Pattern Recognition Letters 33 (2012) 1075–1082

Contents lists available at SciVerse ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Online multiple instance gradient feature selection for robust visual tracking

Yuan Xie [a], Yanyun Qu [b,*], Cuihua Li [b], Wensheng Zhang [a]

[a] State Key Lab. of Intelligent Control and Management of Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[b] Video and Image Lab., Department of Computer Science, Xiamen University, Xiamen 361005, China

ARTICLE INFO

Article history:
Received 17 May 2011
Available online 10 February 2012
Communicated by A. Fernandez-Caballero

Keywords:
Gradient-based feature selection
HOG
Multiple Instance Learning
Online object tracking

ABSTRACT

In this paper, we focus on learning an adaptive appearance model robustly and effectively for object tracking. There are two important factors to affect object tracking, the one is how to represent the object using a discriminative appearance model, the other is how to update appearance model in an appropriate manner. In this paper, following the state-of-the-art tracking techniques which treat object tracking as a binary classification problem, we firstly employ a new gradient-based Histogram of Oriented Gradient (**HOG**) feature selection mechanism under Multiple Instance Learning (**MIL**) framework for constructing target appearance model, and then propose a novel optimization scheme to update such appearance model robustly. This is an unified framework that not only provides an efficient way of selecting the discriminative feature set which forms a powerful appearance model, but also updates appearance model in online **MIL** Boost manner which could achieve robust tracking overcoming the drifting problem. Experiments on several challenging video sequences demonstrate the effectiveness and robustness of our proposal.

1. Introduction

Visual tracking is a challenge problem in computer vision. It is well known that a good appearance model is very important for robust and efficient tracking. However, it is difficult to design a good appearance model because object target exhibits significant appearance change. Many tracking methods employ static appearance model such as works (Adam et al., 2006; Comaniciu et al., 2000; Isard and Maccormick, 2001; Avidan, 2001), these methods tend to fail when the appearance of the objects change significantly. As a result, there is the need for an adaptive appearance model to cope with appearance change during tracking. Therefore, in this paper we focus mainly on the following two points: (1) How to design an efficient appearance model. (2) How to update the appearance model in the online manner robustly.

With respect to modeling the object appearance, many works prefer to design adaptive appearance model using the current information both from the object and the background (Grabner et al., 2006; Grabner and Bischof, 2006; Jepson et al., 2003; Wang et al., 2005). Such way refers to treat object tracking as a binary classification problem which trains a model to separate the object from the background via a discriminative classifier.

Feature selection under boosting framework has been initially introduced by Tieu and Viola (2000) in the context of image retrieval. Then, Viola and Jones (2001) applied boosting feature selection to robust and fast object detection task, this was a seminal work that bridged the gap of weak learner design in boosting and feature selection step, paved the way of boosting in the area of computer vision, e.g., Opelt et al. (2004), Torralba et al. (2005), Yang et al. (2004). Those works take an exhaustive feature selection scheme over a very large hypothesis space. However, with the online constraint (Avidan, 2007; Grabner et al., 2006; Collins et al., 2005), the exhaustive feature selection over large feature space is strictly prohibited. To solve this problem, Grabner and Bischof (2006) proposed a novel feature selection method where a set of selectors were constructed to choose the feature by minimizing the training error from random guess feature pool.[1] But, it only picks up the most discriminative feature from such feature pool, if those features in the pool are less powerful, such scheme is far from efficiency. Liu and Yu (2007) propose a novel feature selection scheme for online boosting based on the gradient descent mechanism. The approach iteratively updates the features (Histogram of Oriented Gradient (Dalal and Triggs, 2005), but not limited to it) in a gradient descent manner. It seems that gradient feature selection is a much more efficient scheme of learning discriminative features.

The traditional tracking under boosting framework is a supervised learning method, therefore sampling the object and background is a critical step for updating the appearance model. Nevertheless, the inaccuracy in samples will degrade the

* Corresponding author. Tel.: +86 010 82614489; fax: +86 010 62545229.
E-mail addresses: yuan.xie@ia.ac.cn (Y. Xie), yyqu@xmu.edu.cn (Y. Qu), chli@xmu.edu.cn (C. Li), wensheng.zhang@ia.ac.cn (W. Zhang).

[1] The size of such random guess feature pool is relatively small comparing with the large feature space.

0167-8655/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2012.01.020

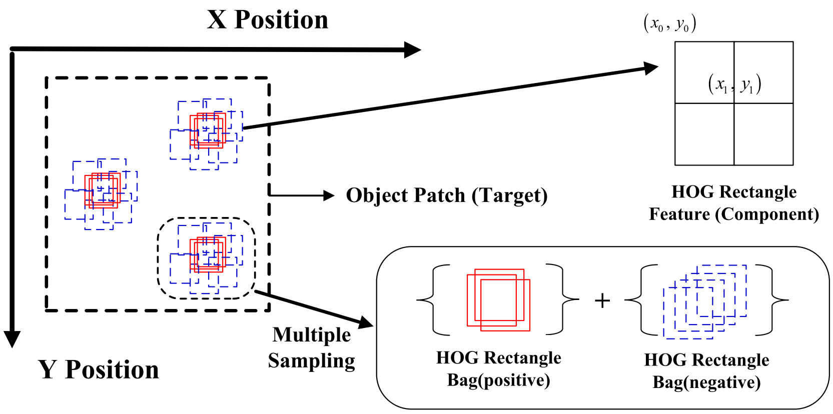

X Position
Y Position
(x_0, y_0)
(x_1, y_1)
Object Patch (Target)
HOG Rectangle Feature (Component)
Multiple Sampling
HOG Rectangle Bag(positive)
HOG Rectangle Bag(negative)

Fig. 1. Image representation and object appearance model.

appearance model and cause drifting. Thus, Viola et al. (2005) and Babenko et al. (2009) introduce the use of a Multiple Instance Learning (**MIL**) (Dietterich et al., 1997) for object detection and tracking. In fact, many applications of concept learning, unambiguously labeled positive and negative samples are not easily available. For example, in the context of object detection, a positive bag could contain a few possible bounding boxes around each labeled object without knowing which one is true "correct", therefore the ambiguity is passed to the **MIL** learning process. Then, **MIL** could handle such ambiguity by minimizing the negative log likelihood of training bags, so a more robust learner could be achieved. However, **MIL** based tracking (Babenko et al., 2009) still employs the exhaustive feature selection mechanism to form the adaptive appearance model which takes the negative influence on the power of tracking system.

