A new visual tracking approach based on salp swarm algorithm for abrupt motion tracking

Huanlong Zhang[1], JunFeng Liu[1], Zhicheng Nie[1], Jie Zhang[1] and Jianwei Zhang[2*]

[1] College of electric and information engineering, Zhengzhou University of Light Industry, Zhengzhou 450002,P.R. China

[e-mail:ing@zzuli.edu.cn]

[2] Software Engineering College, Zhengzhou University of Light Industry, Zhengzhou 450002, P.R. China

*Corresponding author: Jianwei Zhang

Abstract

Salp Swarm Algorithm (SSA) is a new nature-inspired swarm optimization algorithm that mimics the swarming behavior of salps navigating and foraging in the oceans. SSA has been proved to enable to avoid local optima and enhance convergence speed benefiting from the adaptive nonlinear mechanism and salp chains. In this paper, visual tracking is considered to be a process of locating the optimal position through the interaction between leaders and followers in successive images. A novel SSA-based tracking framework is proposed and the analysis and adjustment of parameters are discussed experimentally. Besides, the qualitative analysis and quantitative analysis are performed to demonstrate the tracking effect of our proposed approach by comparing with ten classical tracking algorithms. Extensive comparative experimental results show that our algorithm has good performance in visual tracking, especially for abrupt motion tracking.

Keywords: Visual tracking, salp swarm algorithm, abrupt motion, swarm intelligence, comparative analysis

1. Introduction

Over the past few decades, visual tracking has become very popular in machine vision and related fields, which enjoys inclusive applications, including security and video surveillance, traffic monitoring, video analysis, to name a few. Although visual tracking research has achieved remarkable advances, there are still many challenging issues, such as illumination variation, partial occlusions, fast motion, etc. Researchers make most effort to design a tracker for copping these challenges, which can be divided into generative trackers [1-4] and discriminative trackers [5-8]. In recent years, trackers based on Correlation Filter (CF) [9-11] and Deep Learning (DL) [12-13] had successfully been proposed and these methods promote the rapid development of visual tracking.

Most of these methods hypothesize that the target has the circumstance of smooth motion, then trackers search for the target in the region near the position of the last frame. However, these assumptions are not always valid. Abrupt and uncertain motion occurs frequently because of intense motions. In this case, there is a high probability that the target gets away from the region.

Aiming at above problems, a direct method is that enlarging the region to fully cover the motion uncertainty. Nevertheless, exhaustive search is extremely time-consuming due to the existence of a large search space in visual tracking. Thence, an effective search method is crucial to reducing workloads. The swarm optimization algorithms as a kind of search strategy combine global exploration with local exploit to achieve global optimization, and have received extent attention. Some researchers have proposed trackers based on swarm optimization algorithm [14-18] and achieved good results.

Recently, Mirjalili et al. [19] proposes a novel nature-inspired swarm optimization algorithm called Salp Swarm Algorithm (SSA), which imitates the foraging behavior of salp chains in the ocean. The merits of SSA are due to the adaptive nonlinear mechanism and salp chains. Leaders always explore the surrounding space of food and followers can exploit the local space. In addition, one main controlling parameter has the ability to balance exploration and exploitation. The SSA based on adaptive nonlinear mechanism is capable to avoid local solutions and enhance convergence speed.

In this paper, visual tracking is seen as an optimization process of searching for the target in the search space using SSA. The SSA tracker is utilized to solve the problem of abrupt motion. The algorithm based on SSA presents a new tracking framework. And the analysis and adjustment of parameters in SSA are discussed experimentally. It is worth noting that SSA is first introduced into visual tracking. The extensive comparative experimental results demonstrate the new tracker better performance than other representational approaches.

2. Related work

This section only reviews the most relational tracking methods for dealing with abrupt motion.

Non-Swarm optimization based trackers for solving abrupt motion: Aiming at the problem of unsatisfactory tracking results when undergoing abrupt motion in numerous popular tracking methods. Zhang et al. [9] used simulated annealing (SA) to improve the tracking effect of the Kernelized Correlation Filters (KCF) with ability of global optimization. When traditional KCF fails to track between image sequences, the SA mechanism is activated to provide a more reliable image patch to go for better tracking results. Su et al. [20] presented an improved visual saliency model and integrated it to a particle filter tracker to tackle the problem of abrupt motion. Zhou et al. [21] proposed to utilize the stochastic approximation Monte Carlo (SAMC) sampling approach in the Bayesian filter tracking architecture and integrate a new Markov-chain Monte Carlo (MCMC) to solve abrupt motion tracking. For abrupt motions in conventional tracking methods, Zhang et al. [4] put forward a sparse representation based on the tracking architecture, which integrates a novel Scale-invariant feature transform (SIFT) flow tracker (SFT).

Swarm optimization based trackers for solving abrupt motion: In order to solve the problem of abrupt motion tracking, many tracking methods based on swarm optimization algorithms have been proposed. Zhang et al. [14] proposed a sequential particle swarm optimization-based tracking framework via introducing the temporal continuity information into the traditional PSO algorithm. In the proposed algorithm, the parameters based on the fitness values of the particles are dynamically updated, which control the movement of the particles in the swarm and numerous experimental results proved the improved method is more effective and robust, especially for arbitrary motion. Lim et al. [22] combined a sampling strategy based on swarm optimization and the Dynamic Acceleration Parameters (DAP) strategy within the PSO framework to represent a new tracking approach to solve abrupt motion. Due to the existence of a swarm intelligence algorithm namely cuckoo search (CS) with better capabilities in global search, Zhang et al. [23] proposed an improved cuckoo search-based KCF tracker (ECSKCF) to further improve the tracking effect of traditional KCF for abrupt motion. Gao et al. [18] introduced a powerful approach named bat algorithm (BA) to deal with various global optimization problems and BA had successfully solved many challenging issues in visual tracking. Zhang et al. [24] presented a new Moth-flame optimization algorithm (MFO)-based visual tracking method. In this work, the spiral flight of moths and the mechanism that reduces the number of flames gradually were employed to enhance tracking capabilities.

3. Salp swarm algorithm

SSA is a bio-inspired optimization algorithm proposed by Mirjalili et al. [19]. This algorithm is proposed by observing the swarming behavior called salp chains. The natural behaviors are presented by mathematically modeling the salp chains. First of all, the population of salps is divided to two groups: leaders and followers. The division of leaders and followers is on the basis of sorting results which are obtained by the fitness function. Usually, the leaders lead the chain toward a moving food source and the followers follow. During the salp chains moving in the search space, leaders perform the global search and followers implement the local search within their own scope.

3.1 The model hypothesis

In a x-dimensional search space, similar to other swarm optimization techniques, the positions of salps are defined and x denotes the number of variables of a considered problem. Thence, a two-dimensional matrix named P includes the positions of all salps. At the same time, we assume that there is a food source named F in the search space as the chased target of the salp chains. Specifically, F is defined as the best position found until the current iteration.

3.2 The update of leaders position

The leaders will perform adaptive position update around the food source F. The position of the leader the following equation is proposed:

$$P_j^i = \begin{cases} F_j + c_1\left(\left(ub_j - lb_j\right)c_2 + lb_j\right), c_3 \geq 0.5 \\ F_j - c_1\left(\left(ub_j - lb_j\right)c_2 + lb_j\right), c_3 < 0.5 \end{cases} \quad (i=1,2,3,...,L1) \; (j=1,2,3,...,x) \quad (1)$$

where P_j^i denotes the position of the ith salp of leaders in the jth dimension. $L1$ denotes the number of leaders. F_j is the location of the food source in the jth dimension, ub_j indicates the upper bound of jth dimension, lb_j indicates the lower bound of jth dimension, c_1, c_2 and c_3 are random numbers. c_2 and c_3 are both generated in the interval of [0,1] at each iteration. What's more, they decide the next direction and step size for leaders.

