MULTIPLE INSTANCE TRACKING BASED ON HIERARCHICAL MAXIMIZING BAG'S MARGIN BOOSTING

Chunxiao Liu, Guijin Wang, Xinggang Lin, Bobo Zeng

Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China

ABSTRACT

In online tracking, the tracker evolves to reflect variations in object appearance and surroundings. This updating process is formulated as a supervised learning problem, thus a slight inaccuracy of the tracker will degrade the updating. Multiple Instance Learning (MIL) is used to alleviate such a problem by representing training samples in bags of image patches (or called instances). Difficulties are then passed on to the learning method to train a classifier that discovers the most accurate instance. This paper proposes a Maximizing Bag's Margin (MBM) criteria for MIL. Combined with MBM, a hierarchical boosting is proposed for updating, in which bag and instance weights are introduced to guide classifier retraining. Our approach effectively improves the updating's efficiency with less computation cost. Experiments demonstrate the benefits of our method.

Index Terms— online tracking, multiple instance learning, online learning, boosting

1. INTRODUCTION

In many visual tracking scenarios, the object's appearance, background and surrounding illuminations may vary over time. Consequently, a fixed observation model will not be consistent with the ever changing situation, thus may fail. This is called model drift. Online tracking schemes have been proposed to adaptively update their models to reflect the variation of the object and background. Discriminative tracking, which trains a classifier to distinguish object from background, has been successively extended to an online method [1, 2, 3, 4, 5, 6]. These methods evolve their classifiers by retraining with positive and negative samples based on the current tracking result. A common sampling method utilizes the estimated object location as the positive sample, around which a set of negative samples are extracted. Therefore, the effectiveness of updating mostly depends on the current tracking result. A slight inaccuracy during tracking will cause more inaccuracy and gradually lead to tracking error.

In this paper, we focus on the model drift problem and explore an updating methodology based on Multiple Instance representation. Our contributions are three folds: Firstly, we propose a novel Maximizing Bag's Margin (MBM) criteria conjugated with a hierarchical boosting structure to solve the Multiple Instance problem. Secondly, we introduce the concept of weight to both bags and instances to evaluate their importance in the training process. Then the weak classifier search in boosting is based on the instances' weight distribution. This improves robustness of updating to outliers in bags or instances. Thirdly, with MBM criteria, a simple weak classifier evaluation principle is used which enables our method to compute much faster and run at twice the speed of that in [1] to easily achieve real-time performance.

The remainder of the paper is organized as follows: The concept of multiple instance learning and its application in tracking are given in Section 2. Then in Section 3, we introduce the MBM criteria for multiple instance learning optimization, and present our hierarchical boosting scheme based on MBM. Experiments are given to illustrate the efficiency of our method in Section 4. Section 5 concludes the paper.

2. MULTIPLE INSTANCE LEARNING

Multiple instance learning (MIL) is a machine learning method representing training sets by labeled bags that are composed of unlabeled instances. In binary classification, a bag is labeled positive if at least one instance in that bag is positive, while a bag is labeled negative if all the instances in it are negative. The goal of MIL in this paper is to find 'actual' positive instances in positive bags.

O. Maron et al. [7] proposed a general framework for solving MIL problem by maximizing Diverse Density (DD) function, defined as follows: $\max \sum_{i=1}^{m} \log(P(lable_i|bag_i))$ where m is the number of bags. It aims to find a concept point in the feature space that is close to at least one instance from every positive bag and far away from all the instances in the negative bags. From margin perspective, Andrews et al. [8] use SVM for instance discrimination. Chen Yinxi et al. [9] extend this idea by formulating a maximum margin problem in a feature space defined by DD function.

For tracking task, B. Zeisl et al. [2] propose to combine the advantages of semi-supervised learning and MIL for model updating. However the adaptivity of the tracker is also limited due to the constraint of loss function used in semi-supervised learning. Babenko et al. [1] employ a gradient-based MIL approach for model updating. They train an instance classifier under boosting framework, in which the classifier discriminative ability is defined by DD function. However, bags and instances are treated equally in this method which may not be prone to train a "wise" instance classifier that can pick the accurate instance out. In addition, as a cost function, DD increases the computation burden due to a large amount of logarithm and exponential calculations. Margin-based solutions have shown their superiorities in image classification task, nevertheless they cannot be readily adapted into an online fashion for model updating in tracking.