This paper proposes a gradient-based feature selection unifying with online Multiple Instance Learning (**MIL**) approach for robust object tracking. The first contribution is the introduction of a novel discriminative object appearance model, such model consists of **HOG** rectangle features and their corresponding feature bags from positive and negative samples. The second contribution is the proposal of an *optimization scheme* updating appearance model robustly. It iteratively updates each feature using gradient descent and **MIL** approaches by maximizing the likelihood of training **HOG** feature bags. The proposed method not only provides an efficient way of building a discriminative appearance model, but also updates model robustly to drifting problem which traditional supervised learning unavoidable encounters. We present the empirical results of our method comparing with several state-of-the-art tracking algorithms on standard challenging video sequences, experimental results show that our method can lead to a more robust and stable tracker than state-of-the-art methods.

The rest of the paper is organized as follows: Section 2 gives an overview of the proposed tracking method. Then we give a short review of gradient descent feature selection and the online Multiple Instance Learning in Sections 3 and 4. Section 5 describes a novel online multiple instance gradient feature selection approach for object tracking. The experimental results and some discussions are reported on Section 6. Finally, we concludes the paper and outlines the direction of the future work on Section 7.

2. System overview

Usually, an object tracking system contains three components: image representation, object appearance model and motion model. We employ Histogram of Oriented Gradient (**HOG**) (Dalal and Triggs, 2005; Laptev, 2004) rectangle feature as the image representation to describe the components[2] of tracking object. Fig. 1 show the structure of **HOG** feature and we will discussed in more detail in Section 3.1. Our appearance model adopts the philosophy of representing object as an assembly of component (Fergus et al., 2005; Leibe et al., 2008; Kwon and Mu Lee, 2009). But unlike to such method, we use the bag of **HOG** rectangles to model a component, and apply boosting framework to combine the **HOG** components into a strong discriminant classifier, which is able to return $p(y=1|x)$ ($p(y|x)$ for short) where x is an image patch[3] and y is a binary variable indicating whether the x is the target. For motion model, supposed at time step $t-1$, our tracker maintains the object location l^*_{t-1}. Then, the simple distribution of target's location l^*_t at t step is:

$$p(l^*_t|l^*_{t-1}) \propto \begin{cases} 1 & \text{if } \|l^*_t - l^*_{t-1}\| < s, \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

where s is the radius of search region. The overview of our tracking system is summarized in **Procedure**. The workflow of our tracking system is similar to Algorithm 1 in the work (Babenko et al., 2009) with differences of central steps such as appearance model and its updating scheme (e.g., step 2 and 3). Supposed we are going to locate the target in the new frame at time step t. The method crops a set of image patch candidates within search region s, then computes their **HOG** components' feature vectors efficiently via Integral Histogram. The proposed method achieves the new location of target by means of **Online Gra-MIL** classifier $\boldsymbol{F}(x)$ (Section 5) trained in previous step, as can be shown from step 3 and 4 in **Procedure**. Accordingly, for each **HOG** component, we can crop the positive and negative **HOG** rectangle bags from positive patches bag $\boldsymbol{X}^{\gamma}$ and negative patches bag $\boldsymbol{X}^{\gamma,\beta}$ respectively at certain position and specified size[4] (step 5 in **Procedure**). Then each **HOG** component will be updated via a new optimization schema which employs gradient feature selection to maximize the likelihood of the current **HOG** feature bags. Consequently, the whole appearance model could be updated under the boosting framework, and the final strong classifier $\boldsymbol{F}(x)$ can be used to the next frame $t+1$ (see more details in Section 5).

[2] We call **HOG** rectangle features as components rather than parts, because they are not semantic parts of object like arms or legs of human.

[3] In fact, x is usually the representation of an image patch in feature space.

[4] The position and size are the variables which are the elements of parameter of **HOG** feature classifier, see more details in Section 3.

```
Procedure: The workflow of the proposed tracking system ()
    Input: New video frame number $t$
1   Crop out a set of image patches, $\boldsymbol{X}^s = \{x | s > \|l(x) - l^*_{t-1}\|\}$;
2   Compute feature vector at the positions of several HOG rectangle components for each candidate patch in $\boldsymbol{X}^s$;
3   Use Online Gra-MIL classifier $\boldsymbol{F}(\boldsymbol{x})$ to estimate $p(y = 1|x)$ for $x \in \boldsymbol{X}^s$;
4   Update tracker location $l^*_t = l(\mathrm{argmax}_{x \in \boldsymbol{X}^s} p(y|x))$;
5   Crop out two bags of image patches
    $\boldsymbol{X}^\gamma = \{x | \gamma > \|l(x) - l^*_t\|\}$ and $\boldsymbol{X}^{\gamma,\beta} = \{x | \beta > \|l(x) - l^*_t\| > \gamma\}$,
    then the HOG rectangle bags for each component can be attained;
6   Update object appearance model using the positive patches bag $\boldsymbol{X}^\gamma$ and negative patches bags $\boldsymbol{X}^{\gamma,\beta}$.
```

3. Gradient feature selection

3.1. Feature and component classifier

The Histogram of Oriented Gradient (**HOG**) feature shows greater capability of description than Haar-like feature, both of them can be computed efficiently through integral method (Integral Image (Viola and Jones, 2001) and Integral Histogram (Fatih, 2005)). Therefore **HOG** feature is widely applied in object detection and tracking application. Liu and Yu (2007) used **HOG** feature with only one block which contains 2 × 2 cells (Fig. 1). For each cell, the 9 bins histogram of gradient magnitude at each orientation are computed, so the total dimensions of **HOG** feature are 36. Note that the **HOG** can be parameterized by (x_0,y_0,x_1,y_1), where (x_0,y_0) and (x_1,y_1) are the two corners (top-left and bottom-right) of the first cell.

