IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received November 25, 2017, accepted January 9, 2018, date of publication January 15, 2018, date of current version March 9, 2018.

*Digital Object Identifier 10.1109/ACCESS.2018.2793298*

# Efficient Tracking of Moving Target Based on an Improved Fast Differential Evolution Algorithm

**LAIJIE LIN[1] AND MIN ZHU[1,2]**
[1]Department of Information Technology, Wenzhou Vocational and Technical College, Wenzhou 325035, China
[2]School of Telecommunication and Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Corresponding author: Laijie Lin (wzyllj@126.com)

This work was supported by the Postgraduate Research and Practice Innovation Program of Jiangsu Province under Grant KYLX15-0840.

**ABSTRACT** Computer vision, which is used to detect and track a specific target in image sequences, has drawn great attention in recent years. The process of tracking can be formulated as a dynamic optimization problem that identifies the optimal position of the target in each image. Differential evolution (DE), who owns the advantages of simplicity, parallel computing, and self-adaptive search for global optimization, is envisioned as a promising algorithm to provide effective target tracking. In this paper, several improvements are made in DE for better adaptability in target tracking. Specifically, we introduce two inferior individuals into the mutation stage, which further enriches the diversity of the population and speeds up the offspring's evolution. We also proceed several image preprocessing and build an adaptive Gaussian mixture model of the target to deal with the complex tracking scenarios. Experimental results show that the designed tracking algorithm based on our improved DE demonstrates a higher tracking accuracy and faster tracking speed in several challenging tracking scenarios.

**INDEX TERMS** Computer vision, target tracking, differential evolution, inferior individuals, image preprocessing, GMM.

## I. INTRODUCTION

As a comprehensive discipline which includes image processing, machine learning, pattern recognition and other adjunct areas, computer vision (CV) has attracted many scholars' attention at home and abroad in recent years. CV is a kind of simulation of biological vision system, its main purpose is to extract information from a series of image processing. As two key steps of simulation, the object detection and target tracking have always been heatedly debated in field of CV lately.

Object detection aims to separate the object from the background in each image, it is necessary for the subsequent modeling, matching and tracking. Various typical detection methods have been researched, such as template matching [1]–[3], feature analysis [4], [5] and statistical analysis [6], [7]. All of them mainly concentrate on three basic methods, i.e., background subtraction, frame difference and optical flow. Background subtraction is the most common detection method, especially for fixed camera, which distinguishes moving targets from background according to the difference in gray value of the image judged by a threshold beyond which moving targets are detected. Frame difference reconstructs the profile of moving targets by computing the difference of two adjacent frames, which is considered as a simple and straightforward way to detect fast moving targets in condition of moving. Optical flow views the sequence images as the continuous optical flow information. Find out the potential relationship between two adjacent frames and establish motion information of moving targets by the variation of the pixels of the image sequences in time domain and the correlation of adjacent frames.

The goal of target tracking is to locate and orientate the moving target accurately in video or image sequences. Different types of reasonable algorithms are available for target tracking based on a portion of prior knowledge extracted from the targets. Feature extraction, e.g., color feature [8], [9], texture feature, edge feature [10], is usually applied to provide an effective description of the target for subsequent matching. The extraction should be abundant in descriptive information as well as efficient in mathematical processing. After feature extraction, the corresponding tracking algorithm is designed. The tracking of the target is usually viewed as a state estimation problem, whose optimal state can be predicted by Bayesia estimation, such as Particle Filter (PF) [11] and Kalman Filter [12]. Essentially, target tracking is equivalent to a dynamic optimization process, which endeavors to search for the optimal location of the target in all candidate locations in the image. To the best knowledge, the current various tracking algorithms roughly fall into three classes: deterministic algorithms [13]–[15] and stochastic algorithms [16]–[18].

After years of effort, abundant tracking algorithms have been proposed for various tracking scenarios. Limited to the complexity and variability in tracking, several existing tracking algorithms are weak in long-time effective tracking. Therefore, many innovative tracking schemes have been proposed for increasingly complex tracking scenarios. For example, [19] and [20] implement a combination of online learning with the traditional tracking schemes. Ross *et al.* [21] proposed an effective IVT Tracker, which can update the appearance model of the target dynamically. Zhang *et al.* [22] introduced Particle Swarm Optimization (PSO) to PF to alleviate the degradation problems of PF, which greatly enhance the tracking accuracy of the algorithm.

In addition, the research [23] shows that the adoption of genetic algorithm to PF can also achieve outstanding tracking results. Getting enlightenment from aforementioned literatures [22], [24], this article leverages differential evolution algorithm, which is one of the popular genetic algorithms, in the application of target tracking. Also, numerical results are reported that demonstrating the excellent and robust performance of our designed algorithm in several challenging tracking scenarios.

## II. TRACKING PROCESS BASED ON DE

As mentioned above, the process of tracking can be viewed as a dynamic optimization problem. DE algorithm leveraged in solving optimal problems is considered simple, parallel and self-adaptive.