3. HIERARCHICAL MAXIMIZING BAG'S MARGIN BOOSTING

Our work is based on [1] to tackle model drift. From margin perspective, we propose a Maximizing Bag's Margin criteria for MIL, followed by a Hierarchical Boosting to solve the online learning problem. Both bags and instances are assigned weights, which can be naturally updated during boosting to dynamically reflect their importance to the current classifier training.

3.1. Maximizing Bag's Margin Criteria

Considering bag samples $\{(X_1, y_1), (X_2, y_2), \ldots, (X_m, y_m)\}$, where X_i denotes a bag containing n instances $X_i = \{x_{i2}, x_{i2}, \ldots, x_{in}\}$ and y_i is the bag's label, $y_i \in \{-1, 1\}$. We define the bag margin as:

$$\mathrm{margin}(X) = yF(X) \tag{1}$$

where $F(X)$ can be regarded as a bag level classifier. The higher response of $F(\cdot)$ denotes a larger margin from the bag sample X to the separating plane.

We propose the Maximizing Bags' Margin (MBM) as the optimization criteria for solving the MIL problem to minimizing the expectation of an exponential term:

$$\min E\{e^{-yF(X)}\}. \tag{2}$$

Note that unlike traditional boosting optimization function [10], the classifier $F(X)$ here is acting on bag representation directly rather than on instances. Therefore it is impractical for implementation. Moreover, our ultimate goal for MIL is to train a classifier for instance level determination, denoted by $f(x)$. To bridge the gap between these two kinds of classifiers, we adopt the Noisy-OR (NOR) model [7] which expresses the contribution of instances to a bag of being positive as follows:

$$p(X_i) = 1 - \prod_{j=1}^{n}(1 - p(x_{ij})). \tag{3}$$

We model the bag and instance probabilities using sigmoid function as

$$p(X) = \frac{1}{1 + e^{-F(X)}}, p(x) = \frac{1}{1 + e^{-f(x)}}. \tag{4}$$

Consequently relationship between the bag level classifier and instance level classifier is built. Thus far, the MBM criteria can be indirectly embedded into the instance level classifier updating.

3.2. Hierarchical MBM Boosting

To this end, model updating in tracking can be viewed as retraining an instance level classifier using bag form samples based on current tracking result. To evaluate the importance of bag and instance for training, we assign weights to each bag as well as each instance contained. To represent instance weight, we simply multiply the bag weight with the instance probability [11]:

$$w_{ij} = w_i \cdot p_{ij} \tag{5}$$

where w_i is the bag weight for X_i, w_{ij} and p_{ij} denote the weight and probability of the j-th instance in bag X_i respectively.

Fig .1 shows the proposed hierarchical boosting structure for updating, aiming at combining MBM in bag level and model updating in instance level. The details can be described as follows: assume that the bag level classifier $F(X)$ has the following form $F(X) = \sum_{t=1}^{T} c_t(X)$, where $c_t(\cdot)$ is the weak classifier for bag. And in instance level $f(x_i) = \sum_{t=1}^{T} h_t(x_i)$ where $h_t(\cdot)$ denotes the weak classifier acting on instance. Both $c_t(\cdot)$ and $h_t(\cdot)$ have real value output. In instance layer boosting, performance of instance weak classifier is simply defined by its weighted response to all the instances:

$$\sum_{i=1}^{M} \sum_{j=1}^{NI_i} w_{ij} h_t(x_{ij})$$

where M denotes the bag number and NI_i denotes the instance number in the i-th bag. In each round, the weak classifier with best performance on current instance weight distribution is selected. To solve the critical reweighting problem for bag and instance, bag-level boosting is executed in pursuit of MBM criteria optimization. Although there is no concrete expression for bag-level classifier, bag weights in the m-th round of boosting can be calculated based on the conclusion in [10] and Eq. 4 as follows:

$$w_i^m = y_i((1 - p(y_i|X_i))/p(y_i|X_i))^{y_i}. \tag{6}$$

Combine Eq. 5 and Eq. 4, the instance weight updating can be obtained. From bag weight updating equation we see that the bag weight will decrease if it has a large classifier response. Thus in the next round, the classifier selection will emphasize the misclassified bags. This is in accordance with the effect

