

## Research Article

# Visual Tracking Based on an Improved Online Multiple Instance Learning Algorithm

Li Jia Wang<sup>1</sup> and Hua Zhang<sup>2</sup>

<sup>1</sup>Department of Information Engineering and Automation, Hebei College of Industry and Technology, Shijiazhuang 050091, China

<sup>2</sup>Faculty of Electrical & Electronics Engineering, Shijiazhuang Vocational Technology Institute, Shijiazhuang 050081, China

Correspondence should be addressed to Li Jia Wang; wanglijia1981@hotmail.com

Received 14 August 2015; Revised 12 November 2015; Accepted 16 November 2015

Academic Editor: Manuel Graña

Copyright © 2016 L. J. Wang and H. Zhang. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

An improved online multiple instance learning (IMIL) for a visual tracking algorithm is proposed. In the IMIL algorithm, the importance of each instance contributing to a bag probability is with respect to their probabilities. A selection strategy based on an inner product is presented to choose weak classifier from a classifier pool, which avoids computing instance probabilities and bag probability  $M$  times. Furthermore, a feedback strategy is presented to update weak classifiers. In the feedback update strategy, different weights are assigned to the tracking result and template according to the maximum classifier score. Finally, the presented algorithm is compared with other state-of-the-art algorithms. The experimental results demonstrate that the proposed tracking algorithm runs in real-time and is robust to occlusion and appearance changes.

## 1. Introduction

The purpose of visual tracking is to estimate the state of a target from frame to frame. It has been widely studied in computer vision, for example, surveillance, robot navigation, and human-computer interaction [1]. Although various tracking algorithms have been proposed [2–11], visual tracking is still a very challenging task due to appearance changes caused by pose, illumination, occlusion, and motion [1].

An effective appearance model is important in the tracking process. The appearance model represents a target by using information extracted from the target region. Since color feature is readily accessible from an image, it is widely used to model a target's appearance [3]. One of the best methods for color-based visual tracking is to realize the MS algorithm [4, 5]. The color histogram of a target is learned in the first frame for color based object tracking algorithm, and then the target is searched in the subsequent frames by measuring the similarity between two histograms. The MS algorithm has been successfully used in visual tracking due to its similarity, efficiency, and robustness [4, 5]. However, the color histogram can only describe the color distribution in a region but ignore the spatial distribution.

Therefore, its representation ability will significantly decline when there are partial occlusions or similar objects. As a result, a color based tracker may wrongly detect the “bad” candidate as tracking result. To overcome the drawback, parts-based color histograms were adopted [6]. The method is robust to occlusions as it combines the color histograms of multiple parts. Recently, compressed sensing theory has attracted much attention in computer vision, for example, face recognition [7], denoising and in-painting [8], and visual tracking [9–12]. It is demonstrated that a compressed sensing based tracking method can cope with partial occlusions, illumination changes, and pose variations [11].

The generative appearance models mentioned above are learned in the first frame. Therefore, they cannot deal with major appearance changes as the track evolves. Discriminative algorithms, which can change appearance model evolution, were proposed [13]. The main idea of these appearance models is to consider the tracking problem as a binary classification task by continuously updating a classifier [14]. The best way to implement this task is to design appearance models for the target and background [14] and after that learn a discriminative classifier according to the models. Grabner

[15] proposed an online boosting algorithm to learn a discriminative appearance model for visual tracking. The tracker took the current tracking location as one positive example and extracted negative samples from neighborhood around the tracking location. However, the results often drifted away as the appearance model was updated with a suboptimal positive example. Zhang et al. [16] implemented a real-time compressed tracking algorithm. Multiple positive examples and negative ones were cropped around the current tracking location to update a discriminative classifier. However, the ambiguity problem may occur and confuse the classifier [17]. To deal with this problem, Grabner et al. [17] proposed a semisupervised boosting approach where only the samples from the first frame were labeled and the subsequent training examples were left unlabeled. The method achieved good performance even if the target disappears from the field of view of a camera. However, the tracker failed when there was great interframe motion.

The methods of tracking-by-detection have progressed in the past few decades. These methods train a classifier by adopting positive samples and negative ones extracted from the current frame. In the tracking process, some candidates around the old object location are extracted, and the trained classifier is employed to find the sample with the maximum classifier score. In fact, Viola et al. [18] argued that the inherent ambiguities included in object detection would cause difficulty for traditional learning methods. To handle the problem, the multiple instance learning (MIL) method was proposed for object detection [18, 19] and object recognition [20–22]. Studies show that the technique can cope with partial occlusions and ambiguities [22, 23]. The basic idea of the MIL method is that examples are called bags and some instances are included in these bags [14]. The bags instead of the individual instance are labeled. A bag is positive if at least one instance is positive. In contrast, a bag is negative when all of the instances are negative. Normally, the positive bag is constructed by the instances sampled around the labeled object. The MIL based tracking method can cope with the ambiguity problem by finding out the most “correct” instance in each positive bag [19]. However, the MIL trackers still have some shortcomings, for example, a heavy computational load, tracking drift, and failure to handle appearance variations. To address these problems, Zhang and Song [24] proposed an improved MIL tracker by weighting instance probabilities. Zhang and Song used an efficient online approach to maximize the bag likelihood function, which resulted in robust and fast tracking. However, when a bag probability is computed, the included instances are assigned different weights based on their distance to the previous tracking location. As a result, the target will be lost when drift occurs, because the real instance far away from the previous tracking location is assigned a large weight. Xu et al. [25] presented an efficient MIL tracker by using a Fisher information criterion, which yielded robust performance. Zhou et al. [26] implemented online visual tracking by using MIL with instance significance estimation. The method provided each instance a significance-coefficient which represents its contribution to the bag likelihood. The method dealt with the drift problem to some extent. In general,

these MIL trackers successfully deal with slight appearance variations and achieve a robust and fast tracking. However, these trackers fail to handle strong ambiguity, for example, large appearance variations, and tracking drift.

