrupt motion. Trackers are implemented in MATLAB R2014a. The experiments are conducted on a PC with Intel Core i7 2.50GHz and 16GB RAM.

The initial parameters for the optimization algorithms were set as follows:

1) ALOT: The number of ants and antlions n is set at 250 respectively. The number of iteration T is set at 100. $t > 0.5T$, $W = 1$; $t > 0.7T$, $W = 2$; $t > 0.9T$, $W = 2.7$ (t is the current number of iteration).

2) CST: The number of host nests n is set at 250. The number of iteration is set at 100. The discovery probability pa is set at 0.5. The step size α is set at 0.5.

3) PSOT: The number of particles n is set at 250. The number of iteration is set at 100. Individual factor $c1 = 2.5$, social factor $c2 = 0.5$, the inertia weight w is set at 0.9-0.4;

4) SAT: Initial temperature T_0 is set at 10, final temperature T_m is set at 0.1^{10}. Cooling rate α is set at 0.9.

To compare the optimization algorithms fairly, each algorithm is set to the same population size and the number of iterations (for ALOT, CST, PSOT). These parameters ensure that all methods perform 25,000 evaluations respectively, which is enough to compare the performances of the optimization algorithm for the tracking data set. The focus of this study is to compare algorithms' mechanisms rather than tracking performance, so other parameters will not be adopted any more. In this study, each tracker is run only one time randomly.

A. ANALYSIS OF EFFICIENCY

1) THE AVERAGE EXECUTION TIME

Fig3 gives the statistics values of the average execution time in seconds (s) of each optimization algorithm.

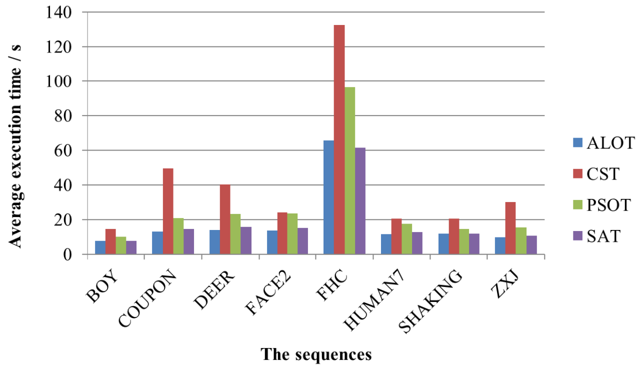


FIGURE 3. Average execution times.

As shown in Fig. 3, it can be observed that CST takes the longest time. The reason may result from the update mechanism of discovery probability. Some bird nests are replaced by new randomly generated nests according to the value of *Pa*. This operation actually increases the number of evaluations. If the value of *Pa* is set greater than 0.5, the smaller possibility of cuckoo's eggs are discovered by the host bird. On this case, the less number of the bird nests are be updated randomly so as that the running time would be reduced. However, this may result in the local minimum problem in tracking. PSOT takes more time than ALOT and SAT since a higher computational complexity. The average execution time of ALOT and SAT are comparable.

2) CONVERGENCE SPEED

Fig. 4 shows the relation between the convergence accuracy and the parameters setup, including the iteration numbers and the population sizes. Here, for SAT the number of iterations is set firstly, then the temperature cooling rate α is adjusted to the same execution times as other algorithms ($\alpha = 0.3, 0.52, 0.65, 0.73, 0.78$). In Fig. 4, the X axis represents the number of iteration, the Y axis represents the population size, and the Z axis represents the similarity value.

In this paper, as shown in Fig. 4, it can be seen that ALOT shows the fastest convergence speed in tracking. Although the

