IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 5, 2018, accepted August 8, 2018, date of publication September 13, 2018, date of current version October 8, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2869869

Binary Artificial Immune Algorithm for Adaptive Visual Detection

WEI GAO[1], JINLING ZHANG[2], QIAOYUAN LIU[2], LONGKUI JIANG[3], YURU WANG[2], MINGHAO YIN[2], AND YUPENG ZHOU[2]

[1]School of Astronautics, Beihang University, Beijing 100083, China
[2]Computer Science and Information Technology, North-East Normal University, Changchun 130024, China
[3]School of Information Engineering, Jilin Business and Technology College, Changchun 130062, China

Corresponding author: Yuru Wang (wangyr915@nenu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61300099, in part by the China Postdoctoral Science Foundation Funded Project under Grant 2015M570261, in part by the Science and Technology Development Plan of Jilin Province under Grant 20170101144JC, in part by the Open Fund of China key Laboratory of Symbolic Computation and Knowledge Engineering of Ministry of Education under Grant 93K172016K14, and in part by the Fundamental Research Funds for the Central Universities under Grant 2412017FZ027.

ABSTRACT A visual model plays an important role in developing an efficient and robust visual tracker. The visual cues employed in the state-of-the-art models are usually predefined and fixed for all the tested videos. However, the discriminative ability of features usually varies among videos. Therefore, using a fixed set is both redundant and noisy: only a subset of any fixed set will present distinct profiles for modeling. Thus, selecting a highly discriminative cue subset in visual modeling should improve the tracking accuracy. In this paper, an optimization method based on a binary artificial immune algorithm is proposed that selects an effective, discriminative feature subset that is adaptive to specific videos. Specifically, a metric is defined to measure the discriminative abilities of visual models. Then, the visual modeling problem is transformed into an optimization scheme, and a binary artificial immune algorithm is introduced and specially designed to solve the modeling problem. Moreover, to preserve the subset of visual cues that are most adaptive to the tracking condition, the selected cues are assigned adaptive weights and modeled in a Sequential Monte Carlo framework. To show its effectiveness, the proposed algorithm is tested on ten representative videos. The experimental results demonstrate the improvement in tracking performance, the improved tracker performs better or comparable with previous excellent trackers from the literature.

INDEX TERMS Artificial immune algorithm, binary, visual model, tracking.

I. INTRODUCTION

Modeling the appearance of a target is a key problem in visual tracking; thus, developing a discriminative visual model has received much attention from researchers. The model is a central issue of tracking because its performance fundamentally determines the robustness and stability of the tracking system. Complex tracking conditions, deformations in the target's appearance, and nebulous boundary areas between the target and its background all pose great challenges to a robust tracker. The goal of visual modeling is to develop a discriminative representation model that helps to easily identify a target from its background.

To achieve the above goal, an initial effort is to find various types of representative visual features such as color [1], edge, texture, and motion [2]. These features have performed well in many applications and have been well developed. Recently, the feature extraction methods most commonly used for these three visual cues are HSV [3], HOG [4], and LBP [5]. Satisfactory results have been achieved in prior studies utilizing these methods, for example, in [7]–[9].

However, given the requirements of many complex applications, these single-cue-based methods are inefficient. The appearance of non-rigid targets always varies in videos and is usually affected by background changes such as illumination, occlusion and motion. In such cases, the single-cue-based models are incapable of coping with the variations in the target's visual appearance. A typical solution adopted by recent existing works is to integrate multiple cues into an integrated model in which multiple cues are weighted and summed [10] into a hierarchical model [11], a Gaussian mixture model [12] or an HMM model [13]. Integrating multiple cues into a single model results in better performance because

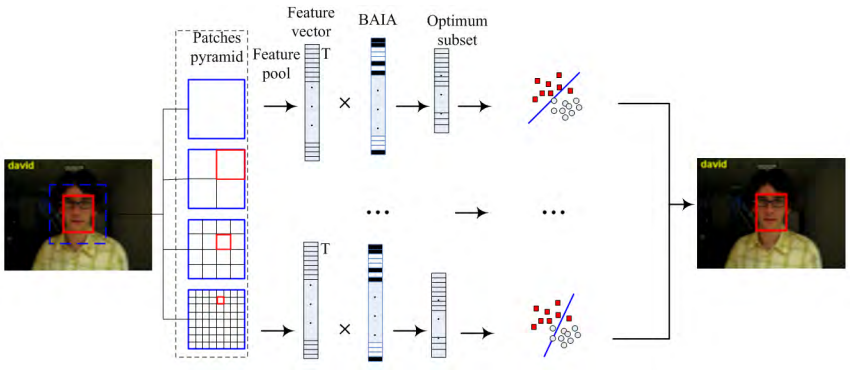

FIGURE 1. shows the overall framework of the proposed optimized adaptive visual model. Note that, the feature vectors should be row vectors.

some of the cues have high discriminative ability for some specific video frames but shows decreased discriminative ability when the conditions change; in such cases, other cues then play a more important role. That is, employing a larger set of cues allows them to compensate for one another over a tracking sequence. This type of model has been successfully employed in existing trackers [14]. However, in all the experiments with these applications, the integration models are fixed for all the tested videos, which is inefficient in many real applications?

For visual models, deep learning has become an important research topic in recent years and shows good performance [15]–[17]. In particular, in the last two years, deep learning has received increased attention: deep learning methods have been reported in many works [15]–[22] and journals [25], [26] and have achieved great success. However, it is unknown which parts of the network and which parameters play more important roles. It is also difficult to provide clear explanations about which features are more important for a specific video. Nevertheless, some excellent works have optimized trackers using other approaches [23], [24].

In this paper, we develop an effective visual model that can be controlled and explained. We propose a novel scheme for building an effective adaptive visual model, called the Binary Artificial Immune Algorithm (BAIA) that addresses the above challenges under an optimization and online updated framework.

The first contribution of this work is a novel resolution to model adaptiveness, which transforms the typical modeling problem into an optimization problem. The second contribution is the introduction of an evolutionary algorithm into the tracking problem; here, we leverage the artificial immune algorithm to select the best subset of features from the feature pool. The artificial immune algorithm is chosen to solve the optimization problem Because it provides diversity; in the algorithm, the cloning and variation of antibodies are helpful in producing new antibodies. In addition, the artificial immune algorithm guarantees convergence and converges quickly, reducing the time required to find the optimal solution. Thus, this algorithm is suitable for optimizing the visual model. This scheme guarantees the discriminative ability of the specific selected visual cues. Furthermore, for the two contributions mentioned above, we also propose integrating the selected feature subsets into a sequential Monte Carlo framework, which ensures that the specifically defined visual model can adapt to variable tracking conditions. Figure 1 shows the overall framework of the proposed BAIA scheme-based visual model. We conduct extensive experiments on a widely used benchmark [27] and demonstrate the superior performance of the proposed method over several excellent state-of-the-art methods, including TLD [28], Struck [29], VTD [30], LOT [31], LSHT [32], CN [33], FCT [34] and KCF [35].