The component classifier according to **HOG** feature can be achieved by using Linear Discriminative Analysis (**LDA**) which inherits the idea of boosted histogram proposed by Laptev (2004). The LDA is applied to the histogram features of positive and negative samples, and results in the optimal projection direction β and threshold θ. Therefore the component classifier can be presented as follow:

$$f_m(x;\boldsymbol{p_m}) = \frac{2}{\pi}\arctan\left(\beta^T h(x_0,y_0,x_1,y_1) - \theta\right), \tag{2}$$

where x denotes the candidate image patch. The parameters of component classifier $\boldsymbol{p} = [x_0,y_0,x_1,y_1,\beta,\theta]^T$, the histogram features $h(\cdot)$ are computed from all training data via integral histogram. We use the arctan $(\cdot)$ function because of its derivability with respect to the parameters $\boldsymbol{p}$.

3.2. Feature selection

Then the feature selection can be seen as a process of updating the parameters of each weak classifier. Therefore, it is natural to use the weighted least square error (**WLSE**) as the objective function for feature updating:

$$\min_{\boldsymbol{p}} \varepsilon(f(x;\boldsymbol{p})) = \min_{\boldsymbol{p}} \sum_{i=1}^{K} w_i (f(x;\boldsymbol{p}) - y_i)^2, \tag{3}$$

where y_i denotes the label of the ith training image patch. Substituting Eq. (2) into Eq. (3), the function to be minimized is:

$$\varepsilon(\boldsymbol{p}) = \sum_{i=1}^{K} w_i \left(\frac{2}{\pi}\arctan\left(\beta^T h(x_0,y_0,x_1,y_1) - \theta\right) - y_i\right)^2, \tag{4}$$

The work (Liu and Yu, 2007) chooses to use the gradient descent method to solve this problem iteratively. Taking the derivative with respect to $\boldsymbol{p}$ gives:

$$\frac{d\varepsilon}{d\boldsymbol{p}} = \sum_{i=1}^{K} 2w_i(f(x_i) - y_i)\frac{df_i}{d\boldsymbol{p}}, \tag{5}$$

where $\frac{df_i}{d\boldsymbol{p}} = \left[\frac{\partial f_i}{\partial x_0}\ \frac{\partial f_i}{\partial y_0}\ \frac{\partial f_i}{\partial x_1}\ \frac{\partial f_i}{\partial x_1}\ \frac{\partial f_i}{\partial y_1}\ \frac{\partial f_i}{\partial \beta}\ \frac{\partial f_i}{\partial \theta}\right]^{\boldsymbol{T}} \cdot \frac{df_i}{d\boldsymbol{p}}$ can be denoted as follows:

$$\begin{aligned}
\frac{df_i}{dz} &= \frac{2}{\pi}\frac{\beta^{\boldsymbol{T}}\frac{\partial h_i}{\partial z}}{1 + \left(\beta^{\boldsymbol{T}} h_i - \theta\right)^2}, \quad z = x_0, y_0, x_1, y_1, \\
\frac{df_i}{d\beta} &= \frac{2}{\pi}\frac{h_i}{1 + \left(\beta^{\boldsymbol{T}} h_i - \theta\right)^2}, \\
\frac{df_i}{d\theta} &= \frac{2}{\pi}\frac{-1}{1 + \left(\beta^{\boldsymbol{T}} h_i - \theta\right)^2},
\end{aligned} \tag{6}$$

where $\frac{\partial h_i}{\partial z}$, $z = x_0, y_0, x_1, y_1$ can be computed using the integral histogram method. For more details about the expansion of $\frac{\partial h_i}{\partial z}$, please refer to the work (Liu and Yu, 2007) in order to acquire deeply investigation. Then, the new **HOG** rectangle feature and its corresponding classifier can be achieved:

$$\boldsymbol{p}_{new} = \boldsymbol{p} - \left(\frac{d\varepsilon}{d\boldsymbol{p}}\right)\bigg|_{\boldsymbol{p}}.$$

4. Online Multiple Instance Learning

The traditional supervised discriminative learning requires a training set $\{(x_1,y_1),\ldots,(x_n,y_n)\}$ containing N instances, where $x_i \in \boldsymbol{X}$ is an instance (for example the feature vector computed for an image patch) and $y_i \in \boldsymbol{Y} = \{-1,1\}$ is the corresponding known label. The task is to learn a classification function $f{:}\boldsymbol{X} \to \boldsymbol{Y}$. In the *Multiple Instance Learning* framework the training set D consists of N bags $D = \{X_i, y_i\}_{i=1}^{N}$, where $X_i = \{x_{i1},x_{i2},\ldots,x_{ij}\}$ and y_i is the bag label. The bag labels are defined as:

$$y_i = \max_j (y_{ij}), \tag{7}$$

where y_{ij} are the instance label, which are assumed to exist but are not known during training phase. In other words, a bag is positive if it contains at least one positive instance, a negative bag means that all instances in the bag are negative. The goal is to learn a classification function that predicts the labels of unseen instances and/or bags.