Generally, if a specific optimization problem is continuous and linear, or can be transformed into a convex problem [25], [26], the local optimal solution searched by several traditional classical methods (e.g., gradient, Hessian matrix, Lagrange multipliers, and gradient descent method) will be equivalent to the global optimal solution. However, the equivalence is inapplicable for non-convex problem and, consequently, a complicated global optimization problem should be solved. Therefore, we introduce some intelligent optimization algorithm to tackle this complex problem about global search.

Evolutionary computation, of which genetic algorithm is the best-known one, is a global probability search algorithm simulating biological evolutionary mechanisms such as natural selection, heredity and mutation. The evolution of a species can be taken as a long-term dynamic optimization problem and, accordingly, an individual of a species who experiences the natural selection mechanism is equivalent to a feasible solution which goes through several iterations. Compared with the traditional optimization algorithms, evolutionary algorithms can realize parallel search through the evolution and finally converge to a global optimal solution with a higher probability. Therefore, the evolutionary algorithms possess fast convergence speed and excellent optimization performance in solving various scientific and engineering problems. The tracking process based on DE algorithm applied in this paper is illustrated in Fig. 1.

In order to overcome the shortcomings in DE algorithm such as stagnant evolution and premature convergence,

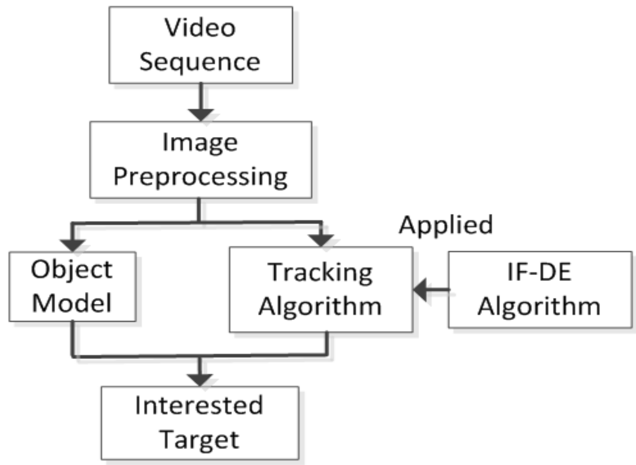


**FIGURE 1.** Tracking process based on DE algorithm.

we propose an improved fast differential evolution (IF-DE) algorithm for better adaptability in target tracking.

The rest of this paper is organized as follows. Chapter III gives a detailed introduction of the DE algorithm and elaborates the improvements in IF-DE algorithm. Chapter IV introduces the tracking framework based on IF-DE. Experiments on several tracking scenarios are conducted in Chapter V to show the performance of proposed tracking algorithm. Finally, Chapter VI gives a conclusion of this paper.

## III. INTRODUCTION OF IF-DE

In contrast to evolutionary algorithm, DE algorithm, which was originally proposed by Storn and Price [27], simplifies the genetic manipulation and utilizes several difference vectors to simulate the variation. DE remains the advantage of global search as well as robustness and easy implementation, so it has been widely applied to many tough optimal problems.

The implementation of DE comprises four stages. In the initial stage, several random vectors are generated to simulate the individuals of a population. Then start the next three evolutionary stages: variation, crossover and selection.

### A. STANDARD DE ALGORITHM

#### 1) POPULATION INITIALIZATION

In the solution space of $R^D$, the initialization of population is expressed as:

$$\{x_i^0 = (x_{i,1}^0, x_{i,2}^0, \ldots x_{i,D}^0, )|i = 1, 2, \ldots|P|\}, \tag{1}$$

where $g = 0$ denotes the initial generation and $|P|$ is the population size. In order to ensure that the initial population covers the entire search space, all dimensions of every random vector, namely individual of a population, should be initialized randomly according to a uniform distribution.

#### 2) MUTATION OPERATION

After the population initialization, DE switches to the evolutionary stage. The target vector $x_i^g$ will mutate into a variant vector $v_i^g$ by the mutation operation. Specifically, two individual vectors $x_{r1}^g, x_{r2}^g$ are randomly selected from the current generation. Then its mutation operation is shown in equation (2):

$$v_i^g = x_{best}^g + f_i(x_{r1}^g - x_{r2}^g), \tag{2}$$

where $x_{best}^g$ represents the optimal individual of the current $g$ generation. Also, the scale factor$f_i$, which is expressed as the scaling of the difference $x_{r1}^g - x_{r2}^g$, regulates the step length and speed of the evolution.

As a general rule, different mutation strategies may bring distinct evolutionary results. (2), adopted in this paper, is known as the DE/Best/1 strategy, while other common variation strategies are shown as follows:

*a: DE/RAND/1*

$$v_i^g = x_{r0}^g + f_i(x_{r1}^g - x_{r2}^g) \tag{3}$$

*b: DE/ BEST/2*

$$v_i^g = x_{best}^g + f_{i1}(x_{r1}^g - x_{r2}^g) + f_{i2}(x_{r3}^g - x_{r4}^g) \tag{4}$$

*c: DE/CURRENT-TO-RAND/1*

$$v_i^g = x_i^g + f_{i1}(x_{r1}^g - x_i^g) + f_{i2}(x_{r2}^g - x_{r3}^g) \tag{5}$$

*d: DE/CURRENT-TO-BEST/1*

$$v_i^g = x_i^g + f_{i1}(x_{best}^g - x_i^g) + f_{i2}(x_{r1}^g - x_{r2}^g) \tag{6}$$

It's worth noting that the first items of above variation formulas, i.e., $x_{r0}^g$, $x_{best}^g$ and $x_i^g$, represent the target vectors of the population (or parent vector), whereas the remaining vectors (e.g., $x_{r1}^g, x_{r2}^g$ and $x_{r3}^g$) are selected randomly from the current $g$ population.