The remainder of this paper is organized as follows: In Section 2, we define the adaptive visual model as an optimization problem. Then, in Section 3, this optimization problem is solved by an artificial immune algorithm. The proposed approach is presented in depth. Section 4 presents the results of extensive experiments conducted on benchmark datasets. Finally, conclusions are drawn in Section 5, and future work is discussed.

II. PROBLEM DEFINITION

A. PROBLEM DESCRIPTION

The importance of using a visual model in tracking problems does not have to be reiterated; finding a method to build a visual model with sufficient discriminative ability is the main concern of this work. To resolve this problem, we must clearly outline the challenges and define the concept of discriminative ability. Specific video frames in Figure 2 are presented to address the difficulties. In the first image Figure 2(a), the target of interest appears visually similar to its background. In the second image Figure 2(b), the boundary between the target and background is unclear. In the third video frame, the target is heavily occluded and shows an incomplete appearance. From these three examples, we can see that the main modeling task is to distinguish the target from the background and that one challenge is to avoid confusing the target with its background. From this point of view, for the second challenge, discriminative ability can be defined

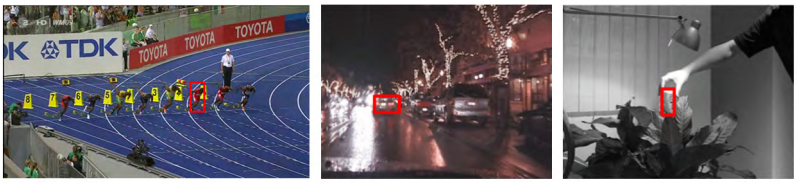

FIGURE 2. Challenges in visual modeling problem:(a)the background interference with similar appearance, (b)confused boundary, and (c) occlusion.

as the distinguish ability degree of the two classes (target and background) in the projected feature space.

The pixels of these two classes can be projected in a feature space when employing a specific feature extraction method. The object detection task is to find the boundary for classification between the two classes. For the abovementioned difficulties, this boundary is difficult to define because the two classes are easily confused. Consequently, a more discriminative visual model provides a clearer boundary between the two classes. That is, the projected points observed by a visual model with sufficient discriminative ability will be easy to classify. For a given video, the most adaptive visual model must exist. Thus, the problem becomes an optimization problem in which the key issue is how to describe ''easy to classify''.

B. OPTIMIZATION PROBLEM DEFINITION

To provide an objective description of the distribution of foreground and its background, in this paper, we propose a discrete sampling-based method. Specifically, some random samples are generated from a uniform distribution $U(a, b)$ as follows:

$$x_i = x_g + \Delta x, \quad \Delta x \sim U(a, b) \tag{1}$$

where x_g is the real state, and $D = \{x_i, y_i\}_{i=1}^{m}, y_i \in \{0, 1\}$ is the generated sample set. Observed from a specific visual model, these samples will show distinguishable similarities(weights) in the observation space.Specifically, the foreground and background samples have comparatively higher and lower weights, respectively. A visual model that preserves a greater discriminative ability will result in a larger discriminative degree between the sample weights. Based on the idea of LDA [36], we describe the above visual modeling problem as one of maximizing the following generalized Rayleigh quotient:

$$J = \frac{w^T S_b w}{w^T S_w w} \tag{2}$$

where S_b and S_w represent the between-class scatter matrix and the within-class scatter matrix, respectively:

$$S_b = (\mu_o - \mu_b)(\mu_o - \mu_b)^T \tag{3}$$

$$S_w = \Sigma_o + \Sigma_b = \Sigma_{x \in X_o}(x - \mu_o)^T + \Sigma_{x \in X_b}(x - \mu_b)(x - \mu_b)T \tag{4}$$

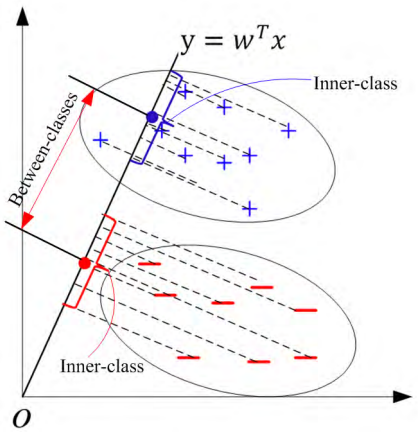

FIGURE 3. The 2D diagram of the discriminative ability measurement. In this figure, "+" and "−" represent the positive and negative samples, the ellipses represent the outer contour of the sample sets, and the blue and red circles are the projected center on the line w.

Assume that the generated sample set is $D = \{x_i, y_i\}_{i=1}^{m}, y_i \in \{0, 1\}$, where, X_i, μ_i, Σ_i represent the sample set, mean value and covariance matrix of the class $i \in o, b$ (the symbol of o represents for the object and b for the background), respectively. If all samples are projected onto a line w, then the projections of the centers of the two classes will be $w^T \mu_o$ and $w^T \mu_b$, the covariance of the two classes will be $w^T \Sigma_o w$ and $w^T \Sigma_b w$ [37].

As shown in Figure 3, for an excellent visual model, the projected points of the samples in the same class should be closer, and the points in the different classes should be farther apart.

III. BINARY ARTIFICIAL IMMUNE ALGORITHM BASED MODEL OPTIMIZATION

Feature selection is important in pattern recognition and machine learning applications such as gene classification, protein prediction, and text classification. Cheng *et al.* [38] proposed a Fisher-Markov selector to identify the most useful features for describing essential differences among the possible populations. Li *et al.* [39] proposed a multiobjective biogeography-based optimization method to select a small subset of informative gene-expression data. Evolutionary algorithms have also shown great success in many other

fields; however, to the best of our knowledge, they have not been applied in the visual tracking field.

Based on the above analysis, it should be possible to build the best visual model for a given feature set. Through the above objective optimization function, we are able to build a visual model with sufficient discriminative ability. The remaining task is to develop an effective feature selection method.

We take the ensemble tracking framework [40] as an example and implement an optimized visual model on it. The objective function defined in the previous section is general and easily generalized to a similar visual model.

A. ENSEMBLE TRACKING FRAMEWORK

In the ensemble tracker, N weak classifiers are weighted and summed as a strong one as follows:

$$H(x) = \Sigma_{n=1}^{N} \alpha_n \cdot h_n(x) \tag{5}$$

Similar discriminative model-based methods have have achieved considerable success in computer vision. These models include AdaBoost [43], [44] and ensemble tracking [45], [46]. For each video frame t, the weights α_n of the weak hypotheses are denoted as a vector, V. In the sequentially arriving datasets, the detection windows are represented as a pyramid patch model. In our design, each weak classifier is assigned to one patch in the pyramid visual model, as mentioned in the next section. At time t, given the input data x, the task of the tracker is to predict its label, y.

$$y = \begin{cases} +1, & H(x) \geq \tau \\ -1, & otherwise \end{cases} \tag{6}$$

where τ is a threshold controlling the positive and negative labels.