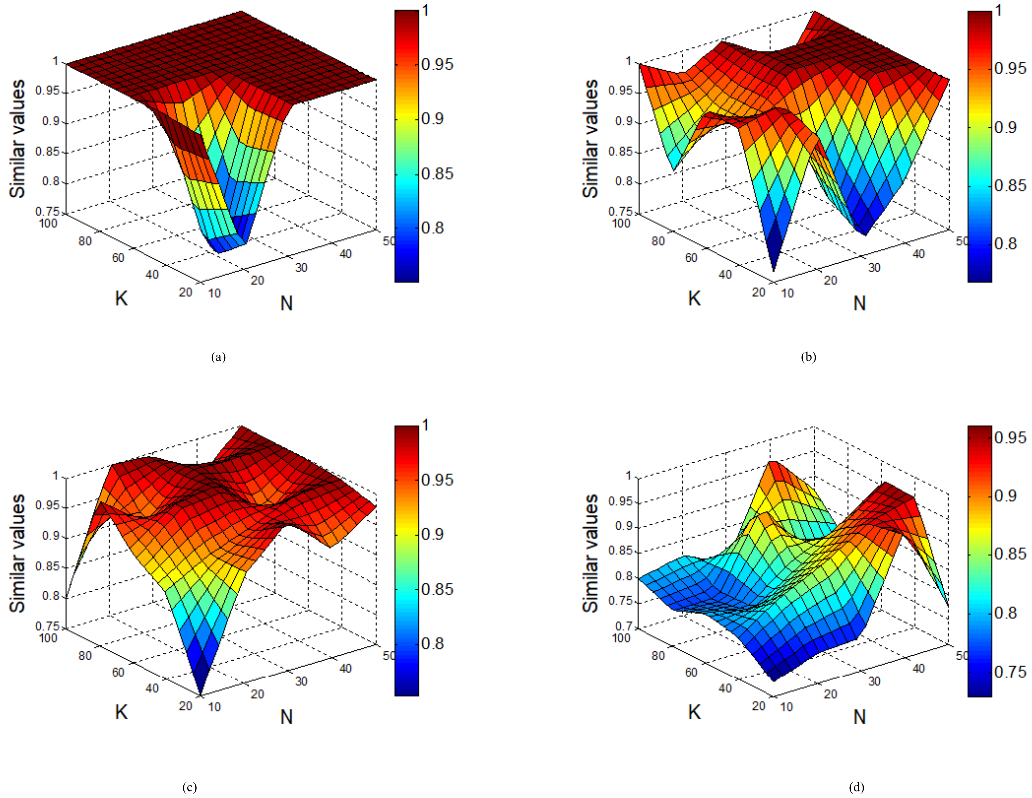


FIGURE 4. Similar values against population size N and number of iterations K . (a) The ALOT. (b) The CST. (c) The PSOT. (d) The SAT.

worst fitness emerges when the population size N is set to 20 and the number of iterations K is set to 40, it can converge to the global optimal for all cases when the population size N is set to 30. The ALOT has an excellent performance that profits from several reasons: 1) The adaptive shrinkage mechanism of random walk space promotes local exploitation with the increasing number of iterations. High exploitation assists the ALO algorithm to rapidly converge towards the target. 2) The best antlion is saved and considered as the elite in each iteration, and the elite guides all ants toward promising regions of the search space.

The CST is far behind the ALOT. Although it has already converged to the optimal occasionally when the population size N is set to 30 and the number of iteration K is set to 60, the CST achieves the tracking successfully for all cases only when the population size is up to 50. One of the main reasons is that the mechanism of update nest is based on discovery probability. Some bird nests are generated randomly, which may deviate from the best nest. The mechanism facilitates exploration in space, but it also results in slow convergence speed.

The PSOT has strong local search capability. In most cases, PSOT can converge to the optimal solution. But its well-known stability problem limits the success rate. As can be seen from Fig. 4 (c), although the tracker firstly converges to the optimal solution when $N = 30$ and $K = 20$, it can only succeed for a few times.

The SAT has the worst convergence, and it fails completely. When $N = 60$ and $K = 50$, the tracker achieves the best result that the similarity value is 0.9616. But it is still a failure. Lacking of local exploitation and optimal solution guide was blamed for all of these.

For ALOT, the greater value of W can lead to the smaller range of ants' random walk, which can achieve the faster convergence speed. For CST, the greater value of Pa, the less quantity of bird nest updated stochastically, the faster convergence speed. For PSOT, the greater learning factor values of $c1$ and $c2$, the particle speeds up to the optimal value, the faster convergence speed. However, for all algorithms, a faster convergence speed is accompanied with the danger of trapping to local optimal.

B. ANALYSIS OF THE ACCURACY

1) QUANTITATIVE ANALYSIS

Table.1 and Table.2 list a per-sequence comparison of ALOT, CST, PSOT and SAT, while Table.1 refers to average overlap rate and Table.2 is concerned with average error center rate. For average overlap rate, the larger the value, the higher the accuracy; for the central error rate, the smaller the value, the higher the accuracy. Table.1 and Table.2 show some information: 1) In those sequences that all trackers are successful, the ALOT has a better performance than others. 2) When viewed from all the sequences, the CST is superior.