3. 3 The update of followers position

According to Newton's law of motion, the following equation for updating the position is

utilized:

$$P_j^i = \frac{1}{2}at^2 + v_0t \quad (i = L1+1, L1+2, ..., n) \tag{2}$$

where $i > L1$, P_j^i denotes the location of ith salp of followers in jth dimension, n denotes

the total number of salps including leaders and followers, t is time, v_0 is the initial speed,

and $a = \frac{v - v_0}{t}$ where $v = \frac{p - p_0}{t}$.

Because the time is iteration in optimization algorithm, the discrepancy between iteration

is equal to 1, and considering $v_0 = 0$, the Eq.2 can be expressed as follows:

$$P_j^i = \frac{1}{2}(P_j^i + P_j^{i-1}) \quad (i = L1+1, L1+2, ..., n) \tag{3}$$

where $i > L1$, P_j^i on the left side of the equation denotes the position of ith salp of

followers in jth dimension in this iteration, P_j^i and P_j^{i-1} on the right side of the equation

indicate the position of ith salp of followers in jth dimension and the position of i-1th salp of
followers in jth dimension in the last iteration,

3. 4 The nonlinear model and the proportional model

As mentioned above, Eq.1 indicates that the update of leaders' position is only related to

the location of the food source F. The random c_1 is described by construct a nonlinear

model. What's more, the nonlinear parameter mechanism can balance exploitation and

exploration in the search space. Therefore, the coefficient c_1 is defined as follows:

$$c_1 = ae^{-(\frac{bl}{L})^2} \tag{4}$$

where, l is the current iteration, L is the maximum number of iterations, a and b are
constant parameters of the nonlinear model.

The population size of salps n is divided into leaders ($L1$) and followers ($n - L1$).
According to Eq.1, Eq.3 and Eq.4, the behavior of leaders and followers plays an important
role respectively in search space. Therefore, the ratio of the number of leaders and followers
is extremely important for the performance of algorithm. The equation for the proportional
model I is as follows:

$$I = \frac{L1}{n - L1} \tag{5}$$

where $L1$ is the number of leaders and n is the total number of salps.

3. 5 The SSA-based pseudo code

The pseudo code of the SSA is presented as follows:

Algorithm1 the Salp Swarm Algorithm
Initial:
1: population size n ; number of leaders $L1$; position of the food source F ; maximum number of iterations K ; upper and lower bound ub, lb ; number of variables dim ;
2: objective fitness function E ;
3: initialize the population of salps $P_i = (i = 1, 2, ..., n)$ considering $ub_i, lb_i (i = 1, 2, ..., \text{dim})$;
Optimization:
1:   **While** ($k_j < K$)
2:     obtain the fitness of each salp
3:     value of F = position of salp of the highest fitness
4:     c_1 value update via Eq.4
5:     **For** each salp (P_i)
6:       **If** ($i \leq L1$)
7:         the location update of the leader salp via Eq.1
8:       **Else**
9:         the location update of the follower salp via Eq.3
10:    **End**
11: **End**
12: modify the locations of salps based on ub_i and lb_i
13: **End**
Output: The position of optimal F

4. SSA-based visual tracking system

The target (food) is given in the current frame (search space). And a group of candidate solutions (leaders and followers) which are randomly generated in the frame by the Salp Swarm Algorithm based on the target position of the previous frame. Meanwhile, the purpose of SSA-based tracker is to find the optimal candidate solution among all candidate solutions. Leaders always guide followers to explore the search space. Based on the above statement, a tracking architecture based on SSA tracker is designed as displayed in **Fig. 1**.

As shown in **Fig. 1**, the tracking target patch is first selected in the initial frame. And the state vector is initialized and a population of n salps is acquired. This state vector is expressed as $x = (x, y, s)$ and (x, y) represents the location of target based on pixel space and s is the scale parameter. Then, an observation model is established to describe the appearance and state target/candidate patches. Next, in order to measure similarity between target patch and candidate patches, a similarity measure strategy is introduced. After that, the best candidate patch is selected by using the salp swarm algorithm. The specific operation of

this process is achieved by maximizing the fitness function. The target location is marked by the SSA optimizer every frame and this displayed patch always indicates the current optimal target. When the last frame is reached, the entire loop ends.
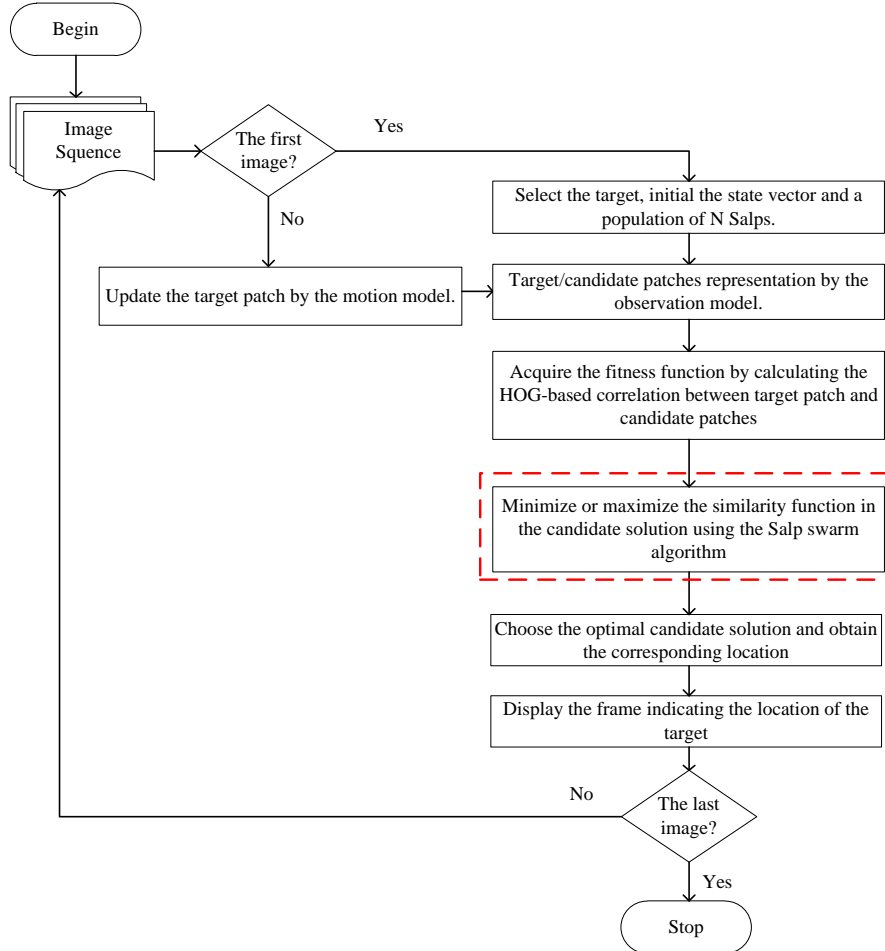

Fig. 1. SSA-based tracking architecture

4.1 The fitness function

The appearance model is an important factor for visual tracking. The HOG feature could capture edges or gradients that are very characteristic of local shape and have the invariance to local geometric and photometric transformations. Visual tracking can be expressed as a process of locating the "best" position of the target in candidate targets according to the fitness value using optimization method. The similarity is computed by:

$$\rho(X,Y) = \frac{Cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}} \tag{6}$$

where $Cov(\cdot)$ denotes covariance and $D(\cdot)$ denotes the variance. X and Y are the HOG feature of the target and candidate samples respectively. The fitness function is shown as

follows:

$$E = 2 + 2 * \rho(X,Y) \tag{7}$$

The fitness value affects how to update the positions of the leaders and followers.

Moreover, the position of the optimal target (food source) is found according to the highest fitness value of all salps.

4.2 Parameter analysis and Adjustment

Proper parameter's selection is a vital aspect for algorithms based on swarm optimization. The convergence speed and tracking accuracy should be taken into consideration simultaneously in the parameter adjustment phase. As the above mentioned, four controlled parameters need to be adjusted for this algorithm, namely population size n, maximum number of iterations K, coefficient in Eq.1 c_1 and proportion between leaders and followers I. Before the experiments, let's first understand the effects of abrupt motion and other problems on visual tracking as presented in **Fig. 2**.