B. SAMPLE REPRESENTATION

The pyramid patch-based visual model [47] is employed to generate weak hypotheses. Each patch is represented by the proposed optimized feature vector and is assigned one weak classifier. As shown in Figure 1, the candidate region is represented as a pyramid-like set of patches. Overall, the detection region is divided into $n \times n$ uniform patches. In addition, larger patches that cover different parts of the target are also modeled by evenly dividing the detection region into 4×4, 2×2, and 1×1 regions. Both global and local features are extracted in this pyramid model. Therefore, the model will be robust against uniform changes in the target's appearance.

We extract the discriminative features from each patch in the pyramid model to form feature vectors. Similar to other state-of-the-art trackers, we collect effective tracking features such as HSV, HOG, and LBP. The HSV color space contains three components that respectively represent hue, saturation, and value. The HSV color system more closely approximates human color perception than does the RGB system and comprises a type of color histogram. When extracting the HSV histogram from the image, we first convert the input RGB image into the HSV color space; then, we form the HSV histogram feature vector by quantizing the HSV spatial feature into a 256-dimensional feature histogram. The HOG features mainly describe a local region and the image features are formed by calculating a histogram of the local gradient direction. The LBP operator is defined as a window with a size of 3×3: the center pixel of the window is used as a threshold, and the gray values of the 8 surrounding pixels are compared. When the value of a surrounding pixel is greater than the value of the central pixel, that pixel is assigned a 1; otherwise, it is assigned a 0. In this way, the 8 points in the 3×3 neighborhood can be compared to produce an 8-bit binary number (usually converted to a decimal number, i.e., LBP code, a total of 256). That is, this procedure obtains the LBP value of the center pixel of the window, which is used to reflect the texture information of the area. For generalization, the feature can be extended in other applications.

C. MODEL OPTIMIZATION AND UPDATING

Many effective evolutionary algorithms exist [48], [49] for solving optimization problems. The theory of the immune network was proposed in 1974 by Jerne [50], and an artificial immune system (AIS) was clearly defined in 1996. The artificial immune algorithm (AIA) achieves the antigen recognition, cell differentiation, memory and self-regulating functions of the biological immune system by imitating the immune system of the human body. AIA is widely used in many optimization-related problems because it has diversity, guarantees convergence, and converges quickly. In addition, the algorithm is both more efficient and has less degeneration compared with other evolutionary algorithms such as the genetic algorithm (GA), ant colony, and artificial bee colony algorithms. For specific problems, the algorithm complexity is different. Binary artificial immune algorithm can converge quickly, which means less time complexity in this problem.The time complexity of BAIA is O(n).

An ''antigen'' corresponds to the objective function and its constraint conditions. Each possible solution is called a ''B-cell'' or ''antibody'' with an affinity and is represented by an n-dimensional real vector. The initial individual B-cell vectors are randomly generated. The best solutions are considered to be B- cells with greater affinity, while poor solutions are B-cells with less affinity. In AIA, a B-cell is a vector that follows cloning and mutation steps to reach the optimal solution. A new candidate B-cell is generated from all the solutions through clone and mutation operators.

In AIA, the cloning strategy is used to change the existing solutions. B-cells with greater affinity are cloned to generate a new B-cell population. Mutation is a probabilistic operator that randomly modifies B-cells at a certain mutation rate. The mutation strategy simulates the characteristics of a supermutation during the B-cell cloning procedure in the immune response.

Based on the feature pool, feature selection is achieved through a binary feature pool procedure, and the selection result is a binary vector in which 0 and 1 respectively

Algorithm 1 Binary Artificial Immune Algorithm Based Optimization(BAIA)

Input: The antigen:the objective function and the constrain
Output: The optimum B-cell:the optimum solution

function BAIA
  initialize the B-cellpopulation A of size N
  for each iteration:
  for i = 0 to N-1 **do**
    Affinity_A[1] = calcuAffinity(A[i])
  end for
  Sort A in ascending order of Affinity_A
  for i = 0 to n-1(n<N) **do**
    A_Clone[i] = A[i]
  end for
  newp_m = random(0,1)
  for i = 0 to n-1 **do**
    if new$p_m > p_m$ **then**
      A_Mutation[i][new$p_m \times n$] =
      !A_Clone[i][new$p_m \times n$]
    end if
    Affinity_A_Mutation[i] = calcuAffinity(A_Mutation)
  end for
  Sort A_Mutation in ascending order of Affinity_A_Mutation
  for i = 0 to $(1/2) \times (n-1)$ **do**
    A_Final[i] = A_Mutation[i]
  end for
  for i = $(1/2) \times (n-1)$ to N-1 **do**
    A_Final[i] = A[i-$(1/2) \times n$]
  end for
  A = A_Final
end function

represent selecting or not selecting a specific feature. The optimization problem is described as follows:

$$\begin{aligned} &Maximizing:\ J = \frac{w^T S_b w}{w^T S_w w} \\ &subject\ to\ x \in \Omega \end{aligned} \tag{7}$$

where, Ω is the decision space. $x = (x_1, x_2, \ldots, x_D) \in \Omega$ is a D-dimensional decision variable vector. The algorithm details is presented in Algorithm 1.

In our experiments, the affinity increases as the iteration steps increase, and the mutation ratio decreases with the increase in B-cell affinity. Therefore, the mutation probability will decrease to a certain degree (e.g., 1%). In the first frame of the video, the above algorithm is able to generate a feature set with optimum discrimination ability on the given feature pool. As tracking progresses, the tracker should be updated to remain adaptive to the variable tracking conditions. In the proposed tracker, the ensemble parameters relevant to the visual cues are modeled in a sequential Monte Carlo framework as in [40].

D. THE BINARY FEATURE SELECTION MECHANISM

After the above algorithm description, we find that for a specific video sequence, the algorithm yields an optimal solution for feature selection. The optimal solution is formatted as a binary string and an affinity value. The affinity value is used to describe the discriminative ability of the optimal solution.

In recent research, binary representations have numerous applications in machine learning and feature extraction [41], [42]. In this paper, ''binary'' means that the individual in the algorithm is expressed in the form of a binary string, which conforms to both the description of the individual in the algorithm and the operators of the algorithm. In each individual, 0 represents elimination and 1 represents selection. Figure 5 shows an example of using binary entities to filter features. The first vector is the initial extracted feature vector; the middle sequence is the optimal solution; the third is the selected feature subset. The three colors in the graph refer to the three types of features.

IV. EXPERIMENTS AND DISCUSSION

In many real applications, the videos to be analyzed depict a variety of situations; thus, using a uniform visual model for all situations is unreliable. The proposed solution in this paper resolves the problem by learning an adaptive visual model and transforming the visual model problem into an optimization problem. Our goal is to select features that have a high discriminative ability for a certain video from a given feature pool. In this study, ten representative videos from the OTB 2013 Benchmark [27] dataset are used in the experiments. The videos include common tracking challenges such as abrupt motion, severe occlusions, complex backgrounds, illumination variations, and deformations.

The proposed method was implemented in the MATLAB development environment by extending the code originally provided by Bai [40]. All the experiments were executed on a computer with a 3.4 GHZ processor and 4 GB of memory. In all the experiments, tracking accuracy was measured by three metrics: ACLE (Average Center Location Error), AOR (Average Overlap Ratio) and OPE (one-pass evaluation).