In this study, to deal with the above problems, we propose an improved online multiple instance learning algorithm (IMIL) for object tracking. The IMIL algorithm has advantages over the other MIL trackers in terms of efficiency, accuracy, and robustness. First, to reduce the computational cost, an inner product based selection method is employed to choose the weak classifiers from a classifier pool. The selection strategy avoids computing the instance probabilities and bag probability  $M$  times. Consequently, compared with the MIL, the IMIL runs in real time. Second, to generate a bag probability, the importance of each instance is with respect to their probabilities. Compared with the WMIL tracker, the strategy avoids introducing interference from the background and then deals with the drift problem. Third, to cope with the variations on illumination and pose, a feedback strategy is proposed to update the parameters of classifiers. In the strategy, the learning rate is tuned by considering the relationship between the maximum similarity score and two given thresholds after detecting the object. Therefore, using the proposed update strategy, the IMIL handle the variations on appearance and illumination.

The rest of this paper is organized as follows. Section 2 details the proposed tracking algorithm; Section 3 presents the experimental results of the proposed tracker compared with other trackers; the paper concludes with a short summary in Section 4.

## 2. The IMIL Based Tracking Algorithm

An improved online multiple instance learning algorithm is proposed for object tracking in video sequences. The basic flow of the IMIL algorithm is shown in Figure 1. Each instance is represented by a set of compressed Haar-like features [14, 24]. Let  $l_t(x) \in R^2$  denote the location of each instance  $x$  in the  $t$ th frame. It is assumed that the tracker can detect the object at the location  $l_t^*$ . Once the object is detected, a set of instances  $X^r = \{x : \|l(x) - l_t^*\| < r\}$  are cropped to construct a positive bag. In contrast, the instances for negative bag are cropped from an annular region  $X^{r,\beta} = \{x : r < \|l(x) - l_t^*\| < \beta\}$ ,  $r < \beta$ . The extracted positive bag and negative one are adopted to train weak classifiers. Then, a strong classifier is generated by selecting the weak classifiers with stronger classifying ability. For the  $t+1$  frame, we assume that the object appears within a radius  $s$  of the tracking location in the  $t$  frame. Thus, some candidate samples  $X^s = \{x : \|l_{t+1}(x) - l_t^*\| < s\}$ ,  $r < s < \beta$ , are cropped within a search radius  $s$  centering at the previous location  $l_t^*$ . The learned strong classifier estimates  $p(y = 1 | x)$  for all samples  $x \in X^s$  and updates the tracking location according to  $l_t^* = l(\arg \max_{x \in X^s} p(y = 1 | x))$ . Furthermore, the classifier updates the appearance model after tracking a new location.

*2.1. Online MIL Boosting Algorithm.* Babenko et al. [14] proposed an online MIL Boosting method for visual tracking. The details of the MIL Boosting tracker are shown

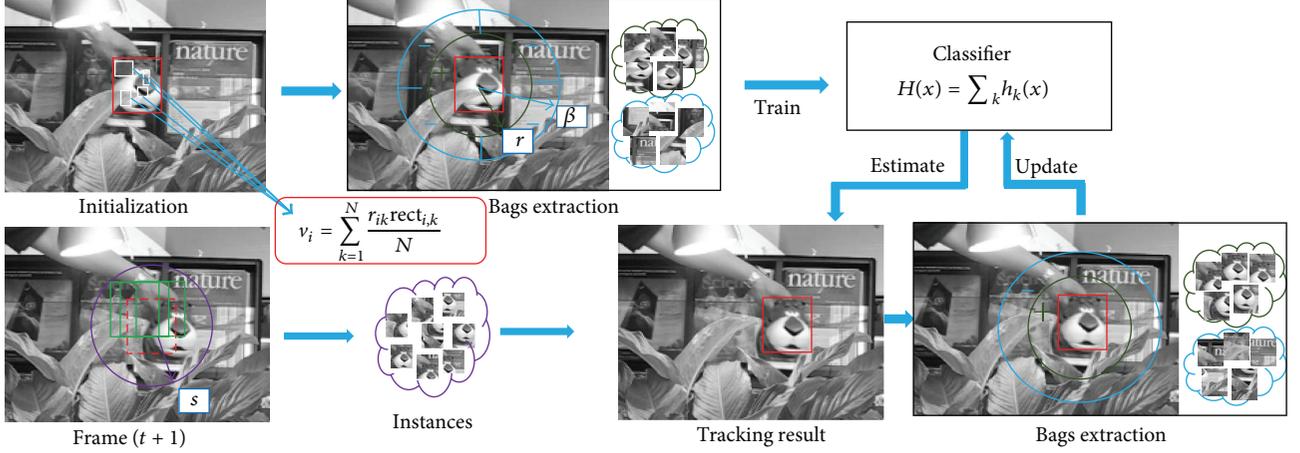
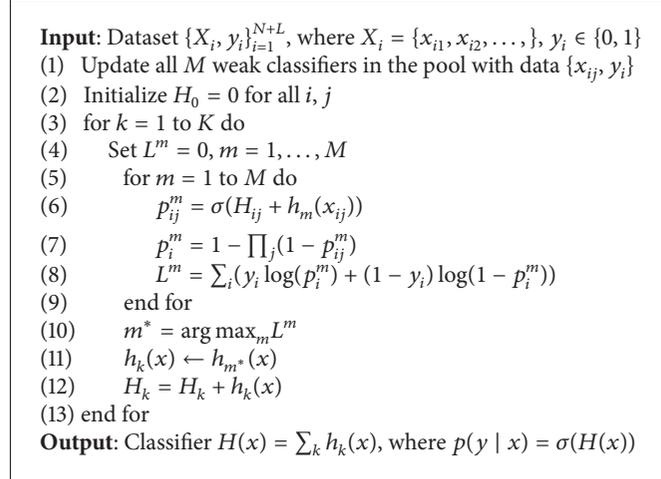


FIGURE 1: The flow chart of object tracking by using IMIL algorithm. The locations of the solid red rectangles are the tracking results. The areas of the brown circles are the ones for selecting instances to generate the positive bags. The areas of the blue circles are the ones for the negative bags. The location of the dashed rectangle in the  $t + 1$  frame is the tracking result from the  $t$  frame. The green rectangles in the  $t + 1$  frame are the candidate samples extracted from a small neighborhood of the result in the  $t$  frame.