A lots of algorithms have been proposed for solving the **MIL** problem so far (Viola et al., 2005; Dietterich et al., 1997; Andrews et al., 2003; Babenko et al., 2009). Among them, most closely related to our work are MILBoost and Online MILBoost proposed by Viola et al. and Babenko et al. respectively. They all use the boosting framework to train a boosting classifier to minimize the negative log likelihood of bags instead of maximizing likelihood of bags:

$$L = -\sum_i (\log p(y_i|X_i)), \tag{8}$$

where the $p(y_i|X_i)$ is the posterior probability of the bag and we mark it with p_i for short. Moreover, we should define $p(y_i|x_{ij})$ the posterior probability of an instance at bag X_i and take the p_{ij} for short, note that exact definition of p_{ij} will depend on the classifier. Similar to the definition of the bag label, the connection between bag probability p_i and the probability of its instance p_{ij} can be achieved by following:

$$p_i = \max_j \{p_{ij}\}. \tag{9}$$

An important observation is that the max operator is not differentiable. Luckily, several differentiable approximations to the max operator exist in the literature. In (Viola et al., 2005) the **Noisy-OR** (**NOR**) model is adopted for doing this:

$$p_i = 1 - \prod_j (1 - p_{ij}). \tag{10}$$

The Online MILBoost iteratively minimizes the negative log likelihood of training bags by choosing the most discriminative feature from a large feature pool in each boosting training phase, see Babenko et al. (2009) for more details.

5. Online multiple instance gradient feature selection

5.1. Component classifier

The proposed algorithm requires component classifier *f* that can be updated in online fashion. Recall the Section 3 that we employ the **HOG** features whose structure are described in Fig. 1 for feature selection. Their corresponding component classifiers can be represented as follows:

$$f_m(x; \boldsymbol{p_m}) = \frac{2}{\pi} \arctan\left(\beta^T h(x_0, y_0, x_1, y_1) - \theta\right), \quad m = 1, \ldots, M,$$

where (x_0,y_0) and (x_1,y_1) are the two corners of a cell, β and θ can be initialized by **LDA** projection, M denotes the number of components for representing the target. The updating rule of those parameters are derived from Eq. (3).

In our case, we are given training data bags $\{(X_1,y_1), (X_2,y_2),\ldots\}$, where $X_i = \{x_{i1},x_{i2},\ldots,x_{ij}\}$, x_{ij} denotes an image patch. For each component, we will get its corresponding training feature bags by the part of parameter (x_0,y_0,x_1,y_1) in $\boldsymbol{p_m}$. Then the proposed approach would like to iteratively optimize each **HOG** component in gradient descent manner and update estimate of $p(y_i|x_{ij})$ (instance probability) to minimize the negative log likelihood of bags (Eq. (8)). In order to measure the contrast between confidence that one sample would be classified as positive or negative, we use $h_m(x)$ (the log odd ratio of weak classifier) to model the instance probability instead of $f_m(x;\boldsymbol{p})$ itself directly. The $h_m(x)$ can be represented as:

$$h_m(x) = \log\left[\frac{p(y = 1|f_m(x))}{p(y = -1|f_m(x))}\right], \tag{11}$$

where $p(y = 1|f_m(x))$ measures the likelihood of x belongs to positive sample, $p(y = -1|f_m(x))$ denotes the confidence to negative one. Such confidences can be model by logistic regression:

$$\begin{aligned} p(y = 1|f_m(x)) &= \frac{e^{f_m(x)}}{e^{f_m(x)} + e^{-f_m(x)}}, \\ p(y = -1|f_m(x)) &= \frac{e^{-f_m(x)}}{e^{f_m(x)} + e^{-f_m(x)}}. \end{aligned} \tag{12}$$

Then the instance probability can be modeled as:

$$p(y_i|x_{ij}) = \sigma\left(H_{m-1}(x_{ij}) + h_m(x_{ij})\right) = \frac{1}{1 + e^{-\left(H_{m-1}(x_{ij}) + h_m(x_{ij})\right)}}, \tag{13}$$

where $\sigma(\cdot)$ is sigmoid function, $H_{m-1}(x)$ is sum of the log odd ratio of the previous component classifiers. Finally, the bag probability $p(y_i|X_i)$ are modeled using **Noisy-OR** model in Eq. (10).

5.2. Proposed algorithm

We propose a novel tracking algorithm that builds a discriminative appearance model of object target by gradient based feature selection and updates such model in a robust way applying **MIL**, then optimizes them under the online boosting framework. It inherits the essential idea from gradient feature selection for online boosting (Liu and Yu, 2007) and online MIL tracking (Babenko et al., 2009). The approach not only could avoid exhaustive feature selection by using gradient feature selection instead, but also can handle the ambiguity that training example passed to learning algorithm when sampling the positive and negative examples. We refer such method as the*online multiple instance gradient feature selection*, and **Online Gra-MIL** for short.

Algorithm 1. Online Gra-MIL feature selection

```
  Input: DataSet $\{X_i, y_i\}_{i=1}^N$, where $X_i = \{x_{i1}, x_{i2}, \ldots\}$,
  $y_i \in \{-1, 1\}$, and an initial set of component classifiers
  $\{f_m(\boldsymbol{p_m}); m \in [1, M]\}$,
  where $f_m(x; \boldsymbol{p_m}) = \frac{1}{\pi}\arctan(\beta^T h(x_0, y_0, x_1, y_1) - \theta)$
  Output: An updated strong classifier $\boldsymbol{F}(\boldsymbol{x})$
1  Initialize weights $w_i = \frac{1}{K}$ and $\boldsymbol{F}(\boldsymbol{x}) = 0$;
2  for $m = 1$ to $M$ do
3      Get the HOG rectangle bags through $\boldsymbol{p_m}$ and $X_i$ for current component;
4      Compute $\varepsilon(\boldsymbol{p_m})$ and $\frac{d\varepsilon}{d\boldsymbol{p_m}}|_{\boldsymbol{p_m}}$ using Eqs. (3) and (4), where
       $\varepsilon(\boldsymbol{p}_m) = \sum_{i=1}^K w_i \left(\frac{2}{\pi}\arctan(\beta^T h(x_0, y_0, x_1, y_1) - \theta) - y_i\right)^2$;
5      $p_{ij}^m = \sigma(H_{m-1} + h_m(x_{ij}))$;
6      $p_i^m = 1 - \Pi_i\left(1 - p_{ij}^m\right)$;
7      $L^m = -\left(\sum_{i|y_i=1} \log(p_i^m) + \sum_{i|y_i=-1} \log(p_i^m)\right)$;
8      if $L^m$ is decreasing then
9          Update $\boldsymbol{p}_m = \boldsymbol{p}_m - \left(\frac{d\varepsilon}{d\boldsymbol{p}_m}\right)|_{\boldsymbol{p}_m}$;
10         go to step 3;
11     end
12     Update $F(x) = F(x) + f_m(x; \boldsymbol{p}_m)$;
13     Update the weights by $w_i = w_i e^{-y_i f_m(x_i)}$ and normalize the weights such that $\sum_{i=1}^K w_i = 1$;
14 end
15 Return The strong classifier
   $\mathrm{sign}[\boldsymbol{F}(\boldsymbol{x})] = \mathrm{sign}\left[\sum_{m=1}^M f_m(x; \boldsymbol{p_m})\right]$;
```