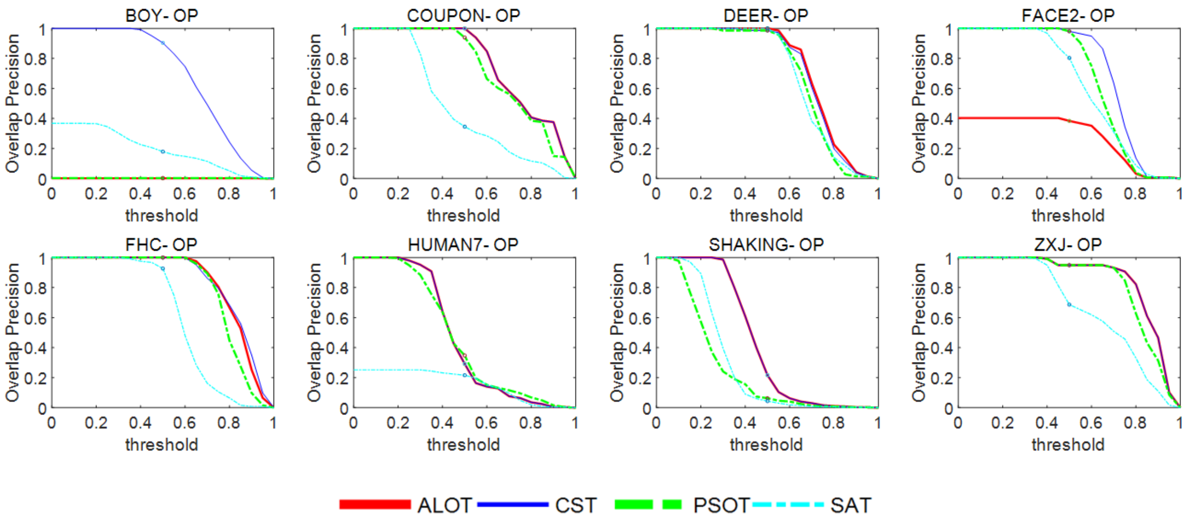


FIGURE 5. Success plots of OPE .

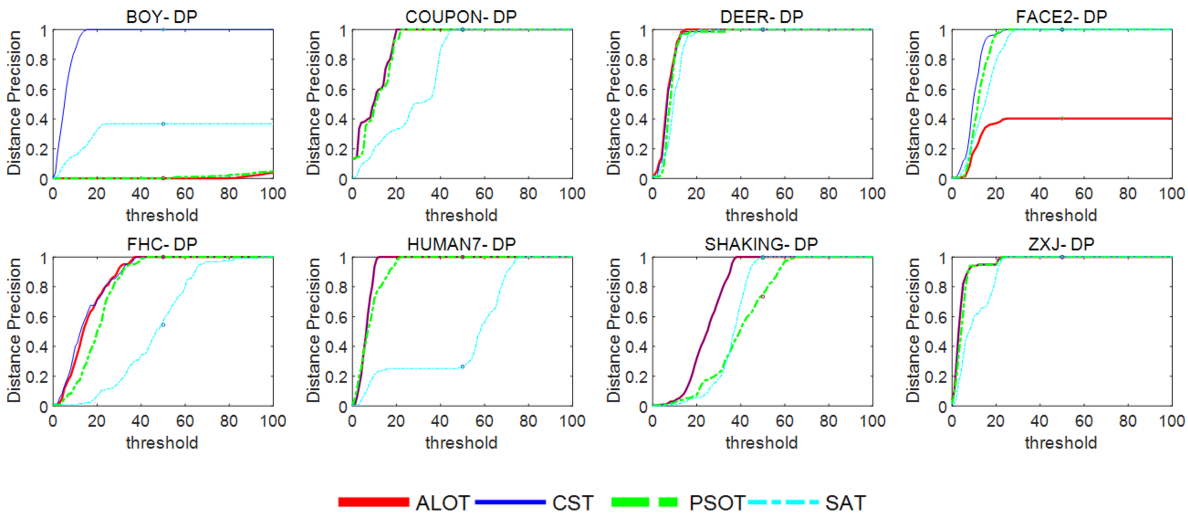


FIGURE 6. Precision plots of OPE .

TABLE 1. Average overlap rate.

Sequences	ALOT	CST	PSOT	SAT
BOY	0.27	0.69	0.01	0.2
COUPON	0.77	0.77	0.74	0.48
DEER	0.74	0.72	0.69	0.70
FACE2	0.28	0.71	0.66	0.62
FHC	0.84	0.83	0.80	0.61
HUMAN7	0.47	0.46	0.47	0.16
SHAKING	0.44	0.44	0.25	0.29
ZXJ	0.85	0.85	0.82	0.67

TABLE 2. Average error center rate.