ALGORITHM 1: Online MIL Boosting.

in Algorithm 1. The training data is  $\{(X_1, y_1), \dots, (X_n, y_n)\}$ , where  $X_i = \{x_{i1}, \dots, x_{im}\}$  is a bag constructed by instances  $x_i$  and  $y_i$  is the bag label according to  $X_i$ . The key of the MIL Boosting method is to design a strong classifier  $H(\cdot) = \sum_{k=1}^K h_k(\cdot)$ .  $h_k(\cdot)$  is a weak classifier relating to a sampled instance (represented by Haar-like feature  $V_k$ ) in the positive bag. The weak classifier  $h_k(\cdot)$  is assumed to be Gaussian distributed with four parameters  $(\mu_k^1, \sigma_k^1, \mu_k^0, \sigma_k^0)$ , where  $p(v_k(x) | y = 1) \sim N(\mu_k^1, \sigma_k^1)$  and  $p(v_k(x) | y = 0) \sim N(\mu_k^0, \sigma_k^0)$ . The weak classifier returns the log odds ratio:

$$h_k(x) = \log \left( \frac{p(y = 1 | v_k(x))}{p(y = 0 | v_k(x))} \right). \quad (1)$$

When the strong classifier receives new data  $\{(X_1, y_1), \dots, (X_n, y_n)\}$ , the instance probability is modeled as

$$p(y_i | x_{ij}) = \sigma(H(x_{ij})), \quad (2)$$

where  $\sigma(x) = 1/(1 + \exp(-x))$  is the sigmoid function. Then, the  $i$ th bag probability is modeled by adopting the Noisy-OR model [18]:

$$p(y_i | X_i) = 1 - \prod_j (1 - p(y_i | x_{ij})). \quad (3)$$

The MIL tracker learns a pool of  $M$  candidate weak classifiers. Then, the  $K$  ( $< M$ ) weak classifiers are selected from the candidate pool by maximizing the log likelihood of bags:

$$h_k = \operatorname{argmax}_{h \in \{h_1, \dots, h_M\}} L(H_{k-1} + h), \quad (4)$$

where  $L = \sum_i (y_i \log(p_i) + (1 - y_i) \log(1 - p_i))$  is the bag log-likelihood function and  $H_{k-1} = \sum_{k=0}^{k-1} h_k$ . Finally, the selected  $K$  weak classifiers generate a strong classifier  $H$  to determine the new tracking location with the maximum score.

After detecting the object in the new frame, the parameters of the weak classifiers are updated online:

$$\begin{aligned} \mu_i^1 &\leftarrow \lambda u_i^1 + (1 - \lambda) \mu^1 \\ \sigma_i^1 &\leftarrow \sqrt{\lambda (\sigma_i^1)^2 + (1 - \lambda) (\sigma^1)^2 + \lambda (1 - \lambda) (\mu_i^1 - \mu^1)^2}, \end{aligned} \quad (5)$$

where  $\lambda$  is a learning rate. It specifies the importance of the tracking result and the template.

The MIL Boosting algorithm deals with the suboptimal problem included in the weak classifiers. However, the bag probability computed by using the Noisy-OR model makes the MIL tracker easily select less effective features and confuse the classifier [27]. Another disadvantage of the MIL tracker is that all of the instance probabilities and the bag probabilities must be updated  $M$  times after selecting a feature, which results in a heavy computational load [24]. Moreover, the weak classifiers update their parameters with fixed learning rate, which is sensitive to the occlusion and variations on illumination and appearance.

**2.2. Bag Probability.** To overcome the drawback mentioned above, the IMIL tracker is proposed. It is assumed that the bag's probability depends on the instance probability equally [28]. The importance of each instance contributing to a bag probability mainly depends on the instance probability. The bigger the instance probability is, the more it contributes to the bag probability. We use an equally weighted sum of instance probabilities to yield superior performance. The positive bag probability is defined as follows:

$$p(y = 1 | X_i) = \sum_{j=0}^{N-1} p(y = 1 | x_{ij}). \quad (6)$$

Compared with the Noisy-OR model and the WMIL tracker, our method computes the bag probability according to each instance probability. In the WMIL tracker, the positive instances are assigned weights to generate the bag probability according to the Euclidean distances between the locations of each instance and the current tracking location  $p(y = 1 | X_i) = \sum_{j=0}^{N-1} w_{j0} p(y = 1 | x_{ij})$  [24]. The nearer the distance is, the more important the instance probability is. However, if the tracking location  $L_0$  drifts away from the real target position (see Figure 2), the instance  $L_{1i}$  far from the current tracking location will be assigned a smaller weight than that for the near instance  $L_{1j}$  due to their distances. As a result, the instance  $L_{1j}$  contributes more to the bag probability than the instance  $L_{1i}$ , which is contrary to the fact. As a result, the obtained bag probability will contain information from the background and finally lead to a tracking failure. In our method, the instance's importance is determined by their probabilities. The instance whose probability is bigger will contribute more to the bag probability. Using the strategy, the tracker achieves superior performance.