Fig. 2(a) shows a deer moving quickly along the beach with the slight blur and multiple similar targets. Moreover, due to the rapid movement of the deer, its appearance has undergone dramatic changes. As presented in **Fig. 2(b)**, the target suffers from a scale variation, abrupt motion and in-plane rotation in the "FACE1" video. All those challenging factors come into being abundant local distractors that make it possible to lose the target for the tracker.

(a)   (b)

Fig. 2. The abrupt motion problem for parameter analysis and adjustment: (a) DEER, and (b) FACE1

The population size n is first analyzed and the maximum iterate number K is set to be 500. The Euclidean distance between the ground-truth and output position is used to evaluate the performance of n. The performance comparison results are presented in **Fig. 3** for different values of n. **Fig. 3(a)** shows the Trajectories of moving food with the different n

and **Fig. 3(b)** shows the Tracking accuracy comparisons of different n.

Here, moving food is the output position of every iterate operation. The position of target is set to be (400,107). Different values of n can obtain various initial positions and output positions as shown in **Fig. 3(a)**. Pentagrams represent the final positions and the red trajectory and the green trajectory have an intersection called optimal position.
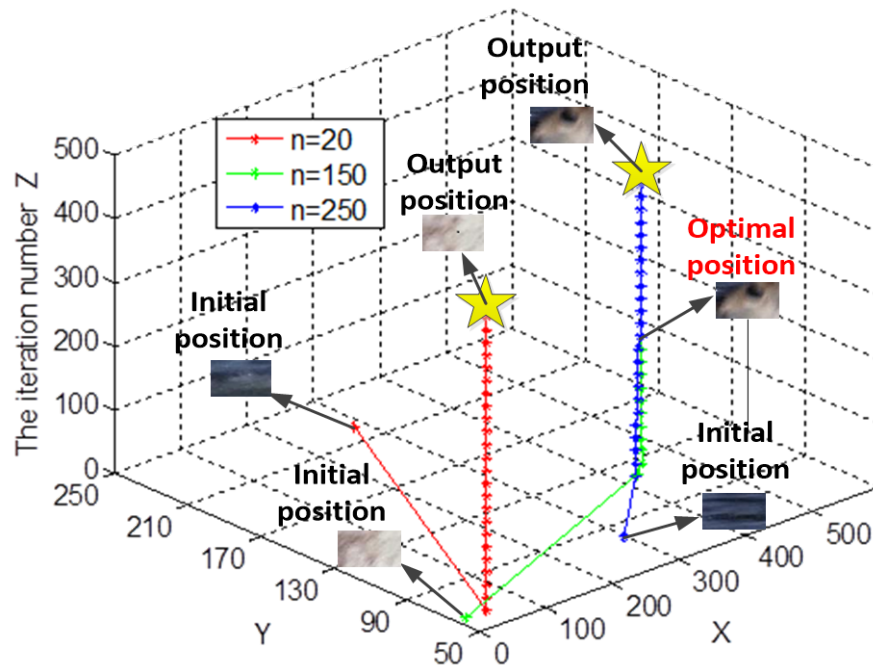

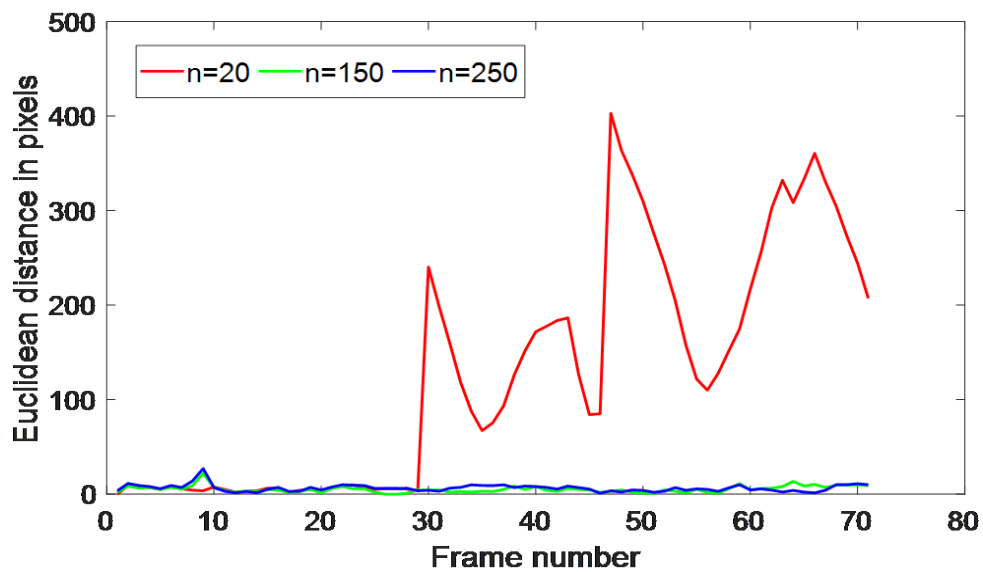**Fig. 3(a).** Trajectories of moving food with the different n

Fig. 3(b). Tracking accuracy comparisons of different n

Fig. 3. The parameter analysis of different population size n

From the **Fig. 3(a)**, we can know that the iterations converge to the position (22, 54) when $n = 20$. Evidently, the target is lost. With only 70 iterations, the optimal position has been obtained and n is 150 at this point. When $n = 250$, the green and blue trajectories converge to form a trajectory after the *238*th iteration. This indicates that the tracking accuracy enhances as the population size increases. However, when the population size reaches a certain value, the tracking accuracy is not improved much and it can increase the tracker's running time.

The **Fig. 3(b)** is a 2D graphic that shows intuitively the Euclidean distance to the target as the frame increases. For $n = 20$, the red line is stable and the distance is very small before 30 frames. After 30 frames, the red line has obvious fluctuations and the Euclidean distance gets more and more bigger. However, the green line and blue line are both relatively stable throughout the tracking process and only a small fluctuation has taken place. Therefore, when $n = 150$ and $n = 250$, they have a similar tracking precision during the tracking process. Considering the accuracy and efficiency, the initial number n is set to be 150.

The comparison results corresponding to various values of I are shown in **Table 1**. I denotes the proportion between number of leaders and number of followers in Eq.5 and it's analyzed by dividing the range of [0,1] into 10 equal parts at intervals of 0.1. The average success frames, average success rate and average max bias are tested as the standard measurements for two challenging videos. The average max bias is defined as the average of maximum bias between output results of tracker and ground-truth of real target.

Table 1. Comparison results of tracking performance for various discovering probability I

I (the proportion)	DEER (with 71 frames)			FACE1 (with 380 frames)		
	Average success frames	Average success rate (%)	Average max bias	Average success frames	Average success rate (%)	Average max bias
0.1	38	54	303	100	26	323
0.2	70	99	36	134	35	59
0.3	47	66	314	90	27	300
0.4	71	100	25	112	29	438
0.5	71	100	26	380	100	32
0.6	71	100	21	252	66	50
0.7	71	100	27	120	32	65
0.8	71	100	23	111	29	56
0.9	71	100	25	134	35	60

We assume the tracker loses the target when DEER's bias reaches to 30 pixels and FACE1's bias reaches to 40 pixels. Besides, tracking experiments of different parameter I are conducted and repeated 3 times. Finally, the average of three experiments is as the final experimental data for every proportion. From the **Table 1**, DEER video's average success

rate reaches 100% when I is 0.4 and subsequently, tracking effect does not fluctuate much. However, only if I is 0.5, tracker doesn't lose the target and achieves good tracking result for FACE1 video. Thus, combining two videos different experimental results, the proportion I is set to be 0.5.