Sequences	ALOT	CST	PSOT	SAT
BOY	123	6	207	140
COUPON	9	9	11	27
DEER	7	8	8	9
FACE2	213	10	12	15
FHC	15	16	20	46
HUMAN7	6	7	8	48
SHAKING	24	24	39	36
ZXJ	4	4	5	10

ALOT has the better performance in accuracy. The strength of ALOT lies in its capability: 1) Exploration of the search space is guaranteed by the random selection of antlions and random walk of ants around them. The operation avoids that tracker traps to local optimal. 2) Exploitation of search space is guaranteed by the adaptive shrinking space of ants. This strategy ensures that the tracking converges to the target.

SAT delivers the worst tracking performance. This is due to a completely random search mechanism. In the SAT, if the new sampling point has greater energy then it will be accepted as the current sampling point. If the new sampling point has less energy than the current sampling point, then it may still be accepted based on the acceptance probability. This technique encourages exploration while it restricts the capability

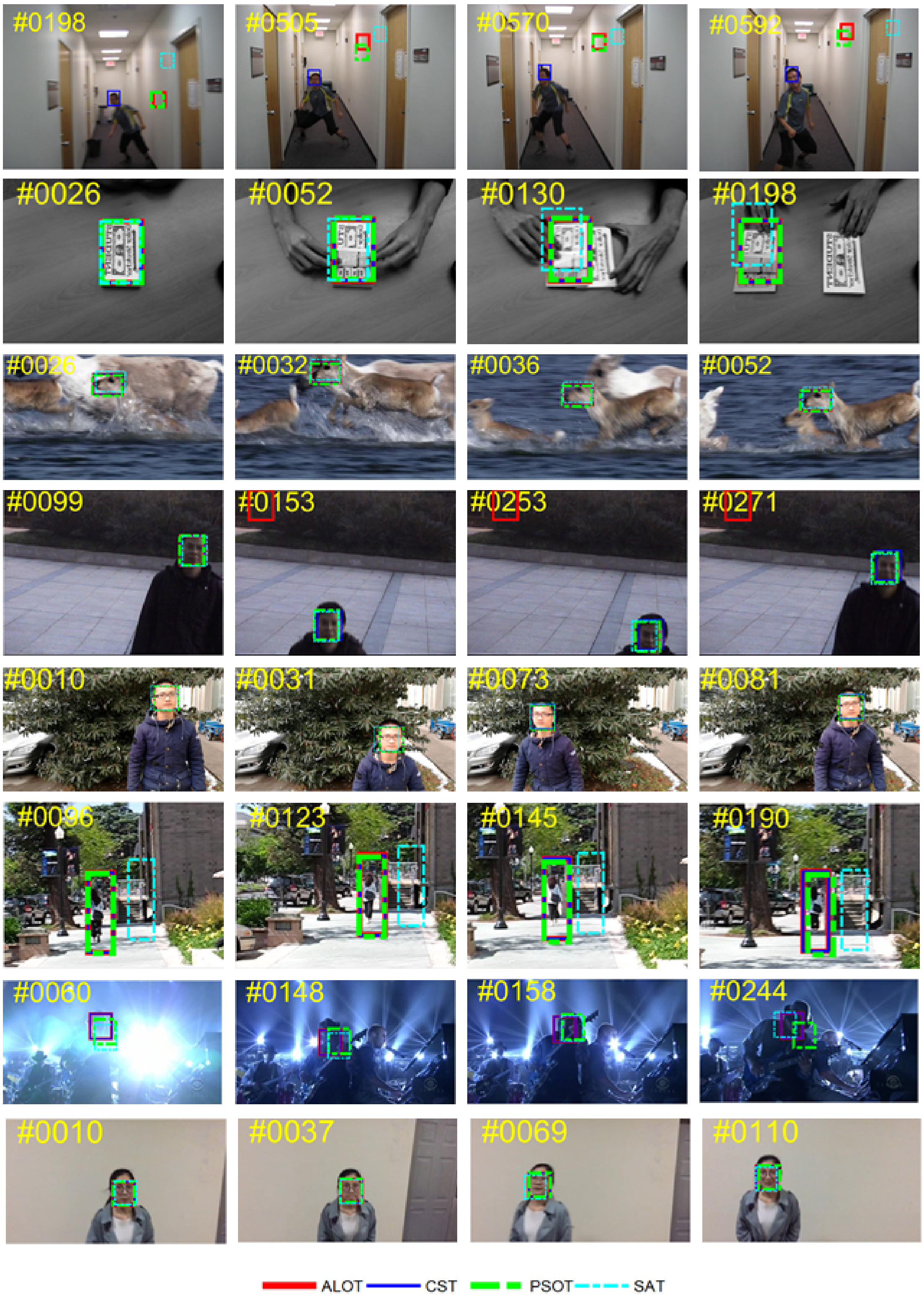


FIGURE 7. The diagram of tracking results .

of exploitation. Therefore, candidates can be escaped from local optimal, but the global optimal also may be missed.