**2.3. Selection Strategy.** The MIL tracker often suffers from a heavy computational load as it selects the weak classifier  $h_k$



FIGURE 2: The illustration of constructing the bag probability. The location of the yellow rectangle is for the current tracking results. The red rectangles are for the positive instances, while the green rectangles are for the negative ones.

with stronger classifying ability from a weak classifier pool  $\phi = \{h_1, \dots, h_M\}$  by maximizing the log-likelihood function. To realize real time visual tracking, a more efficient criterion based on the inner product is employed to select weak classifiers from the classifier pool [24]. Using this criterion, the instance probabilities and bag probabilities do not need to be computed  $M$  times after selecting one weak classifier. Inspired by this, we select the weak classifier  $h_k$  as follows:

$$h_k = \operatorname{argmax}_{h \in \phi} \langle h, \nabla L(H) \rangle |_{H=H_{k-1}}, \quad (7)$$

where  $\langle h, \nabla L(H) \rangle = 1/(N + L) \sum_{j=0}^{N+L-1} h(x_{ij}) \nabla L(H)(x_{ij})$ ,

$$\begin{aligned} \nabla L(H)(x_{ij}) &= y_i \left( \frac{(\sigma(H(x_{ij}))(1 - \sigma(H(x_{ij}))))}{\sum_{m=0}^{N-1} \sigma(H(x_{im}))} \right) \\ &\quad - (1 - y_i) \left( \frac{(\sigma(H(x_{ij}))(1 - \sigma(H(x_{ij}))))}{\sum_{m=N}^{N+L-1} (1 - \sigma(H(x_{im})))} \right), \end{aligned} \quad (8)$$

where  $p(y | x_{ij}) = \sigma(H(x_{ij}))$ . The IMIL method is shown in Algorithm 2.

**2.4. Weak Classifiers.** The weak classifiers of the MIL update their four parameters with fixed weights [16], which may introduce errors when there are inaccurate tracking results. Normally, the tracking process is formulated as a searching problem which aims to select the candidate with the maximum classifier score ( $H_{\max}$ ). Usually, the most ‘‘correct’’ samples detected by the tracker are similar in the most frames. However, when there is an occlusion or a great interframe motion, the update strategy will introduce errors from background and lead to a suboptimal result in the rest of the frames. Consequently, drift will occur or the target will be lost.

To avoid the problems mentioned above, a feedback strategy is proposed for updating the classifiers. Our goal is to detect the candidate area not only the most similar to the foreground but also the most dissimilar to the background.

**Input:** Dataset  $\{X_i, y_i\}_{i=1}^{N+L}$ , where  $X_i = \{x_{i1}, x_{i2}, \dots\}$ ,  $y_i \in \{0, 1\}$ .

- (1) Update all  $M$  weak classifiers in the pool with the data  $\{x_{ij}, y_i\}$
- (2) Initialize  $H_0 = 0$  for all  $i, j$
- (3) for  $k = 1$  to  $K$  do
- (4)   Compute  $\nabla L(H)(x_{ij})|_{H=H_{k-1}}$  by (8)
- (5)   Set  $L^m = 0, m = 1, \dots, M$
- (6)   for  $m = 1$  to  $M$  do
- (7)     for  $i = 0$  to  $1$  do
- (8)       for  $j = 0$  to  $N + L - 1$  do
- (9)          $L^m \leftarrow L^m + h_m(x_{ij})\nabla L(H)(x_{ij})|_{H=H_{k-1}}$
- (10)       end for
- (11)     end for
- (12)   end for
- (13)    $m^* = \arg \max_m L^m$
- (14)    $h_k(x) \leftarrow h_{m^*}(x)$
- (15)    $H_k = H_{k-1} + h_k(x)$
- (16) end for

**Output:** Classifier  $H(x) = \sum_k h_k(x)$ .

ALGORITHM 2: Online IMIL.

The maximum classifier score  $H_{\max}$  obtained by applying the strong classifier denotes the similarity between the tracking result and target. In our method,  $\text{TH}_l$  and  $\text{TH}_s$  are set as higher threshold and lower threshold, respectively. To make sure the classifier is far from the background, the update strategy changes the learning rate  $\lambda$  considering the relationship between the  $H_{\max}$  and  $\text{TH}_l$  or  $\text{TH}_s$ :

$$\lambda = \begin{cases} 0.85, & H_{\max} > \text{TH}_l \\ 0.25, & \text{TH}_s < H_{\max} < \text{TH}_l \\ 0.05, & H_{\max} < \text{TH}_s, \end{cases} \quad (9)$$

where  $\lambda$  is the learning rate. In this strategy, it specifies the importance of the tracking result, while  $1 - \lambda$  specifies the importance of the template.  $H_{\max} > \text{TH}_l$  means that the candidate area is more similar to the template (foreground), while  $H_{\max} < \text{TH}_s$  means that the candidate area is more similar to the background. Considering the relationships between the  $H_{\max}$  and the two thresholds, two weights are assigned to the tracking result and template to specify their importance when the classifier is updating.

The feedback strategy can deal with appearance changes and avoid excessive updates as the track evolves. When the target is successfully tracked, the maximum classifier score  $H_{\max}$  is greater than  $\text{TH}_l$ , which means that the similarity between the tracking result and template is very high. Then, the weight of the tracking result is increased to the largest value to cope with appearance variations. In other words, the new classifier depends mainly on the tracking result. When there is a serious occlusion or the target is lost, the most “correct” target includes numerous background information. The maximum classifier score  $H_{\max}$  is less than the lower threshold  $\text{TH}_s$ , which means that the similarity between the tracking result and template is very low. Then, the weight of the tracking result should be a small value to avoid introducing more errors to the classifier. As a result, the new

classifier depends mainly on the template. When there are partial occlusions or some appearance variations, the  $H_{\max}$  is between the lower threshold and the higher one. In such a case, the classifier should update according to the tracking result and the template simultaneously.