For SSA tracker, one of the most critical parameters is c_1 due to its adaptive nonlinear mechanism. The merit of the parameter is that it enables to enhance balance ability between exploration and exploitation in the search space. In fact, the parametric model of c_1 can be expressed as $c_1 = a * e^{-(b*\frac{l}{L})^2}$ in **Eq.4**. a denotes the max exploration boundary for leaders. b shows the followers' exploitation step. Based on the previous discussion, the illustration in **Fig. 4** is made to show the relationship between tracking accuracy and parameter selection.
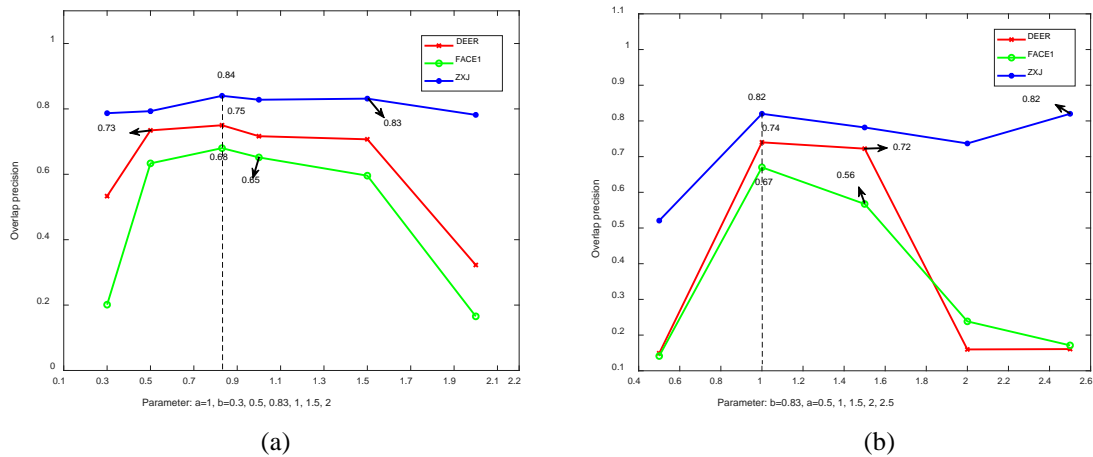

Fig. 4. The relationship between tracking accuracy and parameter selection

Particularly, the overlap precision (OP), which is computed as the percentage of frames in a sequence where the intersection-over-union overlap with the ground-truth bounding box is larger than a threshold, is utilized to indicate the tracking accuracy. In order to obtain better performance for parameters, the values of a and b are fixed separately, and another parameter takes a series of values for various video sequences. Given video sequences DEER, FACE1 and ZXJ. As shown in **Fig. 4**, when a is taken as 1, a series of b is $0.3, 0.5, 0.83, 1.0, 1.5, 2.0$ and when b is taken as 0.83, a series of a is $0.5, 1.0, 1.5, 2.0, 2.5$.

Simultaneously, the max values of OP are presented for three sequences. Known by the trend of change and the maximum overlap rate, the optimal parameter values are set to be

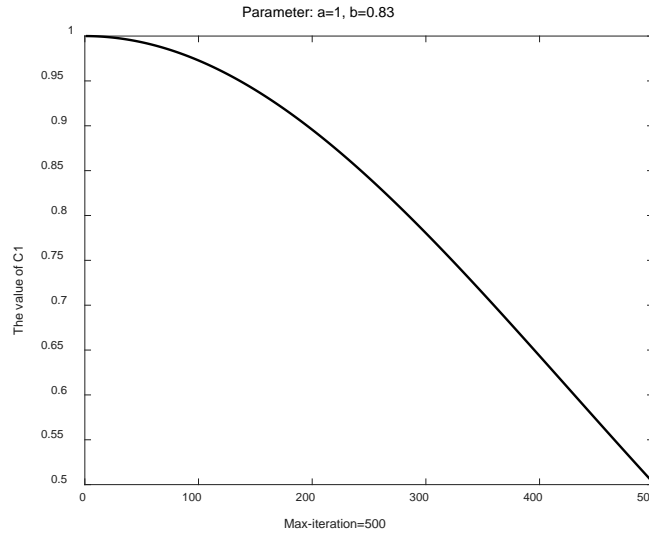$a = 1, b = 0.83$. And the nonlinear model $c_1 = 1 * e^{-(0.83*\frac{l}{L})^2}$ is shown in **Fig. 5**.

Fig. 5. The nonlinear model $c_1 = a * e^{-(b*\frac{l}{L})^2}$

5. Experiments and discussions

We confirm the advantages of SSA-based tracker in visual tracking, especially the abrupt motion problem by choosing a bunch of challenging videos and some different state-of-the-art methods. These challenging videos are made up of three groups that their target displacement is various among the image stream. The first group consists of four sequences MAN, MHYANG, FISH and BOY and their motion displacement between frames is less than 30 pixels. The second group's displacement is between 30 and 50 pixels, including the HUMAN7, JUMPING, DEER and FACE1 sequences. These sequences used in tracking algorithms are obtained on the website **http://www.visual-tracking.net**. In addition, we construct some large displacement motion videos to demonstrate advantages of the proposed method. The third group that their motion displacement is more than 70 pixels contains ZXJ, BLURBODY, FHC, ZT and BLURFACE. Note that the image sequences are listed in **Table 2**.

Table 2. The image sequences

Video	Max displacement	X Max displacement	Y Max displacement	X Total pixels	Y Total pixels
MAN	5	5	3	241	193
MHYANG	7	7	4	320	240
FISH	15	15	13	320	240
BOY	21	21	19	640	480
HUMAN7	31	31	21	320	240
JUMPING	36	18	36	352	288
DEER	38	38	34	704	400
FACE1	39	22	39	720	576
ZXJ	70	70	18	568	320
BLURBODY	76	76	26	640	480
FHC	188	188	104	1920	1080
ZT	256	256	149	1920	1080
BLURFACE	202	202	71	640	480

The proposed tracker is implemented by using MATLAB R2017b. The experiments were performed on a PC with Inter(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHZ and 16.0GB RAM. Our tracker SSA is compared with 10 classic trackers, including High-Speed Tracking with Kernelized Correlation Filters (KCF) [7], Exploiting the Circulant Structure of Tracking-by-detection with Kernels (CSK) [25], Accurate Scale Estimation for Robust Visual Tracking (DSST) [26], Fast Compressive Tracking (FCT) [27], Fast Tracking via Spatio-Temporal Context Learning (STC) [28], Least soft-threshold squares tracking (LSST) [29], Context-Aware Correlation Filter Tracking (CACF) [30], Examples of Adaptive MCMC (AMCMC) [31], Wang-landau monte carlo-based tracking methods for abrupt motions (AWLMC) [32] and Enable Scale and Aspect Ratio Adaptability in Visual Tracking with Detection Proposals (KCFDP) [33]. At the same time, in order to ensure the consistency of the experimental results, the values of the parameters taken by our tracker are consistent in all experiments.

5.1 Algorithms analysis

Generally speaking, appearance representation model is designed depending on the challenging problem that the tracker is confronted with. So, it is important to elect appropriate target representation method for different algorithms. This section analyzes the different representation methods of other algorithms for the target.

CSK is mainly proposed to introduce circulant matrices into Correlation filter (CF)-based target tracking. The candidate targets are from the dense sampling in a window. Then the max response is used to locate the tracked object. CSK represent the target using the

gray-based method, which makes dense sampling easily. KCF is an improvement on CSK. It efficiently incorporates thousands of negative samples from the target's environment and replaces pixels with the Histogram of Oriented Gradient (HOG) features, obtaining great tracking performance. DSST adopts the same target representation method as KCF. But, DSST can adapt the scale variation. To improve precision, KCFDP uses the combination of HOG, intensity, and color name (CN) because of introducing the regional proposals into the detection to induce more flexibility of candidate patches. STC and CACF adopt the Spatio-Temporal context to represent the target, which constructs the relation between the target and its surrounding regions. The relation often keeps stable in image sequences. So, the context can better represent the tracking target. In FCT and LSST, target is represented by sparse theory, which can handle the partial occlusion problem in visual tracking. AMCMC and AWLMC conduct target tracking based on improved Monte Carlo sampling method. The density-of-states of the candidate regions are estimated to guide tracker for adapting abrupt motion. Since color features have little dependence on the size, direction and perspective of the image itself, the two methods obtain a better tracking performance using region color feature.