### 3) CROSSOVER OPERATION

Finish the process that the target vector $x_i^g$ mutate into variant vector $v_i^g$ by the above-mentioned mutation operation. Then, the cross operator is implemented on $v_i^g$ to generate the trial vector $u_i^g$. The crossover operation is shown as follows:

$$u_{j,i}^g = \begin{cases} v_{j,i}^g, & if\ rand(m,n) \le CR_i \\ x_{j,i}^g, & otherwise, \end{cases} \tag{7}$$

where $CR_i$ is the crossover probability, locating in the interval $[m, n]$. During the process of crossover, if $rand(m, n)$ is smaller than the crossover probability $CR_i$, then the trial vector $u_i^g$ inherits the *j-th* dimension of variant vector $v_i^g$, otherwise inherits the corresponding dimension of the parent vector $x_i^g$.

### 4) SELECTION OPERATION

Actually, not all the mutated individuals can be selected into the next population by above two evolutionary operations. DE selects "the superior" from $x_i^g$ and $u_i^g$ to compose the offspring population. Selection operation is shown as follows:

$$x_i^{g+1} = \begin{cases} u_i^g, & if\ f(u_i^g) \le f(x_i^g) \\ x_i^g, & otherwise \end{cases} \tag{8}$$

In this case, the test function $f(\cdot)$ is used to calculate the fitness value of the individual. When the trial vector $u_i^g$ is superior to the target vector $x_i^g$, the population determines as it is successfully updated in the $g$-*th* generation. Then the corresponding $f_i$ and $CR_i$ are called successful evolutionary parameters.

## B. CURRENT WORKS

Reviewing the main improvements on DE, current works mainly concentrate on parameter adaption, variation strategy and some introductions of traditional optimal algorithms.

With regard to parameter adaption, several classical DE algorithms have been proposed. For instance, the SaDE algorithm, proposed in [28], not only utilizes adaptive evolutionary parameters, but also employs different mutation strategies. Based on SaDE, then Zhang and Sanderson [29] further proposed an innovative JADE algorithm. By some reasonable distributions, JADE algorithm is more flexible in parameters' adjustment. What's more, JADE randomly selects an individual as $x_{best}^g$ from the $100p\%$ optimal individuals, which further avoids the local convergence of the algorithm. Different from JADE, Liu and Lampinen [30] adjusted the parameters by a fuzzy controller, and proposed a new adaptive DE algorithm (called FADE). From their verified experimental results, all these improved parameter adaptations have achieved excellent performance.

When DE have multiple variation strategies, how to select an optimal one dynamically is very important to improve the performance of the algorithm. For this reason, Qin and Suganthan [28] switched the current variation strategy dynamically based on the previous evolutionary results. In this case, if an individual succeeds in updating, its corresponding variation strategy will be adopted with a higher probability in the next generation. Based on this study, Qin *et al.* [31] alternated four mutation strategies and their corresponding adaptive parameters and further improved the performance of DE algorithm. Similarly, Mallipeddi *et al.* [32] also proposed an EPSDE algorithm with a pool of three alternate mutation strategies. Based on SaDE and EPSDE, Wang *et al.* [33] made some changes that all the variation strategies in the strategy pool are adopted through parallel computation, then the optimal trial vector is selected into the next generation.

## C. IMPROVEMENTS ON DE ALGORITHM

### 1) EXPANSION FOR SEARCH SPACE

As is above mentioned, the evolutionary parameters have a great effect on the performance of DE algorithm. However, even with all those elaborate parametric manipulations, it is impossible to avoid the inherent evolutionary stagnation in DE entirely.

After three evolutionary operations introduced in section III-A, each individual will evolve into next generation and has superior evaluation, or it will be eliminated as an inferior individual. Such successful evolution requires a specific time, known as evolution stagnation. As the evolution continues, the whole population tends to be mature. Then the differences between the individuals become smaller and smaller. Consequently, the probability of successful evolution decreases with the increasing stability of the population.

To alleviate the effects of evolution stagnation, an improved fast differential evolution (IF-DE) algorithm has been proposed in this article. We extend the search space by utilizing those inferior parents, which further improve the performance of the algorithm, especially in the middle and later stage of evolution. Fig. 2 shows the two extended vectors in our algorithm. Then we give a detailed introduction of such two extended vectors in the following sections.