### 3. Experiments

We compared the proposed object tracking algorithm (IMIL) with the 3 latest trackers on 6 challenging video sequences. The three trackers are online MIL Boosting tracker (MIL) [14], WMIL tracker [24], and significance-coefficients MIL [26]. For the compared trackers, the binary code released by the authors is used [26]. The six video sequences are “David indoor” [24], “Occluded face” [24], “Tiger 2” [24], “Cliff bar” [24], “Coke can” [14], and “Coupon book” [26] (Figure 3). Using these video sequences, we evaluate the IMIL tracker’s ability of handling the problems of illumination changes, occlusion, pose variations, and appearance changes. There are serious illumination variations in “David indoor” and “Coke can.” And the “can” moves fast in the “Coke can” video. In the sequences “Occluded face 2” and “Tiger 2,” the face is often occluded by a book and the “Tiger” is often occluded by the leaves. Moreover, there are still pose changes in these video sequences. The pose variations exist in the video “Cliff bar,” while the appearance of the “dollar” changes in the video “Coupon book.” The proposed IMIL is implemented in the MATLAB and run on a core 2 CPU, 2.33 GHz, and 2 GB RAM computer.

**3.1. Parameters Setting.** For the online MIL Boosting tracker [14], the search radius  $s$  is set to 35, and about 1000 samples are extracted for detecting the object location. Set  $r = 4$  and sample about 45 positive instances for generating the positive bag;  $\beta = 50$ ; the negative instances in the negative bag are about 65. The number of the candidate weak classifiers is 250. 50 classifiers are selected from weak classifier pool to generate

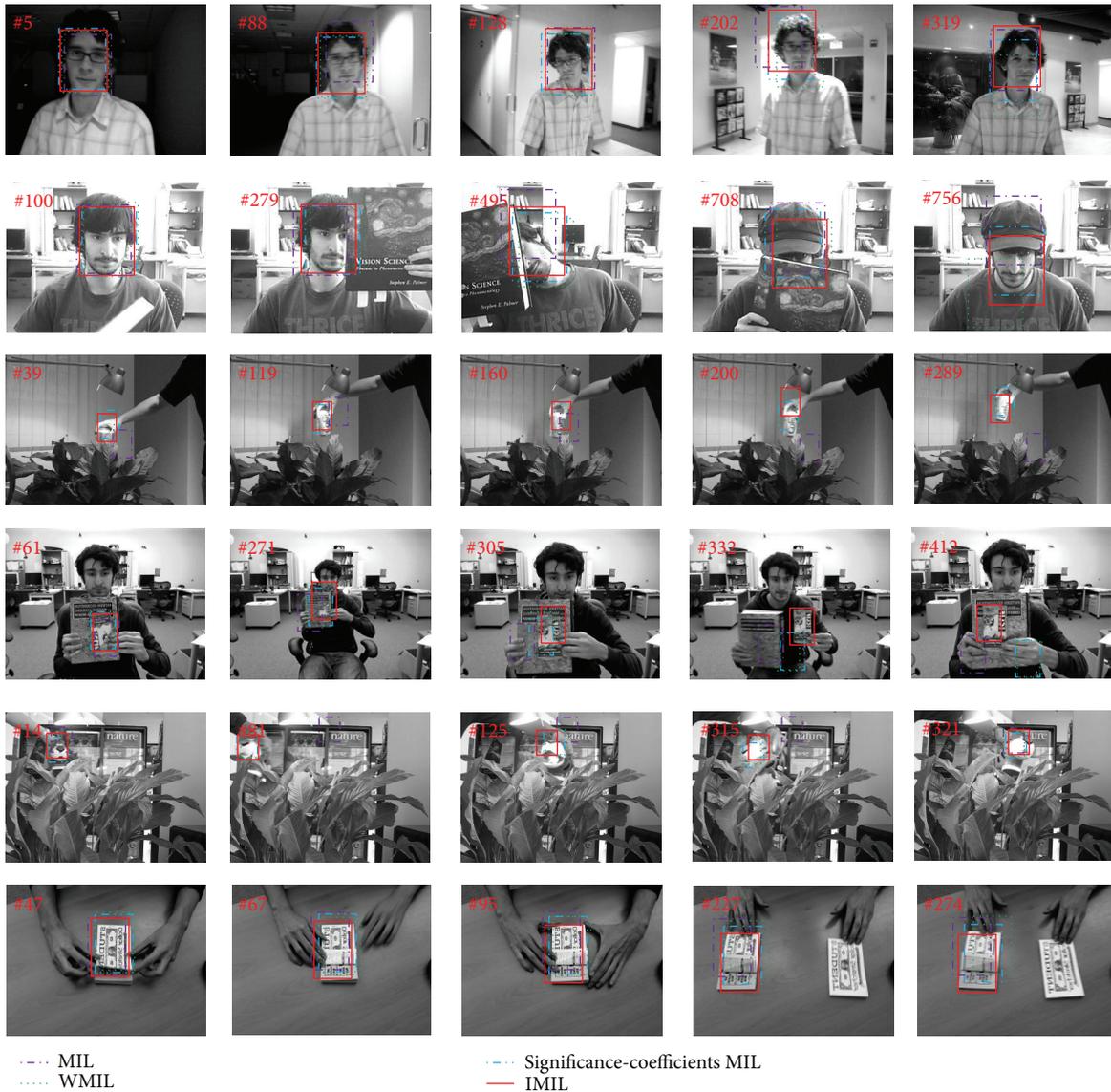


FIGURE 3: Tracking object location by using MIL [14], WMI [24], significance-coefficients MIL [26], and IMIL. The video sequences are “David indoor,” “Occluded face,” “Coke can,” “Cliff bar,” “Tiger 2,” and “Coupon book” from the top to the bottom.