A. PARAMETER SETTINGS

The main controlling parameters of the algorithm include the encoding ratio, iterations, and the mutation probability. The encoding ratio is the ratio of the selected subset to the feature pool size, which determines the size of the selected subset. The convergence of the artificial immune algorithm is determined by the number of iterations; therefore, its setting directly determines the optimization performance. The mutation probability is another factor that controls the optimization speed: a too-small ratio leads to slow convergence, and a too-large ratio leads to ''hopping'' of the solutions. Figure 4 shows curves that illustrate the influence of the three parameters on the tracking accuracy. All the curves were created using the video ''boy'' as an example. The resulting optimal parameter settings are subsequently employed for

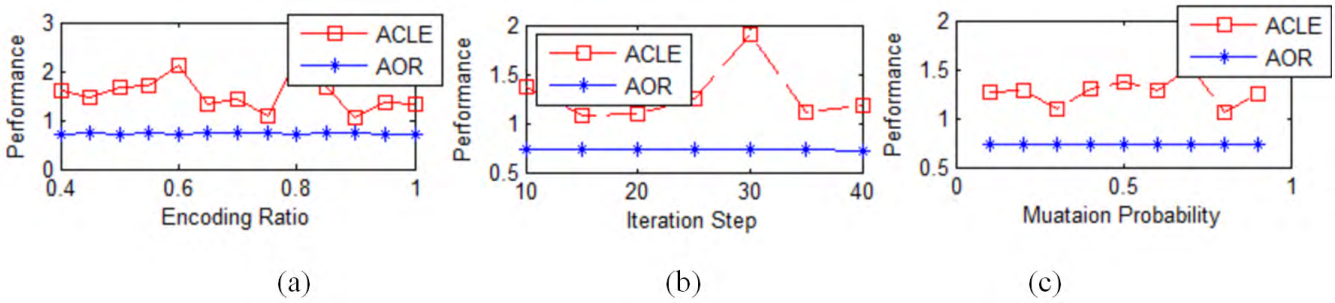

FIGURE 4. The influence curves of the parameters on the tracking performance, and the controlling parameters includes (a) Encoding ratio, (b) Iteration step, and (c) Mutation Probability.

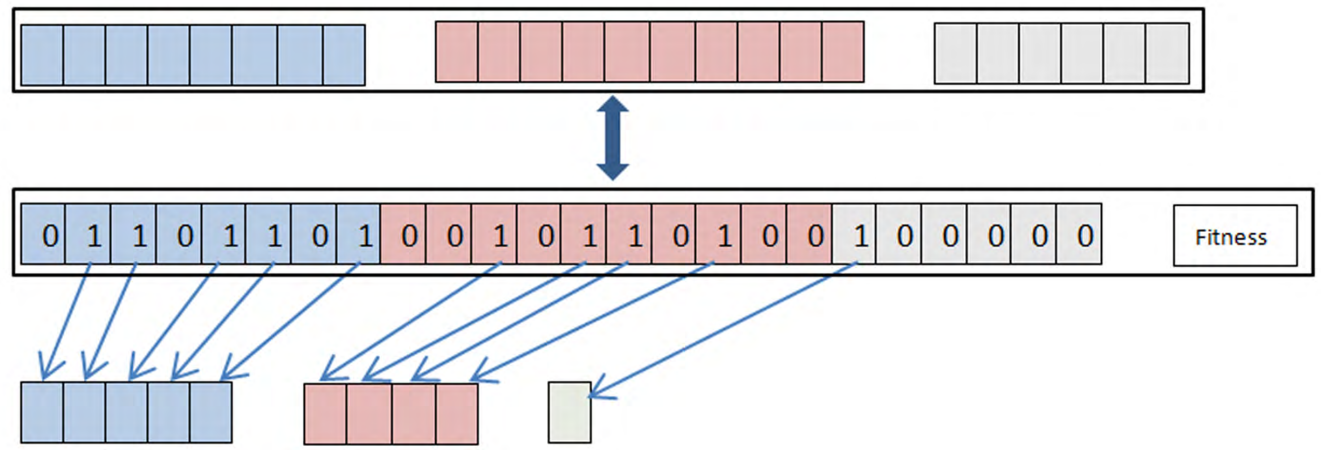

FIGURE 5. The Binary feature selection mechanism.

the other test videos. The curves show that the parameters have less influence on the AOR and more influence on the ACLE. Specifically, when the encoding ratio is set to 0.75 or 0.9, the tracker achieves the lowest ACLE value and the highest AOR value (although the difference is small). When the encoding ratio is set to 0.8, 0.85, 0.95 or 1, the ACLE increases. These results demonstrate that some features are redundant to the tracking performance and should be removed from the visual model. Between the 0.75 and 0.9 values, we choose 0.75 as the encoding ratio to improve the efficiency. The tracker shows better performance when the number of iterations is set between 15 and 20. The tracker obtains the best performance when the mutation probability is set to 0.3 with a 0.01 decrease at each iteration step. In the subsequent experiments, the encoding ratio, iterations, and mutation probability are set to 0.75, 20, and 0.3, respectively, with a 0.01 decrease at each iteration. Overall, the tracking performance is not particularly sensitive to the parameters. Similar results can be obtained using the ''Girl'' as a test video, although it performs slightly worse as an example.

B. COMPARISON AND DISCUSSES

This paper focuses on developing a feature selection strategy to improve the visual tracking performance. Therefore, we first transform the problem of visual modeling into an optimization problem and then use an artificial immune algorithm to solve it. Thus, the main concern of this paper is the feature selection strategy. In contrast, the optimization algorithm selected here is only one possible solution: many optimization algorithms such as the genetic algorithm or the bee colony algorithm could be used to solve the optimization problem. A large number of studies have investigated the computational complexity and effectiveness of these algorithms. The selected artificial immune method requires only approximately 20 iterations to achieve our feature selection goal. The algorithm is not required to converge to a certain accuracy; therefore, a discussion of different optimization methods is of little significance to the feature selection problem. Therefore, we do not compare the selected algorithm with others in this paper; instead, we compare our method with several state-of-the-art discriminative tracking models, including the following: TLD [28], Struck [29], VTD [30], LOT [31], LSHT [32], CN [33], FCT [34] and KCF [35]. In addition, the proposed method is also compared by applying the tracker without a feature selection step. These specific trackers were selected because they perform well, are popular and their code is open source.