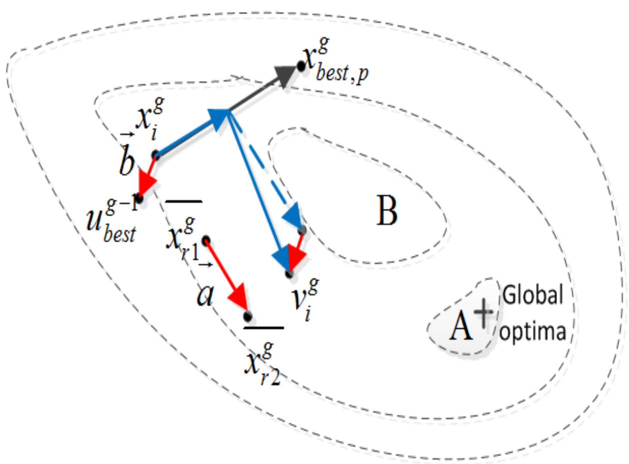


**FIGURE 2.** Two extended vectors directions in IF-DE.

#### a: INTRODUCTION OF INFERIOR PARENT INDIVIDUAL

In the middle and later stage of evolution, main concentration will be attached to local search to satisfy the accuracy of the solution. However, the tradeoff between the global search speed and the local search precision needs to be considered sensibly to avoid the evolutionary stagnation. JADE in [29] gives an excellent mutation operation, as is shown in (9):

$$v_i^g = x_i^g + f_{i1}(x_{best,p}^g - x_i^g) + f_{i2}(x_{r1}^g - x_{r2}^g), \tag{9}$$

where $x_{best,p}^g$ is selected form the $100p\%$ optimal individuals in current generation.

JADE selects one of $100p\%$ optimal individuals as $x_{best}^g$, which will maintain the diversity of population to the next generation. Moreover, the differential individuals $(x_{r1}^g, x_{r2}^g)$ are derived not only from the current population $|P|$, but also from part of the previous population, which is saved in set $|A|$. In JADE algorithm, the set $|A|$ is a collection of those inferior parent individuals in previous generation. However, such random selection from the union $|P \cup A|$ usually fails, with a high probability, in fully utilization of the information stored in set $|A|$.

Then we make some modifications in the selection of differential individuals, seen in formula (10):

$$v_i^g = x_i^g + f_{i1}(x_{best,p}^g - x_i^g) + f_{i2}(\overline{x}_{r1}^g - \overline{x}_{r2}^g),\ g > 0.5 \cdot g_{\max} \tag{10}$$

We divide the evolution process into two segments. In the first semi-period, we refer to same selection in JADE. In the second-half cycle, the individual $\overline{x}_{r1}^g$ is selected directly from the set $|A|$ and $\overline{x}_{r1}^g$ from the current population $|P|$.Such selection in IF-DE algorithm can make full advantages of the information from two consecutive generations, which can be adapt to the evolution dynamically. The extended vector, formed by two inferior parent individuals, is denoted as $\overrightarrow{a}$ in Fig. 2.

#### b: INTRODUCTION OF ELIMINATED TRIAL INDIVIDUAL

In order to further enrich the diversity of the population, now we consider to extend another search direction. Since the inferior parent individuals can be introduced into the population, the eliminated trial individuals can also be valued, if by elaborate control, and be employed into the population.

Obviously, according to the mechanism of DE algorithm, the eliminated trial individuals, who have inferior fitness values, need to be abandoned directly. However, in the later stage of evolution, all those trial individuals, mutated from the increasingly mature population, are provided with high quality as well. If the trial individuals are abandoned without consideration, all the valuable information carried inside them will be discarded.

It's worth noting that the introduction of trial individuals should not destroy the structure of the original solution. So we introduce the *p-best* eliminated trial individual from the previous generation. The designed mutation operation is shown as follows:

$$\begin{aligned} v_i^g = x_i^g &+ f_{i1}(x_{best,p}^g - x_i^g) + f_{i2}(\overline{x}_{r1}^g - \overline{x}_{r2}^g) \\ &+ f_{i3}(x_i^g - u_{best,p}^{g-1}), \quad g > 0.5 \cdot g_{\max}, \end{aligned} \tag{11}$$

where $(x_i^g - u_{best,p}^{g-1})$ is our second extended vector (denoted by $\overrightarrow{b}$ in Fig. 2), and $f_{i3}$ is an adaptive scalar parameter. From the figure, we can see that the extended vector $\overrightarrow{b}$ provides additional feedback information from the previous generation, assisting the individuals to achieve more successful evolution. What's more, Fig. 2 also shows that $\overrightarrow{b}$ avoids the local space exploitation by disturbing the direction of global search slightly. In the end, the proposed IF-DE algorithm has a faster speed to evolve and find the global optima.

### 2) PARAMETER ADAPTION

According to the mutation operation in (11), four adaptive parameters $f_{i1}, f_{i2}, f_{i3}$ and $CR_i$ need to be controlled.

 

Firstly, we adjust the parameters $f_{i1}, f_{i2}$ referring to the SPSO algorithm [22], which simulates the PSO information interaction.

$$f_{i1} = \frac{f(x^g_{best})}{f(x^g_i) + f(x^g_{best})}, \quad f_{i2} = \frac{\overline{f_i}}{\overline{f_i} + f(x^g_{best})}, \tag{12}$$

where $\overline{f_i} = (f(x^g_i) + f(\overline{x^g_{r1}}) + f(\overline{x^g_{r2}}))/3$. Similar to PSO, the adjustment not only refers to historical best individual, but also the current best individual. So the dynamic adjustment on $f_{i1}, f_{i2}$ achieves a better balance on global exploration and local exploitation in the later stage of evolution.