Similar to ALOT, CST is also equipped with efficient local exploitation and global exploration capabilities. The advantage of the *Lévy* flight in comparison to other random walks patterns, such as Gaussian distributed walk, is that it is a scale-free pattern. This operation enhances the ability to diversify. The optimal nest will found, then be used to guide the search in the next iteration. The probability of discovering intrusion facilitates exploration, but a randomly generated nest may result in a poor similar value, then the global optimal may be lost.

In contrast to SAT, PSOT has a weaker global exploration. Particles are updated based on the individual optimal and the global optimal which may reduce the diversity of particles. The reducing results in the precocity of PSO algorithm that makes the algorithm get into local optimal.

In tracking process, the parameter of step size is very important to the tracking accuracy. It is difficult to decide how much step size for the algorithm. Too small or too large step size can greatly affect the performance of the tracking algorithm in practice. Too small step size may lead to failing to jump out of local optimal. If the step size is too large, it is possible to lose the global optimal. For example, the flight speed v in the PSOT and the step size α in the CST play significant roles in tracking accuracy.

Fig. 5 shows the average success plot. The Y axis shows ratio of frames, where, amount of overlap is above threshold, to total frames. More the area under the curve then the tracker is better. Fig. 6 shows the precision plot of the average precision values of all sequences. Here, the Y axis also shows the ratio of frames, where, distance of predicted and ground truth bounding box is below the threshold, to total frames. In precision plots if the slope is higher, then the tracker is better, this is because more sequences have distance of centers lower than the threshold.

2) QUALITATIVE ANALYSIS

In the BOY sequence (row#1), shaking camera causes the abrupt movement of the camera, which poses great challenges to the tracker due to the motion uncertainty and the drastic appearance change. In the COUPON sequence (row#2), there is a similar target and the target appears deformed. In the DEER sequence (row#3), the target undergoes fast motion and some frames are blurred. Meanwhile, there are similar targets. In the FACE2 sequence (row#4), the target has the motion displacement of 88 pixels and slight illumination changing. In the FHC sequence (row#5), the motion displacement reaches 188 pixels. In the HUMAN7 sequence (row#6), the target refers to fast motion and drastic camera shaking. In addition, the brightness of the light changes a lot when target goes through the shade. In the SHAKING sequence (row#7), the target involves appearance changes and dramatic illumination variation. In the ZXJ sequence (row#8), the target concerns blur and the large displacement. The target has the motion displacement of 70 pixels.

It can be obtained from the Fig. 7 that CST has the best performance. Benefiting from the more number of evaluations, CST can track the target throughout every sequence successfully. The ALOT fails in BOY sequence and FACE2 sequence. This is possibly due to the fact that the ALOT is only half times of iterations used to global search. It is essentially making a dent in the ability of explore. Therefore, for ALOT, it is necessary to increase the number of evaluations in some cases (We have verified that the ALOT can realize the tracking of BOY and FACE2 sequence when the number of evaluations reaches 37500. Although a little bit of running time is added, there is a better tracking accuracy).

C. DISCUSSION

Something can be got from the experiments. On the one hand, tracking algorithms make the best of the current optimal candidate to update the position of candidates, which will speed up the convergence speed and improve the tracking accuracy. For example, in ALO algorithm, elite antlion and the antlion randomly selected by roulette jointly affect the ant walk path; in the CS algorithm, the optimal nest of the current iteration is used to perform *Lévy* flight; in PSO algorithm, the position update of every particle is affected by the individual optimal and the current global optimal; for SA algorithm, however, the new sampling point is simply accepted or not. This mechanism lacks sufficient development of the sampling point. So the SA algorithm has lower convergence than others.

On the other hand, it is crucial to balance the ability of global exploration and local exploitation. ALO algorithm proposes random walk mechanism and randomly selecting antlion to promote exploration; then, with the increase of the number of iterations, both adaptive shrinkage of ant space boundary and elite development emphatically, make the iteration result close to global optimal gradually. In CS algorithm, some of the new nest can be generated by *Lévy* walk around the optimal nest obtained so far, which will speed up the local search. While a large part of the new nest is generated by randomization, it should be far enough away from the current optimal nest to ensure that the algorithm does not fall into the local optimal solution. In PSO algorithm, particle cognitive emphasizes that the particle is capable of developing in local area. And particle social emphasizes that the capabilities of global search. However, because the global exploration of the particle is guided by the current global optimal, its global exploration ability is weak, and it is easy to get into local optimal. SA algorithm has a very powerful global exploration capability. But there is no mechanism to focus on local development. Although the algorithm can jump out the local optimal, the target may be lost.