a strong classifier. The learning rate of the weak classifier is 0.85. For the WMIL tracker [24], set  $s = 25$  to search the target;  $r = 4$  to crop the positive instances;  $a = 2r$  and  $\beta = 1.5s$  to determine the negative instances. The number of candidate weak classifiers is set to be  $M = 150$ . 15 weak classifiers are selected from classifiers pool to generate a strong WMIL classifier. The learning parameter is still 0.85. For the significance-coefficients MIL [26], set  $r = 4$  and  $\beta = 50$  to generate the positive and negative bags; the learning rate is set to be 0.85. The number of the candidate weak classifiers is  $M = 150$ , and 15 weak classifiers are selected for generating a strong classifier. For the IMIL tracker, set  $s = 25$  for detecting the target;  $r = 4$  is set for sampling positive instances;  $a = 2r$  and  $\beta = 1.5s$  are set for extracting negative instances. To generate a strong classifier, 30 weak classifiers are selected from the classifier pool which includes 120 weak classifiers.

**3.2. Tracking Object Location.** We perform our experiments on the above 6 video sequences. For all sequences, the images are converted into gray scale before processing. For each sequence, the classifier is learned in the first frame. Then, the locations in the subsequent frames are tracked by these trackers. For updating the classifier, the proposed IMIL algorithm tunes the parameters of the tracking result and template according to the maximum similarity in the tracking process. Therefore, compared with MIL, WMIL, and significance-coefficients MIL trackers, the IMIL tracker achieves the best performance when there are appearance changes, pose and illumination variations. Furthermore, the IMIL method computes a bag probability according to instance probability. Therefore, an instance contributes more to a bag probability if the instance probability is bigger. As a result, the IMIL tracker overcomes the tracking drift problem

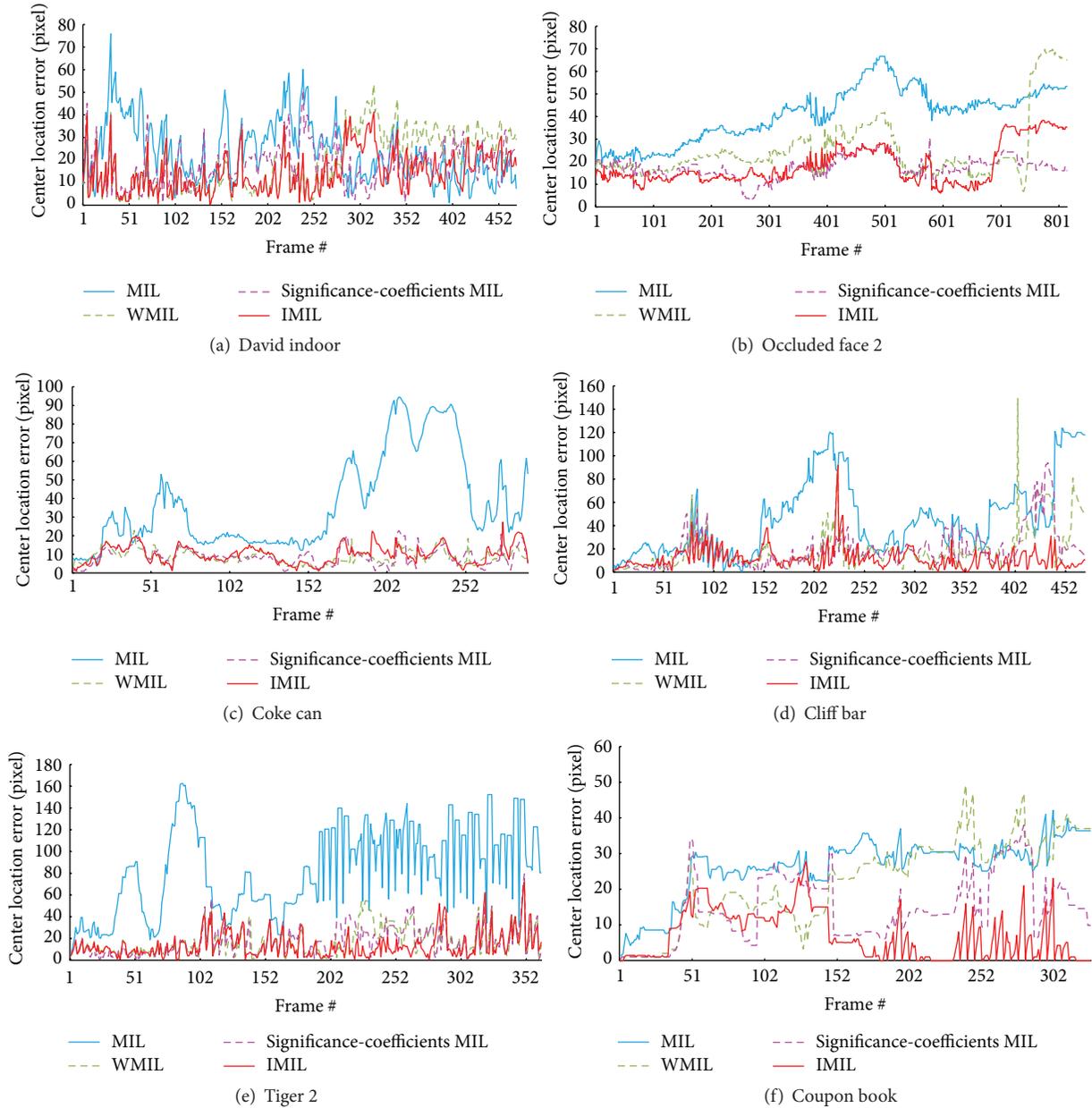


FIGURE 4: The center location errors for the six videos.

existing in the WMIL when there is fast moving (e.g., Coke can) in the tracking process.