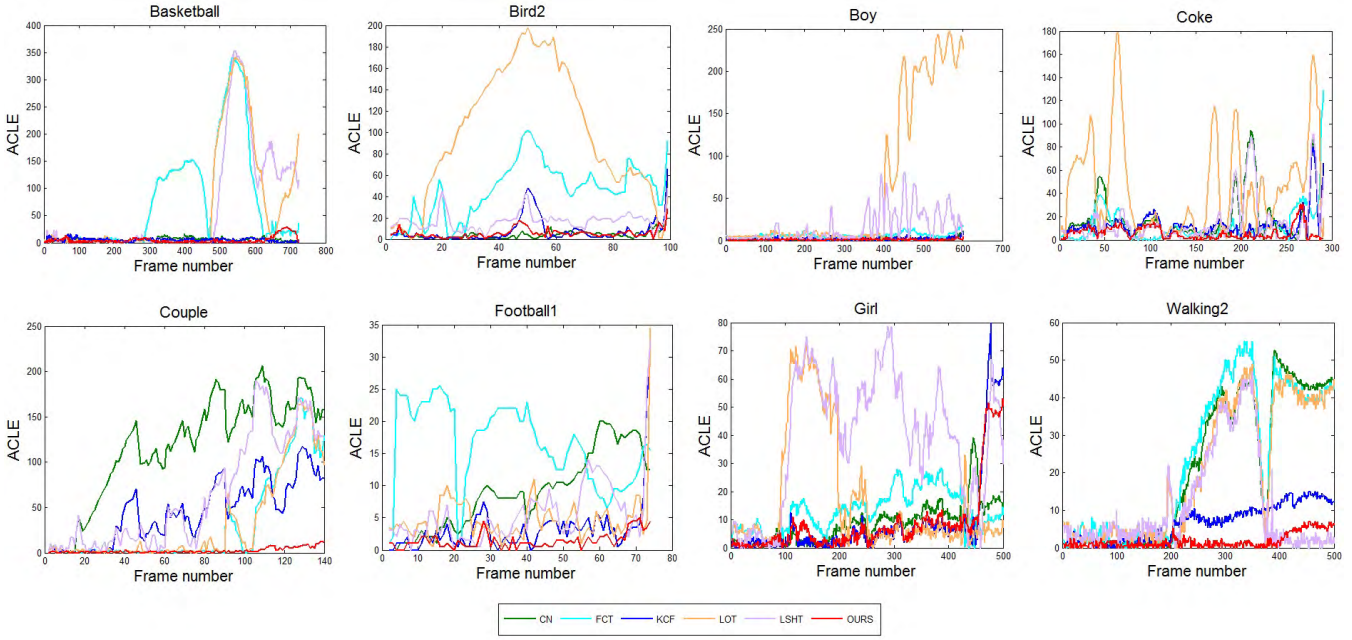

FIGURE 6. ACLE comparison curves on eight representative challenging video sequences.

TABLE 1. Tracking performance comparison (ACLE(unite: pixels)) with the state-of-the-art trackers, where the bold data is the best.

	VTD	MIL	TLD	STRUCK	LOT	LSHT
BOY	7.17	6.37	3.79	2.97	42.1	15.6664
Basketball	16.87	95.13	-	103.3	54.04	70.6381
Car4	19.26	30.37	53.14	13.3	144.86	87.7067
Coke	44.71	25.78	23.3	7.89	44.38	35.5948
Couple	65.94	31.24	4.42	16.39	30.59	64.3309
Football1	5.44	4.87	11.86	3.37	5.9247	-
Girl	6.19	11.78	5.94	**2.66**	20.07	24.4459
Walking2	13.84	20.02	7.02	3.16	19.87	2.7886
bird2	-	-	-	-	47.4949	17.1513
Dog1	-	-	-	-	-	-

	CN	FCT	KCF	Mul-Feature	AIA+LDA
BOY	4.77	6.07	1.97	1.16	**1.0624**
Basketball	9.45	77.4	**5.02**	16.511	8.9724
Car4	20.2	37.7	**9.58**	17.0669	15.0812
Coke	30.9	12	12.67	7.6778	**6.5431**
Couple	123	31.9	45	3.077	**2.4712**
Football1	-	16.4	3.25	2.363	**1.1849**
Girl	12.5	12.96	8.39	11.7735	9.515
Walking2	47.7	23.5	6.66	3.4609	**1.7174**
bird2	34.255	45.8163	7.6735	9.6875	**5.0204**
Dog1	-	-	-	-	3.4941

TABLE 2. Tracking performance comparison (AOR[0.00,1.00]) with the state-of-the-art trackers, where the bold data is the best.

	TLD	MIL	Struck	VTD	LOT	LSHT
BOY	0.69	0.51	0.68	0.6	0.47	0.3393
Basketball	0.06	0.22	0.18	0.63	0.46	0.4363
Car4	0.33	0.23	0.44	0.35	0.03	0.2097
Coke	0.33	0.24	0.58	0.13	0.01	0.1589
Couple	0.61	0.48	0.53	0.2	0.45	0.1778
Football1	0.36	0.6	0.46	0.54	0.61	0.5834
Girl	0.56	0.39	**0.68**	0.58	0.36	0.2499
Walking2	0.41	0.29	**0.47**	0.32	0.31	0.3788
bird2	-	-	-	-	0.183	0.5421
Dog1	-	-	-	-	-	-

	CN	FCT	KCF	Mul-Feature	AIA+LDA
BOY	0.61	0.63	0.65	**0.7354**	0.7347
Basketball	0.64	0.23	**0.68**	0.58	0.6593
Car4	0.38	0.24	**0.59**	0.45	0.4546
Coke	0.30	0.36	0.39	0.6173	**0.697**
Couple	0.10	0.48	0.22	0.6038	**0.6432**
Football1	-	0.17	0.48	0.7392	**0.7655**
Girl	0.42	0.36	0.51	0.4163	0.5086
Walking2	0.35	0.28	0.38	0.4623	0.4437
bird2	0.151	0.100	0.5769	0.6641	**0.7664**
Dog1	-	-	-	-	0.4421

Tables 1 and 2 shows a comparison of the ten trackers with regard to their ACLE and AOR scores; the ACLE and AOR curves of typical video sequences are shown in Figures 6 and 7. As shown, the proposed algorithm achieves the best results in most of the examples but performs slightly worse on some videos, such as basketball, car4 and girl, whose common features are occlusion (OCC) and offset (OPR). These results may occur for two reasons: first, the feature extraction algorithm selected for the extraction feature stage may not be comprehensive enough; second, the base tracker (RET) we selected is not good at dealing with occlusion and offset. In addition, a more efficient feature extraction algorithm can be considered in subsequent work.

Figures 6 and 7 visually show the performance of our method at different tracking stages. Our method is basically stable for most videos and shows obvious improvement compared to the other tested methods. In addition, Figure 8 shows the performance of the proposed method in OPE (one-pass evaluation) on seven videos compared with the 29 other trackers as reported in [27]. The seven videos included Boy, Basketball, Football1, Coke, Couple, Girl, and Walking2.