The crossover probability $CR_i$ is controlled linearly in formula (13).

$$CR_{i+1} = CR_i + (CR_0 - CR_{\min})/T, \tag{13}$$

where $T$ is the current iteration number of the algorithm. An appropriate value needs to be set for the initial $CR_0$. Because in the later stage of evolution, all the individuals tend to assimilate, so the crossover probability should keep an increasing trend. Storn shared several valuable experiences in [27] about the scope of parameters. For example, the initial $CR_0$ is usually set as 0.5, and $CR_i$ that lies in the interval of [0.6, 0.9] usually achieves a better performance during the whole evolution.

Lastly, we give an adaptive control on parameter $f_{i3}$. Two aspects need to be considered when we adjust the parameter. First of all, the value of $f_{i3}$ should not be too large to destroy the solutions' structure of maturing individuals. Secondly, $f_{i3}$ needs to be associated with the diversity of the whole population. Referring to [28], the following formula presents the adjustment of $f_{i3}$

$$f_{i3} = \frac{\lambda_{PC}}{\lambda_{PC} + \lambda_{FC}}, \tag{14}$$

where

$$\begin{cases} \lambda_{PC} = 1 - (1 + PC) \cdot e^{-PC} \\ \lambda_{FC} = 1 - (1 + PC) \cdot e^{-FC} \\ \lambda_{PC}, \quad \lambda_{FC} \in (0, 1). \end{cases} \tag{15}$$

The parameters $PC$ and $FC$ denote the information of diversity in current population, which is defined as

$$\begin{cases} PC = \sqrt{\dfrac{1}{|\mathrm{P}|} \displaystyle\sum_{i=1}^{|\mathrm{P}|} \sum_{j=1}^{D} (x^g_{j,i} - x^{g+1}_{j,i})^2} \\ FC = \sqrt{\dfrac{1}{|\mathrm{P}|} \displaystyle\sum_{i=1}^{|\mathrm{P}|} (f_i(x^g_i) - f_i(x^{g+1}_i))^2} \end{cases} \tag{16}$$

According to the conclusion in [28], the evolution enters into the later stage when $PC$ and $FC$ have small values and the main task of searching should concentrate on the local precision of the solution. At this moment, the introduced vector $\overrightarrow{b}$, however, further extend the global search. As a result, such adaptive adjustment achieves a balance between multiple performances of the algorithm.

## IV. TRACKING FRAMEWORK BASED ON IF-DE

### A. ESTABLISHMENT OF TARGET MODEL

Gaussian Mixture Model (GMM) has been widely used for object detecting among many research works. Essentially, GMM belongs to the method of frame difference, as is stated in Chapter I. The image is segmented into background and target in every frame and then the target can be detected directly. The corresponding description is as follows

$$D(x, y) = |F(x, y) - B(x, y)|, \tag{17}$$

where $F(x, y)$ and $B(x, y)$ denote the current image and the background image, respectively.

Then we give a brief introduction on GMM. It is built with multiple (assuming $M$) Gaussian distribution for a single pixel in the image. Theoretically, if $M$ is large enough, GMM can approximate any continuous probability density distribution.

The application of adaptive GMM in target tracking was first proposed by Stuaffer and Grimson [34], who utilized a weighed Gaussian distribution to approximate the gray value of the same pixel from the previous $k$ frames. During the process of tracking, we need to update the parameters of GMM dynamically to adapt to the complex and changeable tracking scenarios.

Referring to [35], we denote $M^k = \{W^k, S^k, F^k\}$ as a description of GMM at *k-th* frame. $W^k$ and $S^k$ represent the inner-frame variations and the stable component. And $F^k$ denotes a fixed template to stabilize the tracker. The three components contain the information of relative stability, mixed variance, and fixed template between two adjacent frames respectively. After the target at *k-th* frame is located, the GMM needs to be updated to adapt to the next frame image. We choose EM algorithm [36] to update the GMM. Due to space limitations, more details about EM algorithm to update the GMM parameters can be found in reference [36].

### B. TRACKING PROCESS BASED ON IF-DE

Based on the updated background model, the tracking algorithm then endeavors to locate the optimal position of the interested target.

As is summarized in Chapter I, lots of excellent tracking algorithms have been proposed for recent years. Among them, a typical PF algorithm is widely used to predict the state of the object. However, PF usually has a problem of sampling impoverishment. For this reason, Zhang *et al.* [22] proposed a SPSO algorithm to introduce more observations into the sampling, which can achieve a longer time tracking.

Our designed tracking algorithm based on IF-DE is similar to [22] but achieves better performance in tracking accuracy and speed than SPSO tracker. The comparison of this two trackers will be experimented in the next chapter. Firstly, we introduce the IF-DE based tracking process as shown in Fig. 3.