5.2 Qualitative analysis

5.2.1 The smooth motion group

In the smooth motion group, we know that our tracker works well in **Fig. 6**. In the MAN image sequence (a), only CSK tracking fails at frame #27. However, all the trackers have similar performance except for CSK, AWLMC and AMCMC at frames #33, #93 and #107. In the MHYANG image sequence (b), nearly all trackers can catch up with the target successfully but FCT. For the FISH sequence presented in **Fig. 6** (c), there is almost no abrupt motion in the target except for obvious changes in illumination at frame #157. Only CSK, AWLMC and AMCMC have bad performance. And they also have similar performance at frame #288, #360. However, AWLMC and AMCMC obtain better performance due to the advantage of sampling mechanism at frame #468. For the Boy sequence presented in **Fig. 6** (d), the sudden movement of the camera is caused because of the shaking camera. However, due to the motion uncertainty and the drastic appearance change, the sudden movement poses a great challenge to the tracker. What's more, the scale variation is also very severe due to the movement of the boy. Firstly, LSST fails at frame #329. Then, the target is lost in the CSK, AWLMC and STC methods at frame #490 and #540. There is slight drift at frame # 580 in FCT. DSST, KCF, CACF, AMCMC, KCFDP and our tracker complete the whole video sequence well. In other words, our algorithm basically presents a more stable result.
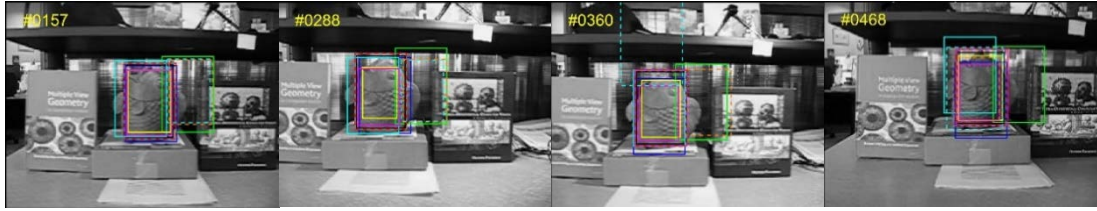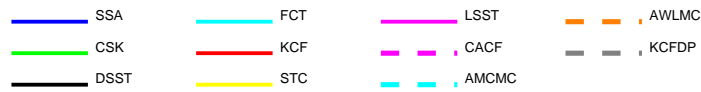
(a)

(b)

(c)

(d)

	SSA		FCT		LSST		AWLMC
	CSK		KCF		CACF		KCFDP
	DSST		STC		AMCMC		

Fig. 6. The tracking results with smooth motion

5.2.2 The slight abrupt motion group

To further test the reliability of our tracker, the motion displacement continues to be expanded. For the HUMAN7 sequence presented in **Fig. 7** (a), the camera shakes violently at frame #117. CACF, AMCMC and our tracker present the better tracking effect in the video sequences, while others all lose the target. In JUMPING sequences that has the motion displacement of 36 pixels as shown in **Fig. 7** (b), the motion blur occurs due to the man's jumping or the camera defocus. Under the circumstances, CSK, DSST, FCT, KCF, STC, AMCMC, AWLMC and KCFDP lose the target unfortunately before the frame #96. However, our tracker has a better performance and recovers tracking quickly. At the same

time, AMCMC and AWLMC recover the target due to the sampling mechanism and our method can compete with LSST. The three video named DEER is presented in **Fig. 7** (c) and it experiences the abrupt motion, motion blur and multiple similar targets. Besides, these challenging factors are also very prominent. At frame #28, DSST, KCF, and STC all lose the object. They cannot complete the video since they are not able to track on several frames. On the whole, CSK and our method obtained the best results. For the FACE1 video as shown in **Fig. 7** (d), the scale changes happen in image frames. CSK and LSST lose the tracking target at frame #210 and #273. And LSST also loses the target at frame 353. Interestingly, AWLMC and AMCMC lose the target at frame #99. However, because the special sampling mechanism make them track the target again at other frames.

(a)

(b)

(c)

(d)

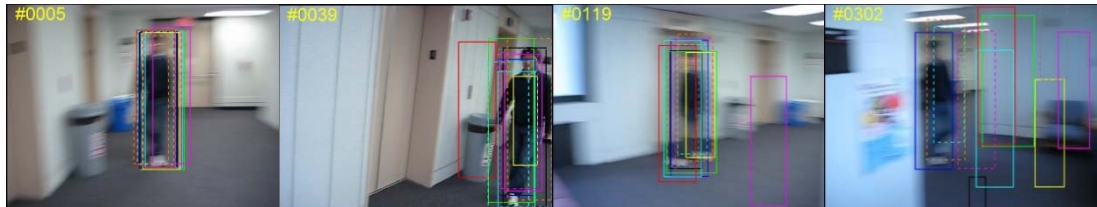**Fig. 7.** The tracking results with slight abrupt motion

5.2.3 The large abrupt motion group

We continue to enhance the motion displacement. And we choose ZXJ, BLURBODY, FHC, BLURFACE and ZT image streams, which their maximal motion displacements are 70, 76, 188, 202 and 256 pixels respectively. Besides, these videos have something challenging in common such as abrupt motion or fast motion. For the ZXJ video as shown in **Fig. 8** (a), only LSST, CACF, AMCMC and our tracker obtain a great performance. From frame #15 to frame #102, more and more trackers lose their targets as the degree of abrupt movement increases. Therefore, our method acquires the better performance for the problem of abrupt motion. In the **Fig. 8** (b), due to the existence of the fast motion and the fuzzy factor that it brings, although most trackers can locate the target successfully before the frame #119, only AWLMC, AMCMC and our tracker still obtain better performance at frame #302. For the FHC video as presented in **Fig. 8** (c), at frame #70, our method and CACF get a better tracking result but other methods either lose the target or have a certain offset. Such performance is duo to the existence of a larger motion displacement at frame #70. In the BLURFACE video sequence as shown **Fig.8** (d), we utilize a violent way that we remove some images in the sequence to design the problem about the frame dropping. At the same time, the sequence has not only the abrupt motion but also severe motion blur. Only our method and AWLMC complete perfectly the whole image sequence by the contrast experiments at frame #310. For the ZT video as shown in **Fig. 8** (e), DSST, CACF, AMCMC, AWLMC and our tracker have a better performance than other methods at frame #108. But AMCMC and AWLMC have a certain drift at that frame.
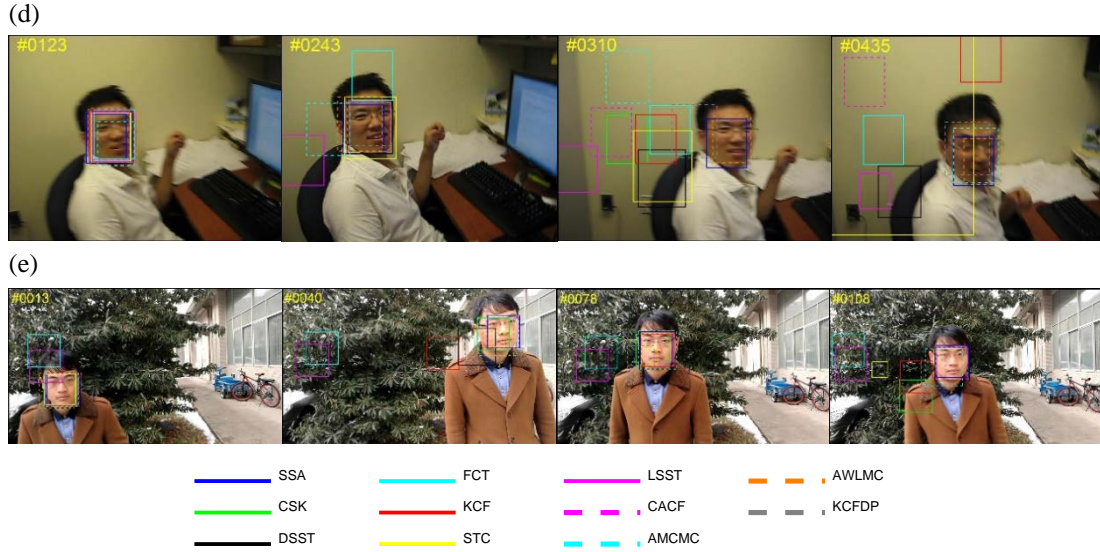
(a)