VI. CONCLUSION

This paper addresses the problem of abrupt motion tracking based on swarm optimization. These trackers improve the accuracy of traditional motion evaluation with the help of the global searching mechanism. In these methods, visual tracking is expressed as an optimization process in which swarm optimization is introduced. And locating the target can be interpreted as searching maximum or minimum similarity in the candidate solutions.

In the paper, three classical swarm optimization algorithms are investigated to determine the type optimization problem, namely, ALO, CS and PSO. The tracking results of trackers are compared with the results of SAT.

The comparison experiments of trackers are carried out. Their efficiency is analyzed by the properties of running time and convergence speed, while their accuracy is evaluated by quantitative analysis and qualitative analysis respectively. Meanwhile, the relation between parameter setup and tracking result is explained.

From the experiments, it can arrive that the ALOT algorithm has better performance in average running time, convergence speed. And CST has better performance in quantitative analysis and qualitative analysis. On the whole, these tracking methods are time-consuming,but there is an obvious advantage that is these methods can predict the state of uncertain motion.

Possible future work is to design a unify framework for long-term object tracking combined the merits of traditional trackers and swarm optimization theory.

REFERENCES

[1] G. Zhu, P. Fatih, and H. Li. (2015). "Tracking randomly moving objects on edge box proposals." [Online]. Available: https://arxiv.org/abs/1507.08085

[2] G. Zhu, P. Fatih, and H. Li, "Beyond local search: Tracking objects everywhere with instance-specific proposals," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, Jun. 2016, pp. 943–951.

[3] M. Mueller, N. Smith, and B. Ghanem, "Context-aware correlation filter tracking," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 1387–1395.

[4] K. Lebeda, S. Hadfield, J. Matas, and R. Bowden, "Texture-independent long-term tracking using virtual corners," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 359–371, Jan. 2016.

[5] H. Zhang, X. Zhang, Y. Wang, X. Qian, and Y. Wang, "Extended cuckoo search-based kernel correlation filter for abrupt motion tracking," *IET Comput. Vis.*, vol. 12, no. 6, pp. 763–769, Sep. 2018.

[6] X. Zhang, W. Hu, N. Xie, H. Bao, and S. Maybank, "A robust tracking system for low frame rate video," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 279–304, Dec. 2015.

[7] H. Zhang, H. Fu, T. Zhou, T. Lou, and H. Chen, "Large displacement object tracking via annealed kernel correlation filter," in *Proc. 36th Chin. Control Conf. (CCC)*, Jul. 2017, pp. 10907–10911.

[8] H. Zhang, J. Zhang, Q. Wu, and H. Fu, "Extended kernel correlation filter for abrupt motion tracking," *KSII Trans. Int. Inf. Syst.*, vol. 11, no. 9, pp. 4438–4460, 2017.

[9] M. Khan, M. Valstar, and T. Pridmore, "A multiple motion model tracker handling occlusion and rapid motion variation," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, 2013, pp. 1–12.

[10] M. Khan, M. Valstar, and T. Pridmore, "A generalized search method for multiple competing hypotheses in visual tracking," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, 2014, pp. 2245–2250.

[11] H. Zhang, S. Hu, and X. Zhang, "SIFT flow for large-displacement object tracking," *Appl. Opt.*, vol. 53, no. 27, pp. 6194–6205, 2014.

[12] H. Zhang, Y. Wang, L. Luo, X. Lu, and M. Zhang, "SIFT flow for abrupt motion tracking via adaptive samples selection with sparse representation," *Neurotoxinmenge*, vol. 249, pp. 253–265, Aug. 2017.

[13] J. Kwon and K. M. Lee, "Tracking of abrupt motion using Wang–Landau Monte Carlo estimation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2008, pp. 387–400.

[14] J. Kwon and K. M. Lee, "Wang-Landau Monte Carlo-based tracking methods for abrupt motions," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 4, pp. 1011–1024, Apr. 2013.

[15] X. Zhou and Y. Lu, "Abrupt motion tracking via adaptive stochastic approximation Monte Carlo sampling," in *Proc. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 1847–1854.

[16] F. Wang, X. Li, and M. Lu, "Adaptive Hamiltonian MCMC sampling for robust visual tracking," *Multimedia Tools Appl.*, vol. 76, no. 11, pp. 13087–13106, 2017.

[17] M. Minami, J. Agbanhan, and T. Asakura, "Manipulator visual servoing and tracking of fish using a genetic algorithm," *Ind. Robot, Int. J.*, vol. 26, no. 4, pp. 278–289, 1999.