Note that this work mainly focus on investigation of appearance model, we employ a sample motion model where the location of tracker at time $t + 1$ is equally likely to appear within the neighborhood of the tracker location at time t, as it depicts by Eq. (1). Algorithm 1 presents the pseudo-code of the Online Gra-MIL. Suppose that we are given an initial feature set containing M **HOG** components feature which compute from two patches bags $\boldsymbol{X}^\gamma$ and $\boldsymbol{X}^{\gamma,\beta}$ (thus, in our experiment each component classifier has two **HOG** rectangle bags for training.). Note that although examples are passed by means of bags, the component classifier in **MIL** are instance classifiers, so require instance labels y_{ij}. Since the instance labels are supposed unavailable in **MIL**, we pass bag label y_i for all instances x_{ij} in this bag to the weak training procedure. For each component classifier $f_m(\boldsymbol{p}_m)$, the online boosting iteratively updates the feature parameters $\boldsymbol{p}_m$ according to the above computed

Table 1
Average center location errors (pixels).

Vide clip	OAB	Fragment	Ensemble	Gra-MIL	MIL	TLD
David Indoor	25	68	95	9	19	20
Sylvester	13	24	80	5	11	55
Face Occlusion	28	29	34	13	11	16
Tiger1	42	47	77	9	20	4/NaN[a]
Girl	56	29	123	17	28	19/NaN
Coke Can	5	35	33	10	17	11

[a] NaN denotes the tracker loses the target for several frames during tracking.

gradient $\frac{d\varepsilon}{d\mathbf{p}_m}$. Because of our method using **MIL**, the selection criteria of component classifier is not minimizing the $\varepsilon(\mathbf{p}_m)$ but the negative log likelihood of training bags. The likelihood function L^m is computed at each iteration and expected to keep decreasing. The iteration will cease if likelihood arrives at a minimum or reduction smaller than a threshold. Then, the updating for the current **HOG** component will stop, such procedure is the same to the other components.

6. Experiments

6.1. Experimental setting

We applied the proposed **Online Gra-MIL** tracking algorithm to test on several challenging video sequences, all of which are publicly available. In addition, we test five other trackers on the same video sequences for comparison. They are Online-AdaBoost (**OAB** for short) (Grabner et al., 2006; Grabner and Bischof, 2006), Fragment Tracking (**FragTrack**) (Adam et al., 2006), Ensemble Tracking (**EnTrack**) (Avidan, 2007), MIL Tracker (**MIL**) (Babenko et al., 2009) and TLD Tracker (**TLD**) (Kalal et al., 2009; Kalal et al., 2010). The code of all those trackers are publicly available except **EnTrack**, thus we implement Ensemble Tracker which is true to original work (Avidan, 2007).

The goal of experiment is to validate our algorithm and demonstrate that the proposed method will lead to a more robust and stable tracker comparing with other tracking algorithms. For this reason all algorithm parameters are fixed for all the test video sequences. For **Online Gra-MIL** positive samples are cropped from all the positions within radius $\gamma = 4$ (pixel) while negative samples are cropped between γ and radius of search region by random sampling. The size of search region is twice as much as the size of target patch (Section 2, Eq. (1)). The number of component classifier M is set to 50 and the same to selectors in both **OAB** and **MIL**. Finally, the number of candidate weak classifiers in the feature pool will be set to 250 for **OAB** and **MIL**.

For **FragTrack** and **TLD**, we use the same parameters as the authors used in their works (Adam et al., 2006; Kalal et al., 2009) for all of our experiments. Some specially tuning will be applied in Ensemble Tracking. **EnTrack** uses a 11 dimensions feature vector that is formed by the combination of local orientation histogram and pixel colors (RGB channels). However all the experiments are tested on the gray scale video sequences, therefore we throw three color dimensions away leaving only local orientation histogram features to **EnTrack**. We acknowledge that such tuning may lead to worse performance than original, but it is a compromise to impartial comparison.