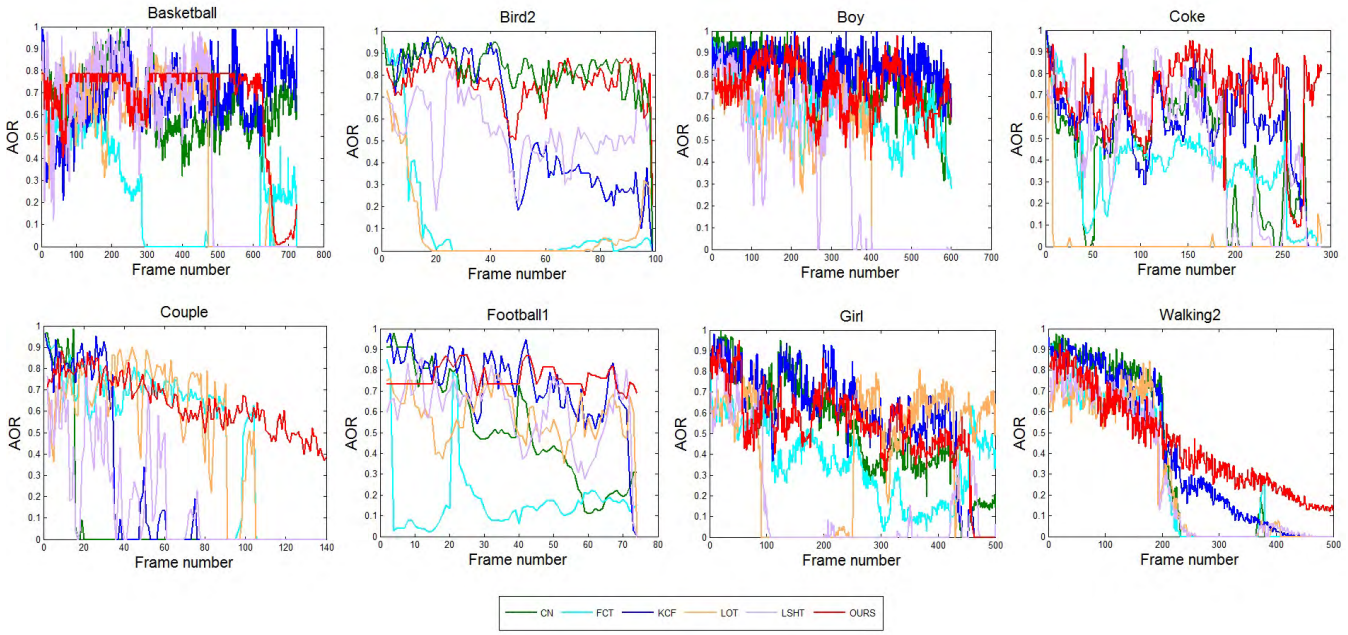

FIGURE 7. Comparison curves of AOR on eight representative challenging video sequences.

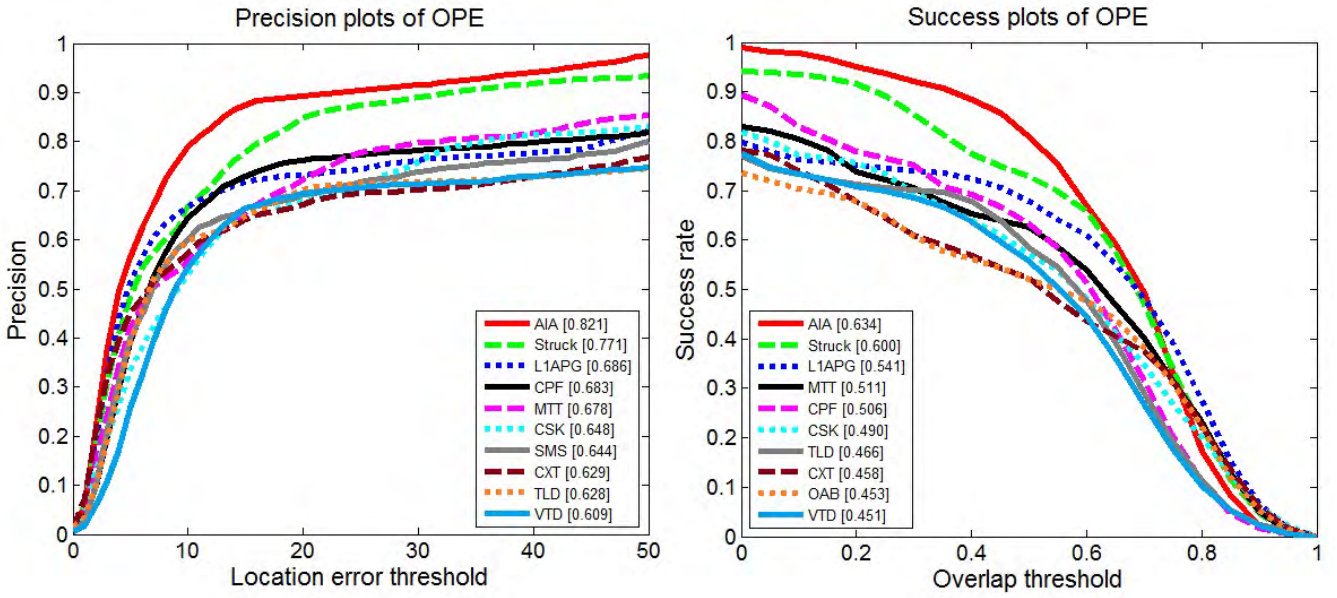

FIGURE 8. Our method in OPE perfomance on seven videos comparing with other trackers.

As the data shows, the proposed method achieved the best results compared with the other state-of-the-art trackers. It is worth mentioning that our method's superiority is obvious when compared with the same tracker using the same framework but without feature selection. This result demonstrates that it is better to refrain from employing all the features in the visual model because some features cause noise that may decrease the tracking performance. Moreover, these results verify that using a fixed visual model is usually redundant, noisy and not adaptive. The visual models employed in other trackers are all fixed; consequently, they are unable to build adaptive visual models for the specific videos.

Figure 9 shows the tracking results on some key frames and the corresponding situation of the visual model.For the video "basketball", an athlete moves continually through a crowd of people with similar appearances. This situation poses great challenges to the robustness of a visual model. For example, at frame ♯283, another athlete in the same green