When the current *k-th* image is tracked, all the individuals converge to $x_{best,k}$. Therefore, it is necessary for the initial population to be redistributed in an appropriate for the

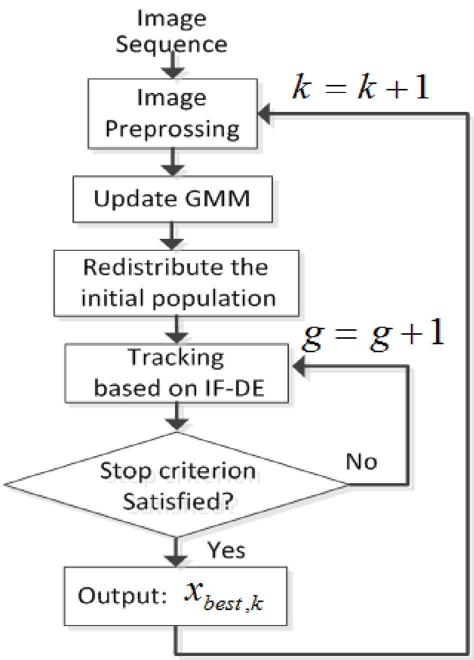


**FIGURE 3.** Tracking process based on IF-DE.

purpose of avoiding the loss of diversity in the next frame. By utilizing the prior experience of the target motiončwe guide the redistribution:

$$x^0_{best,k+1} \sim N(x_{best,k}, \sum), \tag{18}$$

where $\sum$ is a covariance matrix, whose diagonal elements are proportional to the predicted direction of the target, defined as:

$$d_k = x_{best,k-1} - x_{best,k-2} \tag{19}$$

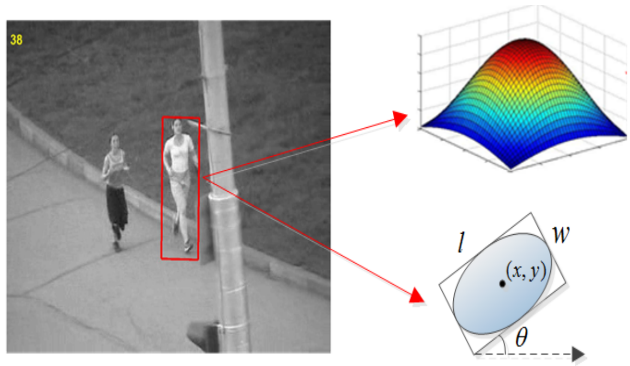


**FIGURE 4.** Dynamic external rectangular on target.

With above redistribution, the initial population can evolve to the global best individual $x_{best,k+1}$ with a higher speed. To be noted, the fitness value of each individual is calculated based on the GMM [22].

To demonstrate a observable tracking effects, the tracked target will be visually displayed by an external rectangular box timely, as is shown in Fig. 4. In this paper, the state of the calibration box is expressed by five parameters, where $(x, y)$ denotes the center of the box, while $l$, $w$ and $\theta$ denote the length, the width and the rotation angle, respectively. Each calibration box corresponds to an individual in the population.

For every continuous frame, the IF-DE based tracking algorithm, designed in this paper, endeavors to find the best calibration parameters, which map to the global best individual $x_{best}$, to locate the best map position of the target. In fact, the best individual of each generation can be a candidate for the position of the tracked target. To achieve a good balance between tracking accuracy and speed, we stop the current

**FIGURE 5.** Tracking performance on "Jumping" datasets.

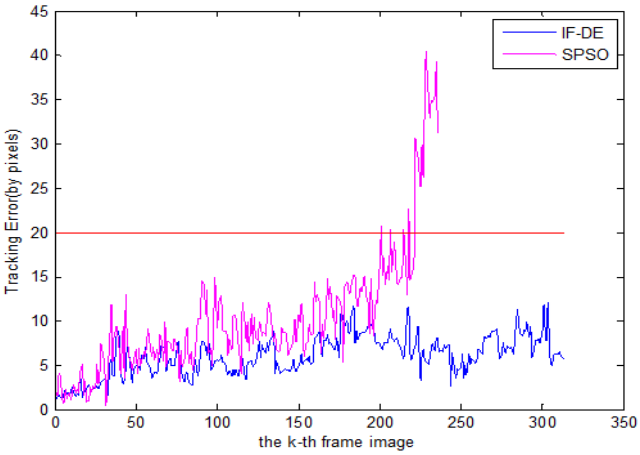


**FIGURE 6.** Tracking accuracy on "Jumping" datasets.

frame tracking when the threshold of tracking error or the max generation is satisfied.

## V. EXPERIMENTAL RESULTS AND ANALYSIS

We conduct several experiments to verify the effectiveness of IF-DE based tracking scheme. Several datasets in literature [37] are selected for performance verification and analysis, which contain several typical tracking scenarios, such as Fast Motion (FM), Illumination Variation (IV), Deformation (DEF), Occlusion (OCC), Background Clutters (BC), Out-of-View (OV) and so on.

The selected datasets in this article are shown in Table I.

Firstly, we conduct an experiment on the dataset "Jumping", comparing with the PF tracker in [37] and SPSO tracker in [22]. Comparisons on the tracking performance are shown in Fig. 5. The first and the second row correspond to the PF tracker and SPSO tracker respectively. Our IF-DE based tracking results are shown in the third row in Fig. 5.