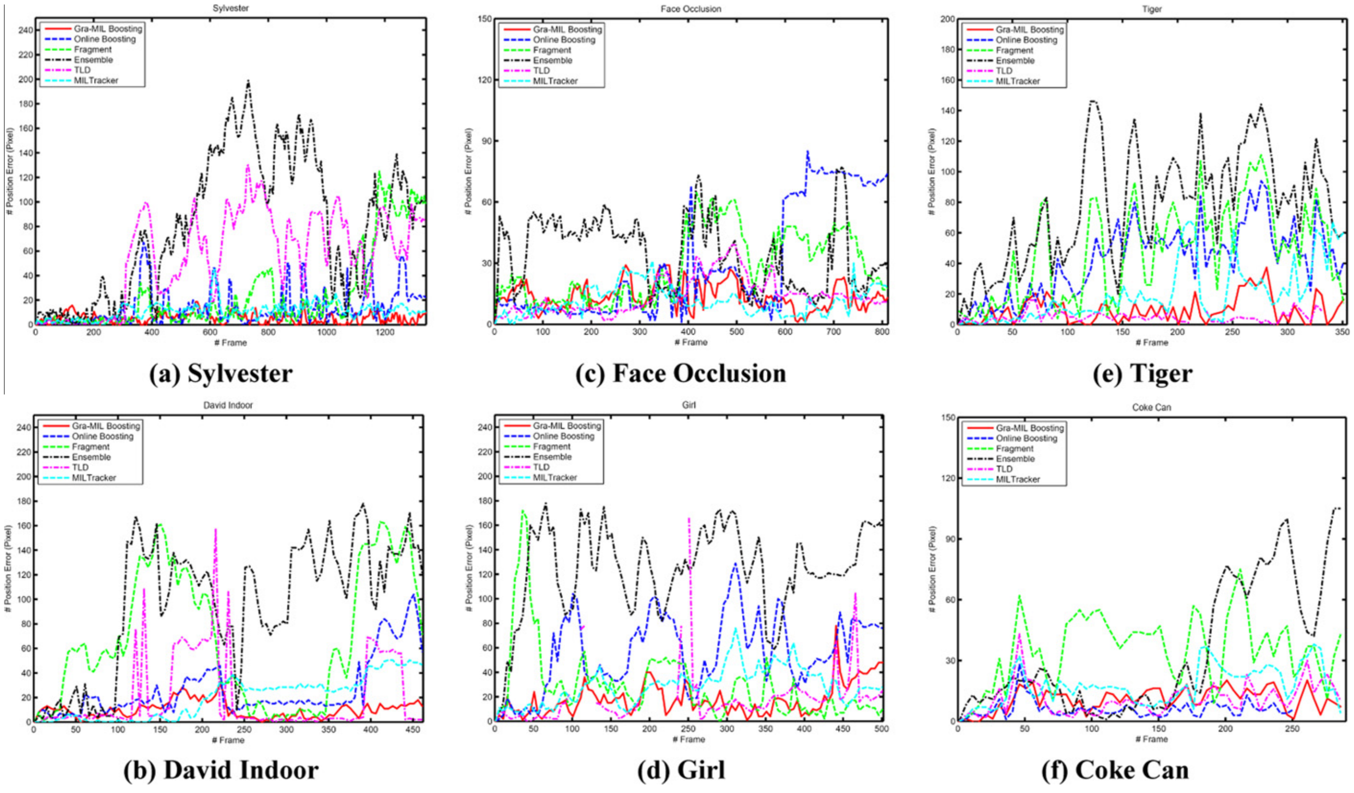
6.2. Quantitative and qualitative analysis

We perform experiments on 6 publicly available standard video sequences. The ground truth of the center position of target for all the sequences are labeled every five frames, such ground truth are provided by Babenko's work (Babenko et al., 2009). All the testing video frames are gray scale, and resized to 320 × 240 pixels. For quantitative analysis, we use average center location errors as evaluation criteria to compare performance, the pixel error in every frame is defined as follow:

$$\text{error} = \sqrt{(x' - x)^2 + (y' - y)^2}, \tag{14}$$

where (x', y') represents the object position given by tracker, (x, y) is the ground truth. The quantitative results are summarized in Table 1.

In the Table 1, each row represents average center location errors of six comparison algorithms testing on a certain video sequence. The number marked with red indicates the best performance in a certain testing video, blue indicates the second best. As



(a) Sylvester

(c) Face Occlusion

(e) Tiger

(b) David Indoor

(d) Girl

(f) Coke Can

Fig. 2. Error plots for all test sequences.

shown in Table 1, our proposed method acquires 3 bests and 3 s bests. Note that the NaN represents the TLD tracker loses the target for several frames in certain sequence. Moreover, we calculate the average center location error excluding those lost frames. For thorough investigation, we draw the error curve according to Eq. (14) for each video sequences (Fig. 2). In addition, Fig. 3 shows the screen captures for some of the video clips. More details of experiments will be discussed below.

6.2.1. Videos: Sylvester & David Indoor

These two video sequences are widely used in state-of-the-art tracking systems, such as works (Grabner et al., 2006; Babenko et al., 2009; Lin et al., 2004; Ross et al., 2008). They include challenging illumination change, scale and pose variety. Our proposed **Online Gra-MIL** achieves the best performance in both sequences while **MIL** get the second in David and Sylvester. Note that the average center location errors of our method are 9 and 5 pixels respectively, both of which are below 10 pixels. That is probably because **MIL** based **HOG** components are robust to lighting change and invariant to moderate scaling. The error curves of those two sequences are presented in Fig. 2(a) and (b), and video clips of David Indoor are shown in Fig. 3.

6.2.2. Videos: Face Occlusion & Girl

Face Occlusion video is designed to test whether a tracking algorithm can handle the partial occlusion and pose changes, e.g., object rotate in the plane or out of plane. As shown in Fig. 2(c), all the algorithms except **EnTrack** could achieve comparable performance in previous four hundreds frames. However, **FragTrack** performs poorly after that, because it could not update in the online fashion, which leads to failure when object appearance changes drastically. **OAB** owns the capability of online updating, but our method contains better feature selection scheme and Multiple Instance Learning which can handle sampling ambiguity and