[18] J.-Y. Chen, H.-H. He, X.-F. Li, and H.-B. Xu, "Fast moving object tracking algorithm based on hybrid quantum PSO," *WSEAS Trans. Comput.*, vol. 12, pp. 375–383, Oct. 2013.

[19] Z. Hao, X. Zhang, P. Yu, and H. Li, "Video object tracing based on particle filter with ant colony optimization," in *Proc. 2nd Int. Conf. IEEE Adv. Comput. Cont. (ICACC)*, vol. 3, Mar. 2010, pp. 232–236.

[20] H. T. Nguyen and B. Bhanu, "Real-time pedestrian tracking with bacterial foraging optimization," in *Proc. IEEE 9th Int. Conf. Advan. Vid. Signal-Based Surveil. (AVSS)*, Sep. 2012, pp. 37–42.

[21] M.-L. Gao, X.-H. He, D.-S. Luo, J. Jiang, and Q.-Z. Teng, "Object tracking using firefly algorithm," *Comput. Vis., IET*, vol. 7, no. 4, pp. 227–237, Aug. 2013.

[22] T. Ljouad, A. Amine, and M. Rziza, "A hybrid mobile object tracker based on the modified Cuckoo Search algorithm and the Kalman Filter," *Pattern Recognit.*, vol. 47, no. 11, pp. 3597–3613, 2014.

[23] M.-L. Gao, Y.-R. Zang, J. Shen, Y.-C. Zhang, and D.-S. Yu, "Visual tracking based on flower pollination algorithm," in *Proc. 35th Chin. Control Conf.*, Chengdu, China, Jul. 2016, pp. 3866–3868.

[24] M.-L. Gao *et al.*, "A novel visual tracking method using bat algorithm," *Neurocomputing*, vol. 177, pp. 612–619, Feb. 2016.

[25] J. Ma, J. Zhao, J. Tian, A. L. Yuille, and Z. Tu, "Robust point matching via vector field consensus," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1706–1721, Apr. 2014.

[26] J. Ma, W. Qiu, J. Zhao, Y. Ma, A. L. Yuille, and Z. Tu, "Robust L_2E estimation of transformation for non-rigid registration," *IEEE Trans. Signal Process.*, vol. 63, no. 5, pp. 1115–1129, Mar. 2015.

[27] J. Ma, J. Jiang, H. Zhou, J. Zhao, and X. Guo, "Guided locality preserving feature matching for remote sensing image registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 56, no. 8, pp. 4435–4447, Aug. 2018.

[28] Y. Wang, S. Hu, and S. Wu, "Visual tracking based on group sparsity learning," *Mach. Vis. Appl.*, vol. 26, no. 1, pp. 127–139, Jan. 2015.

[29] K. Meshgi, S.-I. Maeda, S. Oba, H. Skibbe, Y.-Z. Li, and S. Ishii, "An occlusion-aware particle filter tracker to handle complex and persistent occlusions," *Comput. Vis. Image Understand.*, vol. 150, pp. 81–94, Sep. 2016.

[30] T. Zhou, H. Bhaskar, F. Liu, and J. Yang, "Graph regularized and locality-constrained coding for robust visual tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 10, pp. 2153–2164, Oct. 2017.

[31] T. Zhou, F. Liu, H. Bhaskar, and J. Yang, "Robust visual tracking via online discriminative and low-rank dictionary learning," *IEEE Trans. Cybern.*, vol. 48, no. 9, pp. 2643–2655, Sep. 2018.

[32] H. Zhao and X. Wang, "Robust visual tracking via discriminative appearance model based on sparse coding," *Multimedia Syst.*, vol. 23, no. 1, pp. 75–84, 2017.

[33] C. Sun, F. Li, H. Lu, and G. Hua, "Visual tracking via joint discriminative appearance learning," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 12, pp. 2567–2577, Dec. 2016.

[34] M. Zhai, M. Roshtkhari, and G. Mori. (2016). "Deep learning of appearance models for online object tracking." [Online]. Available: https://arxiv.org/abs/1607.02568

[35] H. Yue, Y. Liu, D. Cai, and X. He, "Tracking people in RGBD videos using deep learning and motion clues," *Neurocomputing*, vol. 204, pp. 70–76, Sep. 2016.

[36] R. J. Mozhdehi and H. Medeiros, "Deep convolutional particle filter for visual tracking," *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3650–3654.

[37] S. Yi, Z. He, X. You, and Y.-M. Cheung, "Single object tracking via robust combination of particle filter and sparse representation," *Signal Process.*, vol. 110, pp. 178–187, May 2015.