(b)

(c)

(d)

(e)

	SSA		FCT		LSST		AWLMC
	CSK		KCF		CACF		KCFDP
	DSST		STC		AMCMC		

Fig. 8. The tracking results with large abrupt motion

5.3 Quantitative analysis

In this paper, our tracking results are estimated by using distance precision (DP), overlap precision (OP) and center location error (CLE) in [34]. DP is the relative number of frames whose center location error is less than a certain threshold in the sequence.

$$DP = \frac{N(thresh)}{N} \tag{8}$$

where N is the total frame in a video, N (thresh) denotes the number of frames with CLE under a threshold. And the DP value is set to a threshold of 50 pixels. OP is defined as the percentage of frames whose bounding box overlap above a threshold $t \in [0,1]$. The OP equation is as follows:

$$OP = \frac{\|G_t \cap T_t\|}{\|G_t \cup T_t\|} \tag{9}$$

where T_t is the track region (e.g., bounding box) and G_t is the ground-truth. $\cap$ and $\cup$ represent the intersection and union of two regions, $\|\bullet\|$ denotes the number of pixels in the region and t is the frame number. We set the threshold to be 0.5*CLE and CLE is computed as the average Euclidean distance between the ground-truth and tracking results.

Table 3. Average overlap rate

Sequences	SSA	CSK	DSST	FCT	KCF	STC	LSST	CACF	AMCMC	AWLMC	KCFDP
MAN	0.78	0.03	0.84	0.71	0.84	0.83	0.79	0.84	0.13	0.12	0.79
MHYANG	0.76	0.80	0.81	0.59	0.80	0.69	0.78	0.78	0.59	0.62	0.74
FISH	0.83	0.21	0.80	0.66	0.84	0.58	0.63	0.83	0.20	0.26	0.85
BOY	0.75	0.65	0.84	0.60	0.77	0.55	0.36	0.79	0.70	0.38	0.77
HUMAN7	0.48	0.34	0.36	0.28	0.28	0.28	0.30	0.49	0.53	0.36	0.38
JUMPING	0.70	0.05	0.14	0.20	0.27	0.07	0.60	0.50	0.32	0.05	0.17
DEER	0.75	0.75	0.64	0.66	0.62	0.04	0.71	0.63	0.51	0.38	0.45
FACE1	0.68	0.33	0.79	0.63	0.72	0.65	0.26	0.72	0.61	0.53	0.74
ZXJ	0.84	0.49	0.48	0.45	0.45	0.46	0.79	0.83	0.76	0.61	0.45
BLURBODY	0.64	0.39	0.46	0.44	0.44	0.16	0.07	0.50	0.54	0.55	0.51
FHC	0.78	0.20	0.22	0.26	0.28	0.20	0.28	0.78	0.56	0.54	0.72
BLURFACE	0.83	0.51	0.53	0.23	0.51	0.48	0.30	0.51	0.79	0.65	0.59
ZT	0.85	0.66	0.65	0.09	0.59	0.35	0.09	0.84	0.69	0.60	0.45
Average	0.74	0.42	0.58	0.45	0.57	0.41	0.46	0.69	0.53	0.43	0.58

Table 4. Average center error rate

Sequences	SSA	CSK	DSST	FCT	KCF	STC	LSST	CACF	AMCMC	AWLMC	KCFDP
MAN	3	78	2	4	2	2	2	2	63	66	2
MHYANG	6	4	2	15	4	4	3	7	13	13	4
FISH	5	41	4	12	4	5	4	4	50	37	4
BOY	4	20	2	7	3	26	59	2	12	33	2
HUMAN7	6	18	26	41	48	33	45	6	12	25	44
JUMPING	5	86	37	37	26	67	6	34	42	129	40
DEER	5	5	17	11	21	510	7	23	26	62	76
FACE1	8	100	5	12	6	7	184	5	21	25	6
ZXJ	5	190	21	27	88	24	5	5	6	11	73
BLURBODY	15	73	91	41	68	148	209	34	32	35	105
FHC	23	578	616	371	365	576	392	23	48	55	26
BLURFACE	13	1574	75	116	85	90	162	112	13	17	33
ZT	16	54	54	642	128	100	685	16	30	47	92
Average	8.7	216.9	73.2	102.8	65.3	122.3	135.6	21.0	28.3	42.8	39.1

Table 3 and **Table 4** illustrate a per-sequence comparison results of our method with CSK, DSST, FCT, KCF, STC, LSST, CACF, AMCMC, AWLMC and KCFDP methods. **Table 3** shows the average overlap rate and **Table 4** refers to the average center error rate. And we marked the two best results in every video sequence with red and green. In the tables, the averages show that our proposed tracking method performance is better than others when it

comes to the problem of the larger motion displacement in consecutive images. For example, we can evidently know that our tracker ranks first in the large abrupt motion group including ZXJ, BLURBODY, FHC, BLURFACE and ZT videos. And our tracker also obtains relatively good results in the slight abrupt motion grouping including HUMAN7, JUMPING, DEER and FACE1 videos.
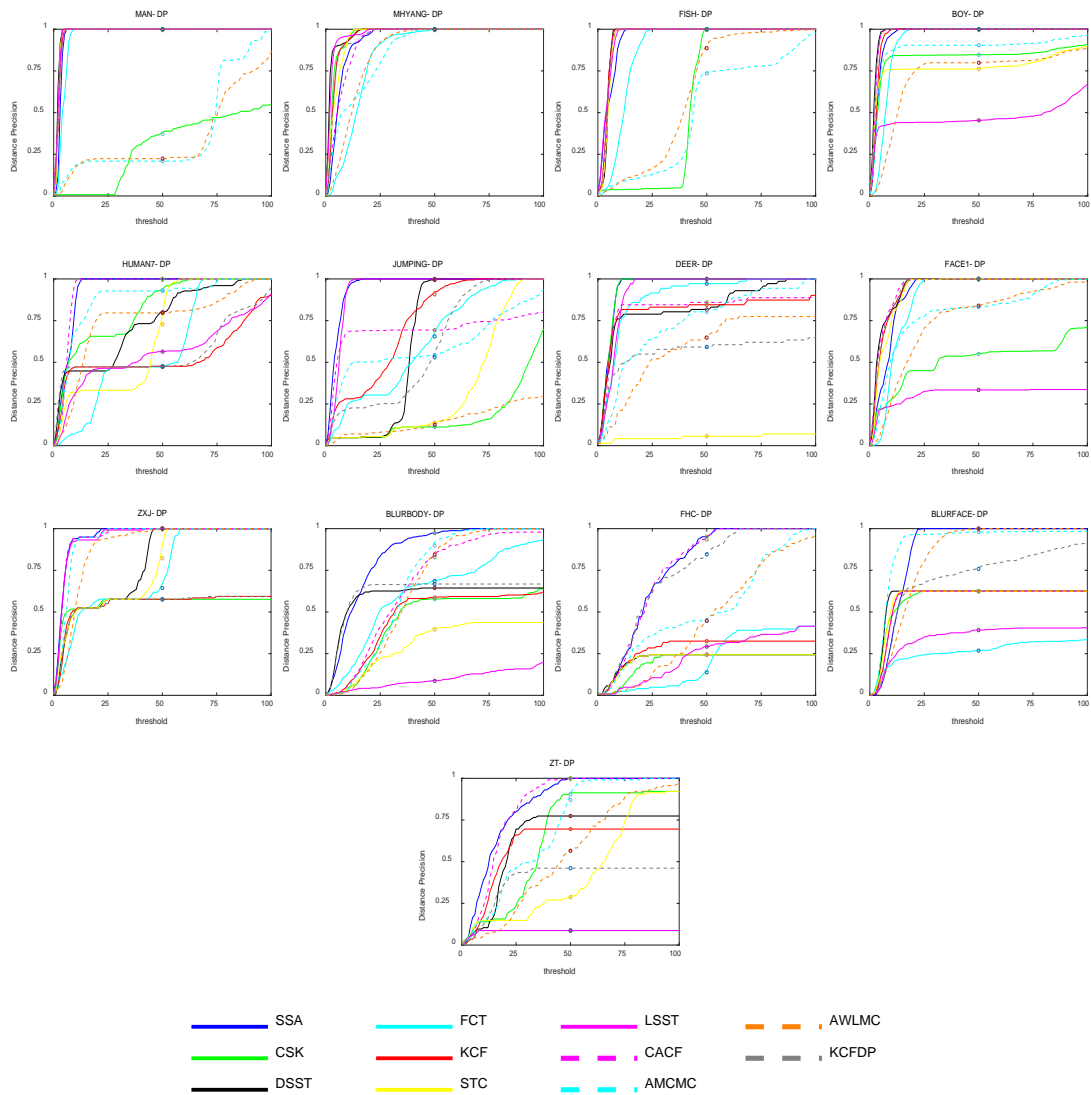