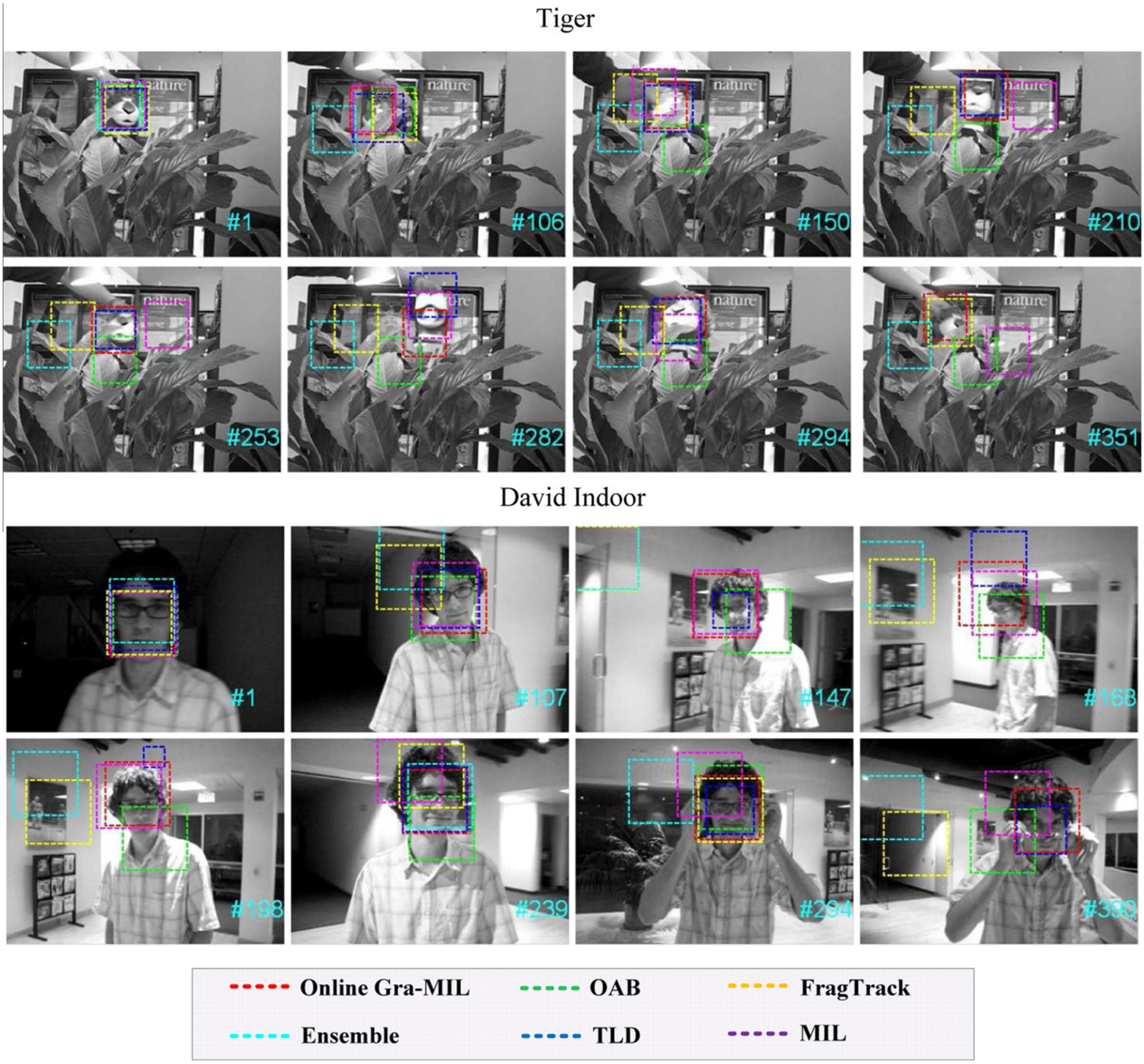


Fig. 3. Screenshots of tracking results on Tiger and David Indoor sequences.

error accumulation. As a result, **OAB** is drifting gradually after frame #600. **Online Gra-MIL**, **MIL** and **TLD** show the good performance in this sequence.

Girl sequence contains severe out of plane rotation and modest pose and scale changes. Only the results of our approach and **TLD** tracker could be comparable. However, **TLD** tracker fails to handle out of plane rotation, leading to lose the target several times, which correspond to the discontiguous line in Fig. 2(d) around the frame #100, #200 and #250. Since we do not include the lost case to calculate location error, **TLD** tracker gets the best in girl sequence. In other words, our approach would achieve more robust performance than **TLD** if the case of lose target was considered.

6.2.3. Videos: Tiger1 & Coke Can

Among all the testing videos in our experiments, Tiger includes most challenges such as frequent occlusions, motion blur, appearance drastically change (tiger opens mouth abruptly, rotates in the plane or out of plane) and illumination change. The performances of six methods are presented in Fig. 2(e) and Fig. 3, only **Online Gra-MIL** and **TLD** have most frame errors below 20 pixels while majority frame of other methods are above 40 pixels. Comparing the best two methods, the error plot of **TLD** is lower than that of the proposed **Online Gra-MIL**, probably due to **TLD** has learnt a robust detector to avoid drift which caused by random motion and drastic appearance change of target during tracking.

OAB wins the competition in Coke Can sequence. It achieves average location errors 5 pixels less than **Online Gra-MIL**. That is partly because Coke Can is a specular object whose gradient characteristic is inconspicuous, then the components of **Online Gra-MIL** can not describe such object efficiently. However, **OAB** loses the target completely after frame #255, which shows that our method is stable than **OAB**.

6.3. Discussion

In most cases our Online Gra-MIL algorithm outperforms **OAB**, **FragTrack** and **EnTrack**, and achieves performances more robust than **MIL** and **TLD**. The reason for such superior performance is that Online Gra-MIL algorithm not only can pick out most discriminative feature by gradient feature selection, but also be able to handle ambiguously labeled training examples, which are provided by tracker itself. Furthermore, **OAB** algorithm is a good tracker, but its bootstrap training might update using exhaustive feature selection with a sub-optimal positive example. Therefore it is far from efficient and over time the error may accumulate so as to degrade the model, and finally cause drifting. **MIL** tracker is also a great tracker which is more robust than **OAB**. However, the limitation of exhaustive feature selection degrades its performance to drifting. We notice that our proposed **Online Gra-MIL** and **TLD** are the comparable trackers which are particularly good at dealing with many challenges in tracking scenario, for example the Tiger sequence. In a word, our method can lead to an efficient and stable online tracker.