**TABLE 1.** Tracking datasets.

| Name | Jumping | Surfer | Jogging | Trellis |
|---|---|---|---|---|
| Size | 352*288 | 480*360 | 352*288 | 320*240 |
| Frames | 313 | 376 | 307 | 569 |
| Scenarios | FM,MB,BC | FM,MB,DEF | MB,OCC | IV,DEF |

From this picture, we can see that the boy in the "Jumping" dataset jumps up and down quickly. The abrupt motion and background clutters bring a great challenge to the tracking algorithm. The PF tracker loses the target at the later frames when the boy's face undergoes abrupt motion blur, which accounts for the sample impoverishment in PF. And both the SPSO tracker and our tracking algorithm track the target successfully. However, during a long time tracking, the tracking window of SPSO tracker freezes at a specific position, resulting in unendurable tracking accuracy or even failure in tracking. This is due to the fact that most particles in SPSO algorithm converge to local optimum during the tracking process, which further limits the ability of long time tracking. Compared to the SPSO tracker, the proposed IF-DE algorithm consumes less time for tracking the "boy" successfully during the whole image sequences. Fig. 6 gives the comparison of tracking accuracy in this two tracking algorithm.

In our experiment, tracking accuracy is evaluated by a quantitative evaluation of MSE. The error is defined as the difference between the tracked location and the real location called Ground Truth in each frame. From Fig. 6 we can see that our IF-DE based tracking algorithm has a higher tracking accuracy. Theoretically, the introduction of inferior individuals in IF-DE further diversifying the population, which can

**FIGURE 7.** Tracking results on "Surfer", "Jogging" and "Trellis" datasets.

proceed more effective search for the global optima. Then the best position of the target will be located with less time for our IF-DE tracking algorithm.

In order to fully demonstrate the performance of the proposed tracking algorithm, we evaluate it with another three datasets in Table I, which contains tougher tracking scenarios. Experimental results are displayed in Fig. 7.

From the tracking results in Fig. 7, we can see that the surfer has arbitrary motions in the ocean and the jogger is partial occluded by a telegraph pole. The boy named "Trellis" undergoes appearance changes and illumination variations in his house. The proposed IF-DE based tracking algorithm still successfully tracks the interested target in above tough tracking scenarios.

## VI. CONCLUSIONS

In this paper, we proposed an improved fast differential evolution algorithm and introduced it into the application of target tracking. Firstly, we improve the traditional differential algorithm by fully exploiting the information from those inferior parent and trial individuals. The utilization of those inferior individuals further diversifying the population and speeds up the offspring's evolution. Secondly, to demonstrate the capability of the new algorithm, we integrated it with the application of target tracking. The experimental results under several challenging tracking scenarios verify that the IF-DE tracking algorithm has advantages in both tracking effectiveness and robustness compared with other similar methods.

## REFERENCES


[1] W. Chantara and Y.-S. Ho, "Object detection based on fast template matching through adaptive partition search," in *Proc. Int. Joint Conf. Comput. Sci. Softw. Eng.*, 2015, pp. 1–6.

[2] A. Banharnsakun and S. Tanathong, "Object detection based on template matching through use of best-so-far ABC," *Comput. Intell. Neurosci.*, vol. 2014, Apr. 2014, Art. no. 919406.

[3] A. Sibiryakov, "Fast and high-performance template matching method," in *Proc. Comput. Vis. Pattern Recognit.*, 2011, pp. 1417–1424.

[4] D. T. Nguyen, P. O. Ogunbona, and W. Li, "A novel shape-based non-redundant local binary pattern descriptor for object detection," *Pattern Recognit.*, vol. 46, no. 5, pp. 1485–1500, 2013.

[5] A. Jungmann and B. Kleinjohann, "Automatic feature classification for object detection based on motion analysis," in *Proc. Int. Conf. Autom., Robot. Appl.*, Dec. 2011, vol. 46. no. 5, pp. 190–195.

[6] Y. Guo, F. Sohel, M. Bennamoun, M. Lu, and J. Wan, "Rotational projection statistics for 3D local surface description and object recognition," *Int. J. Comput. Vis.*, vol. 105, no. 1, pp. 63–86, 2013.

[7] D. A. Klein and S. Frintrop, "Center-surround divergence of feature statistics for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, Nov. 2012, pp. 2214–2219.

[8] X. Gao, Y. Su, X. Li, and D. Tao, "A review of active appearance models," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 40, no. 2, pp. 145–158, Mar. 2010.

[9] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *Proc. Comput. Vis. Pattern Recognit.*, Jun. 1998, pp. 232–237.

[10] A. Opelt, A. Pinz, and A. Zisserman, "A boundary-fragment-model for object detection," in *Proc. Eur. Conf. Comput. Vis.*, 2006, pp. 575–588.

[11] C. Yang, R. Duraiswami, and L. Davis, "Fast multiple object tracking via a hierarchical particle filter," in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, 2005, pp. 212–219.

[12] A. Azarbayejani and A. P. Pentland, "Recursive estimation of motion, structure, and focal length," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 6, pp. 562–575, Jun. 1995.

[13] C. Bao, Y. Wu, H. Ling, and H. Ji, "Real time robust L1 tracker using accelerated proximal gradient approach," in *Proc. Comput. Vis. Pattern Recognit.*, 2012, pp. 1830–1837.