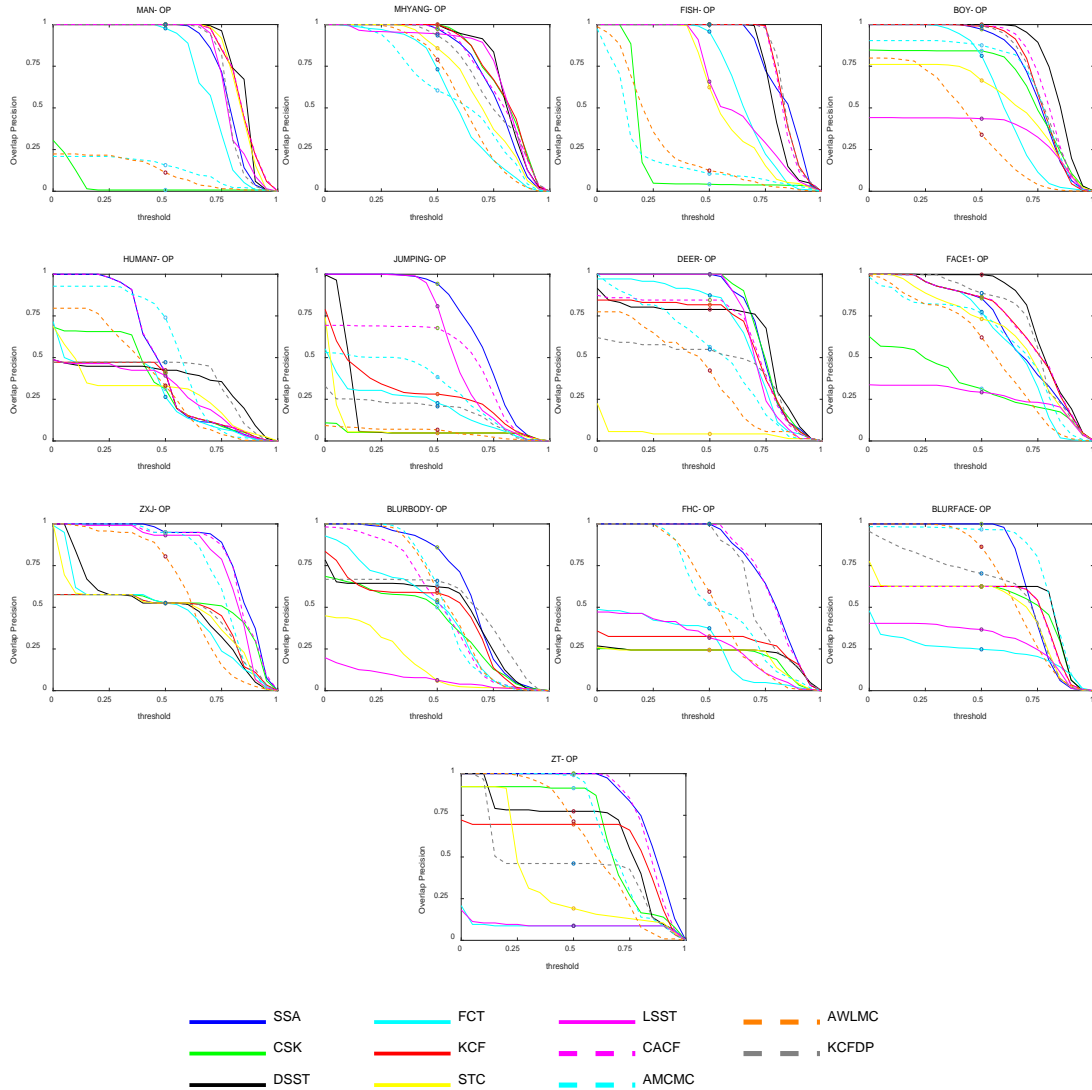**Fig. 9.** The average precision of success plots

Fig. 10. The average precision of success plots

Fig. 9 and **Fig. 10** show the DP and OP of 13 different videos, respectively. We use the form of "line chart" to better analyze the experimental results. It is clearly seen in the line charts that our tracker performs much better than 10 other trackers when meeting a larger motion displacement. All in all, compared with 10 trackers, the proposed method has a greater advantage from table formats and line charts for the problem of abrupt motion.

6. Conclusion

In this paper, for dealing with the problem of abrupt motion, a novel nature-inspired optimizer is proposed and integrated into the visual tracking framework. Visual tracking is

expressed as a process of locating the optimal target by salp chains in the images stream. The parameters analysis and adjustment of the SSA in the tracking framework are discussed. To confirm the tracking effect of the proposed tracker, comparative experiments of qualitative and quantitative analysis of the SSA-based tracking algorithm with ten classic trackers, namely, CSK, DSST, FCT, KCF, STC, LSST, CACF, AMCMC, AWLMC and KCFDP are conducted. Extensive comparative results show that the SSA-based tracker is better than others, especially for the target with abrupt motion. According to the author's knowledge, the SSA-based tracking method is first introduced into a visual tracking system. Choosing the current popular convolutional neural network (CNN) architecture to solve the visual tracking problem will be the focus of our future work.

References

[1] A. D. Jepson, D. J. Fleet, and T. F. EI-Maraghi, "Robust online appearance models for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1296-1311, October, 2003. Article(CrossRefLink)

[2] D. Ross, J. Lim, M.-H Yang, "Adaptive probabilistic visual tracking with incremental subspace update," in *Proc. of European Conf. on Computer Vision*, vol. 3022, pp. 470-482, 2004. Article(CrossRefLink)

[3] H. Zhang, S. Hu, X. Zhang, and L. Luo, "Visual Tracking via Constrained Incremental Nonnegative Matrix Factorization," *IEEE signal processing letters*, vol. 22, no. 9, pp. 1350-1353, September, 2015. Article(CrossRefLink)

[4] H. Zhang, Y. Wang, L. Luo et al., "SIFT flow for abrupt motion tracking via adaptive samples selection with sparse representation," *Neurocomputing*, vol. 249, no. 2, pp. 253-265, August, 2017. Article(CrossRefLink)

[5] S. Avidan, "Support vector tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 8, pp. 1064-1072, August, 2004. Article(CrossRefLink)

[6] N. Wang, M. Yang, and D.Y. Yeung, "Learning a deep compact image representation for visual tracking," *Nips*, pp. 809-817, January, 2013. Article(CrossRefLink)