7. Conclusion and future work

In this paper we have presented a novel adaptive appearance model buildup and updating method under boosting framework called *online multiple instance gradient feature selection*. The proposed algorithm not only allows us to achieve an efficient way of updating the discriminative feature set using gradient feature selection scheme, but also could overcome drifting problem to some extent with the help of **MIL**. Our novel method is applied to several difficult standard videos, experiments demonstrate the efficiency and stability of our proposal. Future directions of this work include unifying the semi-supervised learning with **MIL** in order to further reduce the amount of drifting.

Acknowledgements

We would like to thank the editors and the anonymous reviewers for their helpful comments. This work is supported by National Key Technology R&D Program (Grant No. 0101050302), National Natural Science Foundation of China (Major Program, Grant No. 90924026), National Defense Basic Scientific Research program of China (Grant No. B1420110155) and the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20110121110020).

References

Tieu, K., Viola, P., 2000. Boosting image retrieval. In: Proc. Computer Vision and Pattern Recognition, pp. 228–235.

Viola, P., Jones, M., 2001. Rapid object detection using a boosted cascade of simple features. In: Proc. Computer Vision and Pattern Recognition, pp. 511–518.

Opelt, A., Fussenegger, M., Pinz, A., Auer, P., 2004. Weak hypotheses and boosting for generic object detection and recognition. In: Proc. European Conf. on Computer Vision, pp. 71–84.

Torralba, A., Murphy, K., Freeman, W., 2005. Sharing features: Efficient boosting procedures for multi-class object detection. In: Proc. Computer Vision and Pattern Recognition, pp. 762–769.

Yang, P., Shan, S., Gao, W., Li, S., Zhang, D., 2004. Face recognition using Adaboosted Gabor features. In: Proc. Automatic Face and Gesture Recognition, pp. 356–361.

Avidan, S., 2007. Ensemble tracking. IEEE Trans. Pattern Anal. Machine Intell., 261–271.

Grabner, H., Grabner, M., Bischof, H., 2006. Real-time tracking via online boosting. In: Proc. British Machine Vision Conference, pp. 47–56.

Grabner, H., Bischof, H., 2006. Online boosting and vision. In: Proc. Computer Vision and Pattern Recognition, pp. 260–267.

Liu, X., Yu, T., 2007. Gradient feature selection for online boosting. In: Proc. Internat. Conf. on Computer Vision, 2007.

Dalal, N., Triggs, B., 2005. Histograms of oriented gradient for human detection. In: Proc. Computer Vision and Pattern Recognition, pp. 886–893.

Viola, P., Platt, J.C., Zhang, C., 2005. Multiple instance boosting for object detection. In: Proc. Neural Information Processing Systems, pp. 1417–1426.

Babenko, B., Yang, M.H., Belongie, S., 2009. Visual tracking with online multiple instance learning. In: Proc. Computer Vision and Pattern Recognition.

Collins, R.T., Liu, Y., Leordeanu, M., 2005. Online selection of discriminative tracking features. IEEE Trans. Pattern Anal. Machine Intell., 1631–1643.

Dietterich, T.G., Lathrop, R.H., Perez, L.T., 1997. Solving the multiple instance problem with axis parallel rectangle. Artif. Intell., 31–71.

Andrews, S., Tsochantaridis, L., Hofmann, T., 2003. Support vector machines for multiple instance learning. In: Proc. Neural Information Processing Systems, pp. 577–584.

Adam, A., Rivlin, E., Shimshoni, I., 2006. Robust fragments-based tracking using the integral histogram. In: Proc. Computer Vision and Pattern Recognition, pp. 798–805.

Lin, R., Ross, D., Lim, J., Yang, M.H., 2004. Adaptive discriminative generative model and its applications. In: Proc. Neural Information Processing Systems, pp. 801–808.

Ross, D., Lim, J., Lin, R.S., Yang, M.H., 2008. Incremental learning for robust visual tracking. Int. J. Comput. Vis., 125–141.

Laptev, I., 2004. Improvements of object detection using boosted histograms. In: Proc. British Machine Vision Conference, pp. 53–60.

Comaniciu, D., Ramesh, V., Meer, P., 2000. Real-time tracking of non-grid objects using mean shift. In: Proc. Computer Vision and Pattern Recognition, pp. 142–149.

Isard, M., Maccormick, J., 2001. Bramble: A Bayesian multiple-blob tracker. In: Proc. International Conference on Computer Vision, pp. 34–41.

Fatih, P., 2005. Integral histogram: A fast way to extract histograms in Cartesian spaces. In: Proc. Computer Vision and Pattern Recognition.

Fergus, R., Perona, P., Zisserman, A., 2005. A sparse object category model for efficient learning and exhaustive recognition. In: Proc. Computer Vision and Pattern Recognition.

Leibe, B., Schindler, K., Cornelis, N., Van Gool, L., 2008. Coupled object detection and tracking from static cameras and moving vehicles. IEEE Trans. Pattern Anal. Machine Intell., 1683–1698.

Kwon, J., Mu Lee, K., 2009. Tracking of a non-rigid object via patch-based dynamic appearance modeling and adaptive basin hopping Monte Carlo sampling. In: Proc. Computer Vision and Pattern Recognition.

Jepson, A.D., Fleet, D.J., El-Maraghi, T.F., 2003. Robust online appearance models for visual tracking. IEEE Trans. Pattern Anal. Machine Intell., 1296–1311.

Wang, J., Chen, X., Gao, W., 2005. Online selecting discriminative tracking features using particle filter. In: Proc. Computer Vision and Pattern Recognition, pp. 1037–1042.

Avidan, S., 2001. Support vector tracking. IEEE Trans. Pattern Anal. Machine Intell., 73–77.

Kalal, Z., Matas, J., Mikolajczyk, K., 2009. Online learning of robust object detectors during unstable tracking. In: Proc. International Conference on Computer Vision Workshop. .

Kalal, Z., Matas, J., Mikolajczyk, K., 2010. P-N learning: Bootstrapping binary classifiers by structural constraints. In: Proc. Computer Vision and Pattern Recognition.