[14] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 263–270.

[15] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. Comput. Vis. Pattern Recognit.*, 2009, pp. 983–990.

[16] D. Wang, H. Lu, and C. Bo, "Fast and robust object tracking via probability continuous outlier model," *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 5166–5176, Dec. 2015.

[17] P. Pérez, C. Hue, J. Vermaak, and M. Gangnet, "Color-based probabilistic tracking," in *Proc. 7th Eur. Conf. Comput. Vis.*, 2002, pp. 661–675.

[18] M. Bray, E. Koller-Meier, N. N. Schraudolph, and L. Van Gool, "Fast stochastic optimization for articulated structure tracking," *Image Vis. Comput.*, vol. 25, no. 3, pp. 352–364, 2007.

[19] S. Chen, S. Zhu, and Y. Yan, "Robust visual tracking via online semi-supervised co-boosting," *Multimedia Syst.*, vol. 22, no. 3, pp. 297–313, 2016.

[20] B. Yang and R. Nevatia, "An online learned CRF model for multi-target tracking," in *Proc. Comput. Vis. Pattern Recognit.*, 2012, pp. 2034–2041.

[21] D. A. Ross, J. Lim, R.-S. Lin, and M.-H. Yang, "Incremental learning for robust visual tracking," *Int. J. Comput. Vis.*, vol. 77, nos. 1–3, pp. 125–141, 2008.

[22] X. Zhang, W. Hu, S. Maybank, X. Li, and M. Zhu, "Sequential particle swarm optimization for visual tracking," in *Proc. Comput. Vis. Pattern Recognit.*, 2008, pp. 521–531.

[23] Y. Oshman and A. Carmi, "Attitude estimation from vector observations using a genetic-algorithm-embedded quaternion particle filter," *J. Guid., Control, Dyn.*, vol. 29, no. 4, pp. 879–891, 2012.

[24] B. Sun, B. Wang, Y. Shi, and H. Gao, "Visual tracking using quantum-behaved particle swarm optimization," in *Proc. Chin. Control Conf.*, 2015, pp. 3844–3851.

[25] W. Wu, B. Wang, Y. Zeng, H. Zhang, Z. Yang, and Z. Deng, "Robust secure beamforming for wireless powered full-duplex systems with self-energy recycling," *IEEE Trans. Veh. Technol.*, vol. 66, no. 11, pp. 10055–10069, Nov. 2017.

[26] W. Wu and B. Wang, "Efficient transmission solutions for MIMO wiretap channels with SWIPT," *IEEE Commun. Lett.*, vol. 19, no. 9, pp. 1548–1551, Sep. 2015.

[27] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces," *J. Global Optim.*, vol. 11, no. 4, pp. 341–359, 1997.

[28] A. K. Qin and P. N. Suganthan, "Self-adaptive differential evolution algorithm for numerical optimization," in *Proc. IEEE Congr. Evol. Comput.*, vol. 2. 2005, pp. 1785–1791.

[29] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.

[30] J. Liu and J. Lampinen, "A fuzzy adaptive differential evolution algorithm," *Soft Comput.*, vol. 9, no. 6, pp. 448–462, 2005.

[31] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential evolution algorithm with strategy adaptation for global numerical optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009.

[32] R. Mallipeddi, P. N. Suganthan, Q. K. Pan, and M. F. Tasgetiren, "Differential evolution algorithm with ensemble of parameters and mutation strategies," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 1679–1696, 2011.

[33] Y. Wang, Z. Cai, and Q. Zhang, "Differential evolution with composite trial vector generation strategies and control parameters," *IEEE Trans. Evol. Comput.*, vol. 15, no. 1, pp. 55–66, Feb. 2011.

[34] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proc. Comput. Vis. Pattern Recognit.*, vol. 2. 1999, pp. 246–252.

[35] S. K. Zhou, R. Chellappa, and B. Moghaddam, "Visual tracking and recognition using appearance-adaptive models in particle filters," *IEEE Trans. Image Process.*, vol. 13, no. 11, pp. 1491–1506, Nov. 2004.

[36] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. Roy. Stat. Soc. B, Methodol.*, vol. 39, no. 1, pp. 1–38, 1977.

[37] Y. Wu, J. Lim, and M. H. Yang, "Online object tracking: A benchmark," in *Proc. Comput. Vis. Pattern Recognit.*, 2013, pp. 2411–2418.

**LAIJIE LIN** received the B.S. degree in semiconductor physics and device from Xidian University, Xi'an, China, in 1991, and the M.S. degree in control engineering from the Wuhan University of Technology, Wuhan, China, in 2007. He is currently with the Wenzhou Vocational and Technical College. His research interests include computer vision, image processing, and artificial intelligence.

**MIN ZHU** received the B.S. degree from the Nanjing University of Posts and Telecommunications, China, in 2002, and the M.S. degree from the Beijing University of Posts and Telecommunications, China, in 2005. She is currently pursuing the Ph.D. degree with the Nanjing University of Posts and Telecommunications. Her research interests mainly include routing protocol and optimization algorithm design.