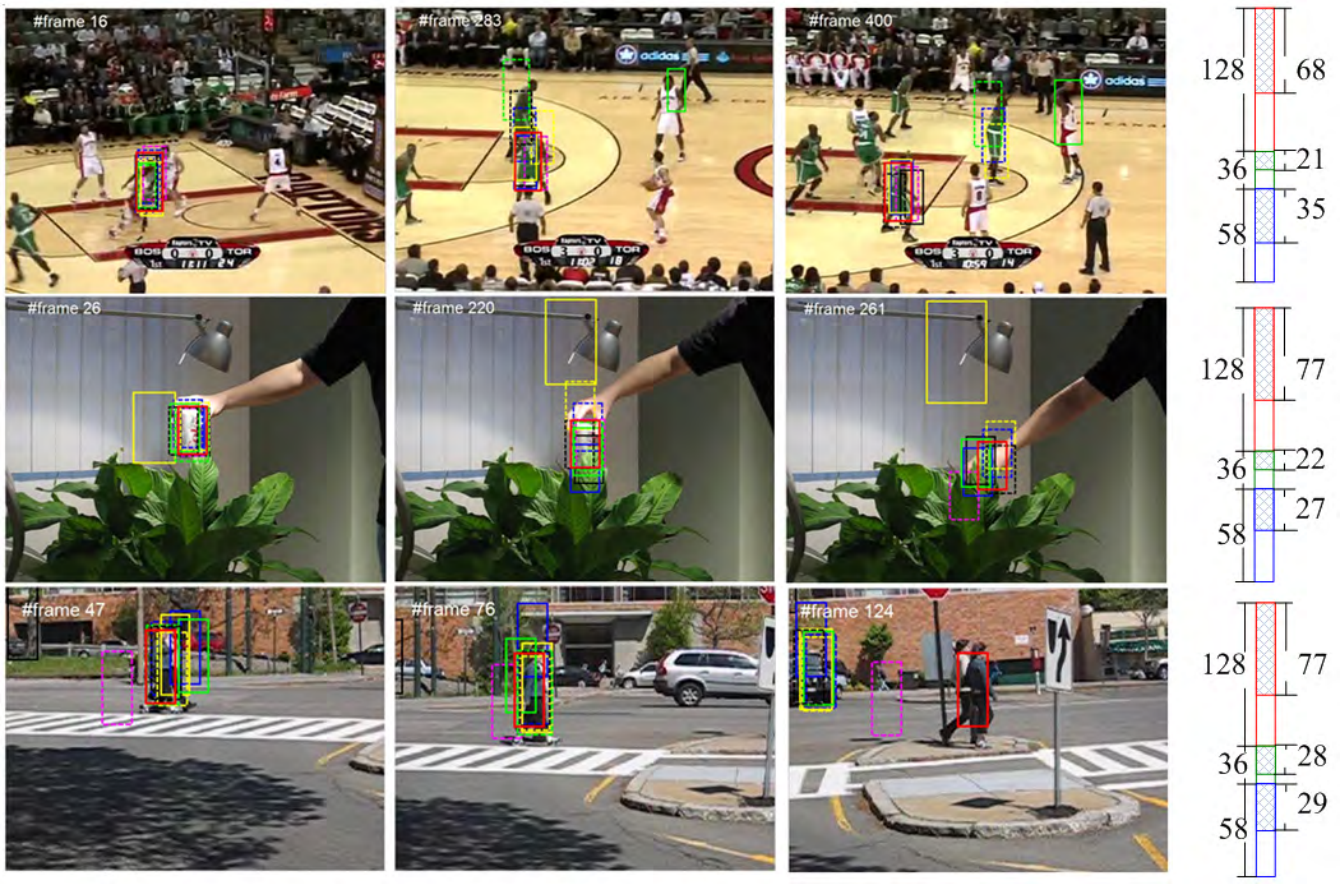

FIGURE 9. Result examples on some key frames of the representative videos.The data for the selected features are shown to the right of the figure. The three original feature vectors of HSV, HOG and LBP are 128 × 1, 36 × 1 and 58 × 1, respectively; the sizes of the selected features are provided at the right of the vector column for the three videos.

clothes moves behind the target. Many trackers with fixed visual models lose the target and lock on the wrong athlete. In our model, the feature selection result shows that more textural features (LBP) and fewer color features (HSV) are selected for such a video. In this sequence, the main challenge is confusion between the target and the other athlete with similar color attributes. To address this situation, the proposed tracker reduces the percentage of color cues after the optimization step. For the video ''coke'', frequent and severe occlusions challenge the visual model. Over the whole sequence, the leaves in the background have textures and edge features similar to the target. Therefore, the proposed tracker selects more color feature elements. Similarly, for the video ''couple'', the color cue shows comparatively superior discriminative ability; thus, the color cue is selected with more feature elements in the final visual model.

V. CONCLUSIONS

In this paper, we proposed a novel idea to achieve an adaptive form of visual modeling. Specifically, by considering visual modeling as an optimization problem, we proposed using a method called the Binary Artificial Immune Algorithm to build an optimum visual model for a given feature pool. In the collected visual cue pool, the discriminative ability of the visual model is measured by discrete random samples and defined as the objective function of the optimization problem. The Binary Artificial Immune Algorithm yielded an excellent solution to the given problem and achieved good tracking performance. Compared with other state-of-the-art trackers, the proposed tracker shows obvious advantages. Finally, the proposed method can easily be extended to other visual models because the problem definition is independent of the feature pool and modeling methods.

REFERENCES

[1] R. Yao, S. Xia, Y. Zhou, and Q. Niu, ''Robust lifelong visual tracking using compact binary feature with color attributes,'' *Neurocomputing*, vol. 213, no. 12, pp. 172–182, 2016.

[2] 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.

[3] G. Bradski, ''Real time face and object tracking as a component of a perceptual user interface,'' in *Proc. IEEE Workshop Appl. Comput. Vis.*, Oct. 1998, pp. 214–219.

[4] N. Dalal and B. Triggs, ''Histograms of oriented gradients for human detection,'' in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 1, Jun. 2005, no. 1, pp. 886–893.

[5] T. Ojala, M. Pietikäinen, and T. Mäenpää, ''Multiresolution gray-scale and rotation invariant texture classification with local binary patterns,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 971–987, Jul. 2002.