[7] J. F. Henriques, et al., "High-Speed Tracking with Kernelized Correlation Filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no.3, pp. 583-596, March, 2015. Article(CrossRefLink)

[8] D. Wang, H. Lu, "Fast and Robust Object Tracking via Probability Continuous Outlier Model," *IEEE Transactions on Image Processing*, vol. 24, no.12, pp. 5166-5176, December, 2015. Article(CrossRefLink)

[9] H. Zhang, J. Zhang, and Q. Wu et al., "Extended kernel correlation filter for abrupt motion tracking," *KSII Transactions on Internet and Information Systems*, vol. 11, no. 9, pp. 4438-4460, September, 2017. Article(CrossRefLink)

[10] M. Danelljan, A. Robinson, and F. S. Khan et al., "Beyond correlation filters: Learning continuous convolution operators for visual tracking," in *Proc. of European Conference on Computer Vision*, vol. 9909, pp. 472–488, September, 2016. Article(CrossRefLink)

[11] M. Danelljan, G. Bhat, and F. S. Khan et al., "ECO: Efficient Convolution Operators for Tracking," in *Proc. of The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6638-6646, 2017. Article(CrossRefLink)

[12] D. Held, S. Thrun, and S. Savarese, "Learning to track at 100 fps with deep regression networks," in *Proc. of European Conference on Computer Vision*, vol. 9905, pp. 749–765, September, 2016. Article(CrossRefLink)

[13] J. Valmadre, L. Bertinetto, and J. F. Henriques et al., "End-To-End Representation Learning for Correlation Filter Based Tracking," in *Proc. of The IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2805-2813, 2017. Article(CrossRefLink)

[14] X. Zhang, and W. Hu et al., "Sequential particle swarm optimization for visual tracking," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1-8, June, 2008. Article(CrossRefLink)

[15] V. John, E. Trucco, and S. Ivekovic, "Markerless human articulated tracking using hierarchical particle swarm optimisation," *Image and Vision Computing*, vol. 28, no. 11, pp. 1530-1547, November, 2010. Article(CrossRefLink)

[16] H. Nguyen, and B. Bhanu, "Real-time Pedestrian Tracking with Bacterial Foraging Optimization," in *Proc. of IEEE Ninth International Conference on Advanced Video and Signal-based Surveillance*, pp. 37-42, 2012. Article(CrossRefLink)

[17] M. Gao, L. Yin, and G. Zou et al., "Visual tracking method based on cuckoo search," *Optical Engineering*, vol. 54, no. 7, pp. 1-10-10, July, 2015. Article(CrossRefLink)

[18] M. Gao, J. Shen, and L. Yin et al., "A novel visual tracking method using bat algorithm," *Neurocomputing*, vol. 177, pp. 612-619, February, 2016. Article(CrossRefLink)

[19] S. Mirjalili, A. H. Gandomi, and S. Z. Mirjalili et al., "Salp Swarm Algorithm: A bio-inspired optimizer for engineering design problems," *Advances in Engineering Software*, vol. 114, pp. 163-191, December, 2017. Article(CrossRefLink)

[20] Y. Su, Q. Zhao, and L. Zhao et al., "Abrupt motion tracking using a visual saliency embedded particle filter," *Pattern Recognition*, vol. 47, no. 5, pp. 1826-1834, May, 2014. Article(CrossRefLink)

[21] X. Zhou, Y. Lu, and J. Lu et al., "Abrupt Motion Tracking Via Intensively Adaptive Markov-Chain Monte Carlo Sampling," *IEEE Transactions on Image Processing*, vol. 21, no. 2, pp. 789-801, February, 2012. Article(CrossRefLink)

[22] K. L. Mei, C. S. Chan, and D. Monekosso et al., "Refined particle swarm intelligence method for abrupt motion tracking," *Information Sciences*, vol. 283, pp. 267-287, November, 2014. Article(CrossRefLink)

[23] H. Zhang, X. Zhang, and Y. Wang et al., "Extended cuckoo search-based kernel correlation filter for abrupt motion tracking," *IET Computer Vision*, vol. 12, no. 6, pp. 763-769, September, 2018. Article(CrossRefLink)

[24] H. Zhang, and X. Zhang et al., "A Novel Visual Tracking Method Based on Moth-Flame Optimization Algorithm," *PRCV*, vol. 11259, pp. 284-294, November, 2018. Article(CrossRefLink)

[25] J. F. Henrigues, R. Caseiro, P. Martins, and J. Batista, "Exploiting the circulant structure of tracking-by-detection with kernels," *Computer Vison-ECCV 2012*, vol. 7575, pp. 702-715, 2012. Article(CrossRefLink)

[26] M. Danelljan, G. Hger, F. S. Khan and M. Felsberg, "Accurate scale estimation for robust visual tracking," in *Proc. of the British Machine Vison Conference BMVC, September*, 2014. Article(CrossRefLink)

[27] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *The IEEE Transactions on Pattern Analysis Machine Intelligence*, vol. 36, no. 10, pp. 2002-2015, 2014. Article(CrossRefLink)

[28] K. Zhang, L. Zhang, Q. Liu, D. Zhang, and M.-H. Yang, "Fast visual tracking via dense spatiotemporal context learning," *ECCV*, vol. 8693, pp. 127-141, 2014. Article(CrossRefLink)

[29] D. Wang, H. Lu and M. Yang, "Robust Visual Tracking via Least Soft-threshold Squares," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp. 1709-1721, September, 2016. Article(CrossRefLink)

[30] M. Mueller, N. Smith, B. Ghanem, "Context-Aware Correlation Filter Tracking," in *Proc. of The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1396-1404, July, 2017. Article(CrossRefLink)

[31] G. O. Roberts and J. S. Rosenthal, "Examples of Adaptive MCMC," *Journal of Computational and Graphical Statistics*, vol. 18, no. 2, pp. 349–367, 2009. Article(CrossRefLink)

[32] J. Kwon and K. M. Lee, "Wang-landau monte carlo-based tracking methods for abrupt motions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 4, pp. 1011–1024, April, 2013. Article(CrossRefLink)

[33] D. Huang, and L. Luo et al., "Enable Scale and Aspect Ratio Adaptability in Visual Tracking with Detection Proposals," *BMVC*, no. 185, pp. 1-12, 2015. Article(CrossRefLink)

[34] Y. Wu, J. Lim and M.-H. Yang, "Online Object Tracking: A Benchmark," in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2411-2418, October, 2013. Article(CrossRefLink)

Huanlong Zhang received the M.S. degree from Henan University, Kai Feng, China, in 2007, and Ph.D. degree at Shanghai Jiao Tong University, Shang Hai, China, in 2015. Now, he is a teacher at Zhengzhou University of Light Industry. His research interests mainly include pattern recognition, machine learning, image processing and computer vision.

Junfeng Liu is a student in College of electric and information engineering from China Zhengzhou University of Light Industry. He is currently studying toward the bachelor's degree in automation specialty, Zhengzhou University of Light Industry, Henan, China. His research interests are machine vision and pattern recognition.
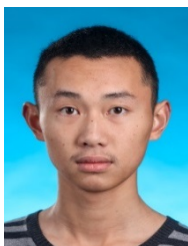
Zhicheng Nie is a student in College of electric and information engineering from China Zhengzhou University of Light Industry. He is currently studying toward the bachelor's degree in automation specialty, Zhengzhou University of Light Industry, Henan, China. His research interests are machine vision and pattern recognition.

Jie Zhang was born in 1987. He is a Ph.D. and a lecture in Zhengzhou University of Light Industry. His research interests are image processing and compressed sensing.

Jianwei Zhang is a professor. He received the M.S. degree from Hua-zhong University of Science & Technology, Wuhan, China, and Ph.D. degrees from PLA Information Engineering University, Zhengzhou, China, in 2010. He is currently working at Zhengzhou University of Light Industry, Zhengzhou, China. His research interests are in broadband information network, distributed system, Information security, etc.