[38] J. Yang, K. Zhang, and Q. Liu, "Robust object tracking by online Fisher discrimination boosting feature selection," *Comput. Vis. Image Understand.*, vol. 153, pp. 100–108, Dec. 2016.

[39] J. Ma, J. Jiang, C. Liu, and Y. Li, "Feature guided Gaussian mixture model with semi-supervised EM and local geometric constraint for retinal image registration," *Inf. Sci.*, vol. 417, pp. 128–142, Nov. 2017.

[40] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 5, pp. 564–577, May 2003.

[41] T. Vojir, J. Noskova, and J. Matas, "Robust scale-adaptive mean-shift for tracking," *Pattern Recognit. Lett.*, vol. 49, pp. 250–258, Nov. 2014.

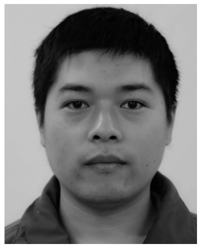
[42] F. Yang, H. Lu, and M.-H. Yang, "Robust superpixel tracking," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1639–1651, Apr. 2014.

[43] Z. Zhou, M. Zhou, and J. Li, "Object tracking method based on hybrid particle filter and sparse representation," *Multimedia Tools Appl.*, vol. 76, no. 2, pp. 2979–2993, Jan. 2017.
[44] T. Zhou, Y. Lu, F. Lv, H. Di, Q. Zhao, and J. Zhang, "Abrupt motion tracking via nearest neighbor field driven stochastic sampling," *Neurocomputing*, vol. 165, pp. 350–360, Oct. 2015.
[45] L. Mihaylova *et al.*, "Overview of Bayesian sequential Monte Carlo methods for group and extended object tracking," *Digit. Signal Process.*, vol. 25, pp. 1–16, Feb. 2014.
[46] S. Mirjalili, "The ant lion optimizer," *Adv. Eng. Softw.*, vol. 83, pp. 80–98, May 2015.
[47] X.-S. Yang and S. Deb, "Cuckoo search via Lévy flights," in *Proc. Nature Biol. Inspired Comput.*, Coimbatore, India, Dec. 2009, pp. 210–214.
[48] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw.*, Nov. 1995, pp. 1942–1948.
[49] S. Bandyopadhyay, S. Saha, U. Maulik, and K. Deb, "A simulated annealing-based multiobjective optimization algorithm: AMOSA," *IEEE Trans. Evol. Comput.*, vol. 12, no. 3, pp. 269–283, Jun. 2008.

HUANLONG ZHANG was born in Lingbao, China, in 1981. He received the M.S. degree from Henan University, Kaifeng, China, in 2007, and the Ph.D. degree from Shanghai Jiao Tong University, Shanghai, China, in 2015. He is currently a Teacher with the Zhengzhou University of Light Industry. His research interests mainly include pattern recognition, machine learning, image processing, and computer vision.

XIUJIAO ZHANG was born in Xuzhou, China, in 1990. She is currently pursuing the bachelor's degree with the Zhengzhou University of Light Industry, Zhengzhou, China. Her research interests focus on machine learning and visual tracking.

YAN WANG was born in Runan, China in 1986. She received the B.S. degree in automation from the School of Computer and Information Engineering, Henan University, China, in 2008, the master's degree in control science and engineering from the School of Automation, Wuhan University of Technology, China, in 2011, and the Ph.D. degree in control science and engineering from the School of Automation, Huazhong University of Science and Technology, China, in 2015. Since 2015, she has been a Lecturer with the School of Electrical and Information Engineering, Zhengzhou University of Light Industry. Her research interests are in the areas of process control, data-driven method, and fault monitoring.



KUNFENG SHI was born in Yucheng, China, in 1986. He received the B.S. degree from the Nanjing University of Posts and Telecommunications in 2008 and the Ph.D. degree from the University of Chinese Academy of Sciences in 2014. Since 2015, he has been with the School of Electrical and Information Engineering, Zhengzhou University of Light Industry, where he is currently a Lecturer. His research interests include 3-D computer vision, with a particular interest in camera calibration and 3-D reconstruction.

JIANWEI ZHANG was born in Nanyang, China, in 1971. He received the M.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2001, and the Ph.D. degree from PLA Information Engineering University, Zhengzhou, China, in 2010. He is currently a Professor. He is currently working at Zhengzhou University of Light Industry, Zhengzhou. His research interests are in broadband information network, distributed system, and information security.

CHAO LI was born in Dongying, China in 1976. He is currently a Vice President of Shandong SNTON Optical Material Technology Co., Ltd., Dongying, China. His major is high-efficiency micropower motor.