**3.3. Quantitative Analysis.** We employ the center location error and overlap rate to evaluate the performance of our method. The center location error measures the position error between central locations of the tracking results and the centers of the ground truth. The overlap rate measures the tracking result on the area of overlap with ground truth bounding boxes [29]. The overlap rate is defined as  $\text{score} = \text{area}(R_T \cap R_G) / \text{area}(R_T \cup R_G)$ . The  $R_T$  is the area of the tracking result, while the  $R_G$  is for the ground truth bounding boxes. The tracking result with the overlap rate exceeding 50% is considered to be a correct detection.

The center location errors for all the trackers are shown in Figure 4. The average center location errors are detailed in Table 1. The smaller the average center location error is, the better the tracking algorithm performs. Bold indicates the best performance. The overlap rate for all the trackers are detailed in Table 2. Bold indicates the best performance. In the IMIL tracker, the bag probability is calculated according to instance probability. When there is tracking drift or fast moving, the bag probability mainly depends on the instance whose probability is the largest. Therefore, the drift problem can be corrected. Moreover, the learning rate is tuned according to the tracking result. As a result, the IMIL handles the problem of occlusion and variations on illumination and pose. Therefore, compared with the other MIL based trackers,

TABLE 1: The average center location errors for all of the trackers. Bold indicates the best performance.

Video clip	MIL	WMIL	Significance-coefficients MIL	IMIL
David indoor	23.631	19.435	17.716	<b>14.739</b>
Occluded face 2	33.805	22.165	15.747	<b>15.368</b>
Coke can	37.158	<b>8.720</b>	8.777	9.905
Cliff bar	43.994	18.225	18.882	<b>11.97</b>
Tiger 2	88.851	17.844	17.279	<b>13.972</b>
Coupon book	26.960	22.232	14.774	<b>7.179</b>

TABLE 2: The overlap rate for all of the trackers. Bold indicates the best performance.

Video clip	MIL	WMIL	Significance-coefficients MIL	IMIL
David indoor	0.65	0.84	0.90	<b>0.92</b>
Occluded face 2	0.78	0.71	0.78	<b>0.85</b>
Coke can	0.45	0.72	0.68	<b>0.86</b>
Cliff bar	0.53	0.64	0.75	<b>0.82</b>
Tiger 2	0.36	0.59	0.67	<b>0.79</b>
Coupon book	0.75	0.86	<b>0.88</b>	0.87

the IMIL tracker is robust to the problem of drift, occlusion, illumination variations, and pose changes. The experimental results show that IMIL algorithm performs well.

**3.4. Computational Cost.** In this section, we compared our method with MIL, WMIL, and significance-coefficients MIL in terms of computational cost. The average computing time processing an image measures the computational cost of these algorithms. The average computing time is defined as  $T = T_{\text{all}}/N_{\text{frames}}$ .  $T_{\text{all}}$  is the total computing time processing the whole video sequence.  $N_{\text{frames}}$  is the number of frames in the sequences.  $T$  is the obtained average computing time. The computational cost of these methods is affected by three main factors: the strategy for selecting weak classifiers, the number of weak classifiers, and the number of the selected weak classifiers for generating a strong classifier. The IMIL tracker is in the lowest computational load due to its advantages as follows: (1) the proposed criterion for selecting weak classifiers avoids computing the instance probabilities and bag probability  $M$  times before choosing one weak classifier; (2) about 120 weak classifiers are learned and 30 weak classifiers are chosen for generating a strong classifier (for MIL and WMIL, the number of the learned weak classifiers is 150 and that of the selected classifiers is 50). The average computational cost for different algorithms conducted on the 6 video clips is shown in Table 3. The results show that the IMIL is improved with lower computational time.

TABLE 3: The average computational cost (s) for different algorithms conducted on 6 video clips.

Video clip	MIL	WMIL	Significance-coefficients MIL	IMIL
David indoor	1.122	0.109	1.021	0.084
Occluded face 2	1.601	0.104	1.108	0.087
Coke can	1.089	0.104	0.81	0.104
Cliff bar	1.092	0.098	0.962	0.079
Tiger 2	1.092	0.103	1.096	0.094
Coupon book	1.164	0.103	0.978	0.086

## 4. Conclusion

In this paper, we presented an improved online multiple instance learning algorithm for visual tracking. A feedback scheme was used to update the parameters of the classifiers, which can handle the appearance changes caused by pose, illumination, and occlusion. We equally summed the instance probabilities to generate a bag probability. The method can avoid introducing the information from background and yield superior performance. A more efficient criterion was proposed to select weak classifiers, which avoided computing the instance probabilities and the bag probabilities  $M$  times after selecting one weak classifier. Finally, numerous experiments on challenging video sequences demonstrated that the proposed algorithm performs well in terms of efficiency, accuracy, and robustness.

## Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

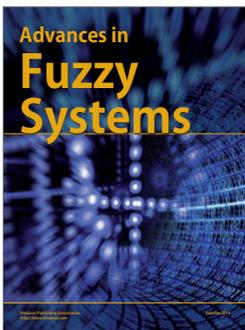
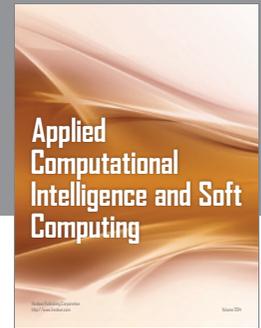
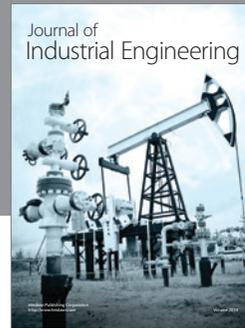
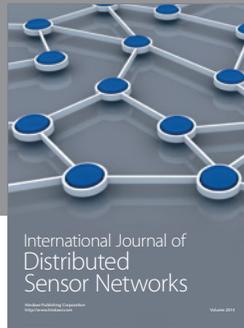
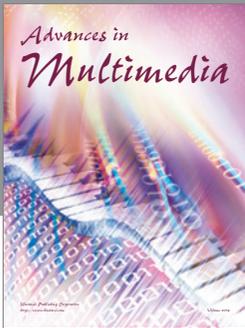
## Acknowledgment

The authors would like to thank the Hebei Province Science and Technology Support Program: 14275601D.