[6] B. Babenko, M.-H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 983–990.
[7] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1631–1643, Oct. 2005.
[8] H. Grabner and H. Bischof, "On-line boosting and vision," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2006, pp. 260–267.
[9] Q. Yu, T. Ba Dinh, and G. Medioni, "Online tracking and reacquisition using co-trained generative and discriminative trackers," in *Proc. Eur. Conf. Comput. Vis.*, 2008, pp. 678–691.
[10] M. Spengler and B. Schiele, "Towards robust multi-cue integration for visual tracking," *Mach. Vis. Appl.*, vol. 14, no. 1, pp. 50–58, 2003.
[11] P. Perez, J. Vermaak, and A. Blake, "Data fusion for visual tracking with particles," *Proc. IEEE*, vol. 92, no. 3, pp. 495–513, Mar. 2004.
[12] H. Wang, D. Suter, K. Schindler, and C. Shen, "Adaptive object tracking based on an effective appearance filter," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1661–1667, Sep. 2007.
[13] Y. Wu and T. Huang, "Robust visual tracking by integrating multiple cues based on co-inference learning," *Int. J. Comput. Vis.*, vol. 58, no. 1, pp. 55–71, 2004.
[14] W. Michael, K. Iljung, and B. Serge, "Object detection using generalization and efficiency balanced co-occurrence features," in *Proc. ICCV*, 2015, pp. 46–54.
[15] L.-T. Laura, C.-F. Cristian, and S. Konrad, "Learning by tracking: Siamese CNN for robust target association," in *Proc. CVPR*, 2016, pp. 33–40.
[16] C. Wang and K. Siddiqi, "Differential geometry boosts convolutional neural networks for object detection," in *Proc. CVPR*, 2016, pp. 51–58.
[17] G. Zhu, F. Porikli, and H. Li, "Robust visual tracking with deep convolutional neural network based object proposals on PETS," in *Proc. CVPR*, 2016, pp. 26–33.
[18] Y. Xiang, A. Alahi, and S. Savarese, "Learning complexity-aware cascades for deep pedestrian detection," in *Proc. ICCV*, 2015, pp. 3361–3369.
[19] M. Tang, I. B. Ayed, and Y. Boykov, "Deep learning strong parts for pedestrian detection," in *Proc. ICCV*, 2015, pp. 1904–1912.
[20] C. Huang, S. Lucey, and D. Ramanan, "Learning policies for adaptive tracking with deep feature cascades," in *Proc. ICCV*, 2017, pp. 105–114.
[21] X. Liu *et al.* (2017). "HydraPlus-Net: Attentive deep features for pedestrian analysis." [Online]. Available: https://arxiv.org/abs/1709.09930
[22] F. Li, C. Tian, W. Zuo, L. Zhang, and M.-H. Yang. (2018). "Learning spatial-temporal regularized correlation filters for visual tracking." [Online]. Available: https://arxiv.org/abs/1803.08679
[23] H. Fan and H. Ling, "Parallel tracking and verifying: A framework for real-time and high accuracy visual tracking," in *Proc. ICCV*, 2017, pp. 5487–5495.
[24] H. K. Galoogahi, A. Fagg, and S. Lucey, "Learning background-aware correlation filters for visual tracking," in *Proc. ICCV*, 2017, pp. 1144–1152.
[25] X. Zhou, L. Xie, P. Zhang, and Y. Zhang, "Online object tracking based on CNN with metropolis-hasting re-sampling," in *Proc. ACM Multimedia*, 2015, pp. 1163–1166.
[26] K. Zhang, Q. Liu, Y. Wu, and M.-H. Yang, "Robust visual tracking via convolutional networks without training," *IEEE Trans. Image Process.*, vol. 25, no. 4, pp. 1779–1792, Apr. 2016.
[27] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *Proc. CVPR*, 2013, pp. 2411–2418.
[28] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 7, pp. 1409–1422, Jul. 2011.
[29] S. Hare, A. Saffari, and P. H. S. Torr, "Structured output tracking with kernels," in *Proc. Int. Conf. Comput. Vis.*, 2011, pp. 263–270.
[30] J. Kwon and K. M. Lee, "Visual tracking decomposition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2010, pp. 1269–1276.
[31] S. Oron, A. Bar-Hillel, D. Levi, and S. Avidan, "Locally orderless tracking," *Int. J. Comput. Vis.*, vol. 111, no. 2, pp. 1940–1947, 2012.
[32] S. He, Q. Yang, R. W. H. Lau, J. Wang, and M.-H. Yang, "Visual tracking via locality sensitive histograms," in *Proc. CVPR*, vol. 9, 2013, no. 4, pp. 2427–2434.
[33] M. Danelljan, F. S. Khan, M. Felsberg, and J. Van de Weijer, "Adaptive color attributes for real-time visual tracking," in *Proc. Comput. Vis. Pattern Recognit.*, 2014, pp. 1090–1097.
[34] K. Zhang, L. Zhang, and M.-H. Yang, "Fast compressive tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 10, pp. 2002–2015, Oct. 2014.
[35] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 3, pp. 583–596, Mar. 2014.
[36] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.
[37] Z. Zhou, *Machine Learning*. Beijing, China, 2016, pp. 60–61.
[38] Q. Cheng, H. Zhou, and J. Cheng, "The Fisher–Markov selector: Fast selecting maximally separable feature subset for multiclass classification with applications to high-dimensional data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 6, pp. 1217–1233, Jun. 2011.
[39] X. Li and M. Yin, "Multiobjective binary biogeography based optimization for feature selection using gene expression data," *IEEE Trans. Nanobiosci.*, vol. 12, no. 4, pp. 343–353, Dec. 2013.
[40] Q. Bai, Z. Wu, S. Sclaroff, M. Betke, and C. Monnier, "Randomized ensemble tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2040–2047.
[41] Y. Duan, J. Lu, J. Feng, and J. Zhou, "Context-aware local binary feature learning for face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 5, pp. 1139–1153, May 2018.
[42] V. Balntas, L. Tang, and K. Mikolajczyk, "Binary online learned descriptors," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 3, pp. 555–567, Mar. 2018.
[43] X. Wen, L. Shao, Y. Xue, and W. Fang, "A rapid learning algorithm for vehicle classification," *Inf. Sci.*, vol. 295, no. 1, pp. 395–406, 2015.
[44] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *Proc. Brit. Mach. Vis. Conf.*, 2006, pp. 47–56.
[45] M. Tian, W. Zhang, and F. Liu, "On-line ensemble SVM for robust object tracking," in *Proc. Asian Conf. Comput. Vis.*, 2007, pp. 355–364.
[46] T. Penne, C. Tilmant, T. Chateau, and V. Barra, "Modular ensemble tracking," in *Proc. Int. Conf. Image Process. Theory Tools Appl.*, 2010, pp. 363–368.
[47] Y. Wang, Q. Liu, M. Yin, and S. Wang, "Large margin classifier based ensemble tracking," *J. Electron. Imag.*, vol. 25, no. 4, p. 043006, 2016.
[48] P. Wang, C. Sanin, and E. Szczerbicki, "Evolutionary algorithm and decisional DNA for multiple travelling salesman problem," *Neurocomputing*, vol. 150, pp. 50–57, Feb. 2015.
[49] D. Gory, X. Ji, J. Sun, and X. Sun, "Interactive evolutionary algorithms with decision-maker's preferences for solving interval multi-objective optimization problems," *Neurocomputing*, vol. 137, pp. 241–251, Aug. 2014.
[50] N. K. Jerne, "Towards a network theory of the immune system," *Annu. Immunol.*, vol. 125, no. 3, pp. 373–389, 1974.

WEI GAO received the B.S. degree in military modeling and simulation from the University of Aerospace Engineering, China, in 2004. She is with the School of Astronautics, Beihang University, where she is currently pursuing the Ph.D. degree in flight vehicle design. Her research interests include automated control and planning for space robots.

JINLING ZHANG received the bachelor's degree from the Department of Computer Science and Technology, Northeast Normal University, Changchun, China, in 2016, where she is currently pursuing the M.S. degree with the Department of Computer Science and Technology. Her current research interests are visual tracking and feature extraction.

QIAOYUAN LIU received the bachelor's degree from the Department of Computer Science and Technology, Northeast University, Shenyang, China, in 2014. She is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Northeast Normal University, Changchun, China. Her current research interests are visual tracking.

LONGKUI JIANG received the M.S. degree from the Department of Computer Science and Technology, Northeast Normal University. His current research interests include computer visions and pattern recognition.

YURU WANG received the Ph.D. degree from the Department of Computer Science and Technology, Harbin Institute of Technology, China, in 2010. Her current research interests include computer visions and pattern recognition.

MINGHAO YIN received the B.S. and M.S. degrees in computer science from the Northeast Normal University, China, in 2001 and 2004, respectively, and the Ph.D. degree in computer science from Jilin University, China, in 2008. Since 2010, he has been the Dean of the Department. He is currently a Professor with the Department of Computer, Northeast Normal University. He has authored two books, and over 100 articles. His research interests include swarm intelligence, automated reasoning, automated planning, and algorithms.

YUPENG ZHOU received the bachelor's degree from the Department of Computer Science and Technology, Northeast Normal University, Changchun, China, in 2014, where he is currently pursuing the Ph.D. degree. His current research interests include local search and combinatorial optimization.