## References

- [1] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Computing Surveys*, vol. 38, no. 4, 2006.
- [2] Y. Motai, S. Kumar Jha, and D. Kruse, "Human tracking from a mobile agent: optical flow and Kalman filter arbitration," *Signal Processing: Image Communication*, vol. 27, no. 1, pp. 83–95, 2012.
- [3] Q. Zhao and H. Tao, "Object tracking using color correlogram," in *Proceedings of the 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (PETS '05)*, pp. 263–270, Beijing, China, October 2005.
- [4] A. H. Mazinan and A. Amir-Latifi, "Applying mean shift, motion information and Kalman filtering approaches to object tracking," *ISA Transactions*, vol. 51, no. 3, pp. 485–497, 2012.
- [5] Z. Zivkovic and B. Kröse, "An EM-like algorithm for color-histogram-based object tracking," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, pp. 1798–1803, IEEE, July 2004.

- [6] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)*, vol. 1, pp. 798–805, IEEE, New York, NY, USA, June 2006.
- [7] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.
- [8] E. J. Candes and T. Tao, "Near-optimal signal recovery from random projections: universal encoding strategies," *IEEE Transactions on Information Theory*, vol. 52, no. 12, pp. 5406–5425, 2006.
- [9] H. Cheng, Z. Liu, L. Yang, and X. Chen, "Sparse representation and learning in visual recognition: theory and applications," *Signal Processing*, vol. 93, no. 6, pp. 1408–1425, 2013.
- [10] H. Li, C. Shen, and Q. Shi, "Real-time visual tracking using compressive sensing," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '11)*, pp. 1305–1312, IEEE, Providence, RI, USA, June 2011.
- [11] S. H. Zhang, H. Yao, H. Zhou, X. Sun, and S. H. Liu, "Robust visual tracking based on online learning sparse representation," *Neurocomputing*, vol. 100, pp. 31–40, 2013.
- [12] F. Chen, Q. Wang, S. Wang, W. Zhang, and W. Xu, "Object tracking via appearance modeling and sparse representation," *Image and Vision Computing*, vol. 29, no. 11, pp. 787–796, 2011.
- [13] A. D. Jepson, D. J. Fleet, and T. F. El-Maraghi, "Robust online appearance models for visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 10, pp. 1296–1311, 2003.
- [14] B. Babenko, M.-H. Yang, and S. Belongie, "Robust object tracking with online multiple instance learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 8, pp. 1619–1632, 2011.
- [15] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via on-line boosting," in *Proceedings of the British Machine Vision Conference (BMVC '06)*, Edinburgh, UK, September 2006.
- [16] K. Zhang, L. Zhang, and M. H. Yang, "Real-time compressive tracking," in *Proceedings of the 12th European Conference on Computer Vision (ECCV '12)*, pp. 864–877, ACM, Florence, Italy, October 2012.
- [17] H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *Proceedings of the 10th European Conference on Computer Vision (ECCV '08)*, pp. 234–247, Marseille, France, October 2008.
- [18] P. Viola, J. C. Platt, and C. Zhang, "Multiple instance boosting for object detection," in *Proceedings of the Neural Information Processing Systems Conference (NIPS '05)*, pp. 1417–1426, 2005.
- [19] P. Dollar, B. Babenko, S. Belongie, P. Perona, and Z. Tu, "Multiple component learning for object detection," in *Computer Vision—ECCV 2008*, vol. 5303 of *Lecture Notes in Computer Science*, pp. 211–224, Springer, Berlin, Germany, 2008.
- [20] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learning," in *Proceedings of the 16th Annual Neural Information Processing Systems Conference (NIPS '02)*, IEEE, December 2002.
- [21] C. Galleguillos, B. Babenko, A. Rabinovich, and S. Belongie, "Weakly supervised object recognition and localization with stable segmentations," in *Proceedings of the 10th European Conference on Computer Vision (ECCV '08)*, Marseille, France, October 2008.
- [22] S. Vijayanarasimhan and K. Grauman, "Keywords to visual categories: multiple-instance learning for weakly supervised object categorization," in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, Anchorage, Alaska, USA, June 2008.
- [23] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, vol. 89, no. 1-2, pp. 31–71, 1997.
- [24] K. Zhang and H. Song, "Real-time visual tracking via online weighted multiple instance learning," *Pattern Recognition*, vol. 46, no. 1, pp. 397–411, 2013.
- [25] C. Xu, W. Tao, Z. Meng, and Z. Feng, "Robust visual tracking via online multiple instance learning," *Pattern Recognition*, vol. 48, no. 12, pp. 3917–3926, 2015.
- [26] T. Zhou, Y. Lu, and M. Qiu, "Online visual tracking using multiple instance learning with instance significance estimation," <http://arxiv.org/abs/1501.04378v1>.
- [27] Q.-H. Zhou, H. Lu, and M.-H. Yang, "Online multiple support instance tracking," in *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition and Workshops (FG '11)*, pp. 545–552, IEEE, Santa Barbara, Calif, USA, March 2011.
- [28] X. Xu and E. Frank, "Logistic regression and boosting for labeled bags of instances," in *Proceedings of the 8th Pacific-Asia Conference, Advances in Knowledge Discovery and Data Mining Lecture Notes in Computer Science (PAKDD '04)*, H. Dai, R. Srikant, and C. Zhang, Eds., pp. 272–281, Sydney, Australia, May 2004.
- [29] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.



**Hindawi**

Submit your manuscripts at  
<http://www.hindawi.com>