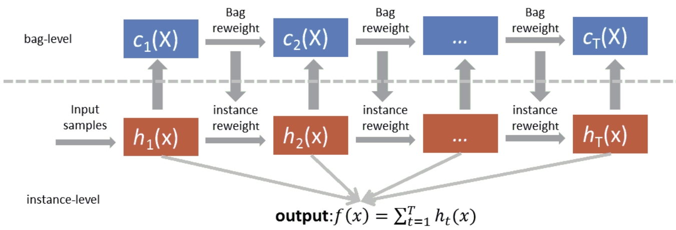

Fig. 1. Hierarchical MBM Boosting. In instance level, weak classifiers are selected successively according to current instance weight distribution. In bag level, weak classifiers are not actually selected from an existing feature pool but derived from instance level classifiers. Then bag and instance weights are updated based on MBM and passed on to instance level for its following selection. The output is the strong classifier in instance level as the result of model updating.

of traditional adaboost reweighting scheme. In instance level boosting, as the learning proceeds, instance with high scores will dominate the subsequent learning. It will help prune the classifier to the most promising instance and enhance the classifier's discriminative ability. The online updating algorithm is summarized in the following table.

Algorithm: HMBM Boosting for Updating

Input : Object bag representation $\{X_i, y_i\}$, where

$X_i = \{x_{i1}, x_{i2}, \ldots, x_{in}\}$, $y_i \in \{-1, 1\}$, $i \in [1, M]$

Update all weak classifier $h_t(\cdot)$

Initialize strong classifier $f = \{\}$, bag weight w_i^0, and instance weight w_{ij}^0, $j \in [1, n]$

Repeat for $m = 1, 2, \ldots, T$

(a) Select a weak classifier with the best performance currently

$$h_m = \arg\max_{h_t} \sum_{i=1}^{M} \sum_{j=1}^{NI_i} w_{ij}^{m-1} h_t(x_{ij})$$

NI_i denotes the number of instance in bag X_i

(b) Update bag weight and instance weight

$w_i^m = y_i (\frac{1-p(y_i|X_i)}{p(y_i|X_i)})^{y_i}$

$w_{ij}^m = w_i^m \cdot p_{ij}$

(c) Update strong classifier

$f = f + h_m$

Output : Strong classifier in instance level $f = \sum_{t=1}^{T} h_t(\cdot)$

---

4. EXPERIMENT

We implement our algorithm based on MILBoost [1] using the same weak classifiers and haar-like features. All the parameters are set as reported in [1] and kept constant. A simple tracker is used in our experiment: suppose that we've detected the object in frame t and updated the classifiers based on the t-frame detection. In the $t + 1$ frame, we evaluate the classifier in a target search region centered at the previous detection. Then shift the tracking window to the image patch that receives maximum response. For bag sampling strategy, each positive bag contains a set of image patches that are close to the current tracking window, at a distance from which negative bags are sampled. Each negative bag contains only one image patch.

We evaluated our algorithm on several challenging video sequences that are publicly available [12, 13]. Ground truths are provided for every 5 frames, and they all use fixed sizes for objects. These video clips exhibit object appearance, illumination change, scale variation, short time occlusion and background clutter.

We compare our HMBMBoost method with MILBoost. Randomness is involved in the boosting initialization to generate a feature pool that would influence the tracking performance. Thus we run the authors' code on these test videos to make sure that we are initialized with the same feature pool. This could isolate the updating models of these two methods for comparison.

4.1. Tracking Performance

Fig. 2 shows some captures of the tracking process for video "tiger2", in which object's appearance and surrounding situation change. Our tracking system achieves a robust performance. In the bottom of each image, we show top 5 highest weight instances in positive bag and negative bags for model updating of current frame. The selected instances in positive bag are very close to the target location, demonstrating the effectiveness of the reweighting operation classifier training by using more promising instances for . MILBoost tracker lost its target after frame 339 while our tracker kept continuous tracking. In Fig. 3 we draw the classifiers' response of frame 339, within a 8 × 8 target searching region centered at the previous tracking location. Our classifier response shows a steep and unimodal distribution illustrating a good discrimination. Due to some randomness in the algorithms, we run each testing video 5 times. Table. 1 summarizes the testing results in the form of (A,B), where 'A' denotes the average location error with standard variation, and 'B' denotes the average number of correctly tracked frames. Location errors are calculated using only correctly tracked frames. In most cases, we produce better performances with more precise and long-lasting tracking. MILBoost presents better results in "girl" and "seq_dk" sequences in which objects show dramatic changes by out of plane rotation. In this situation, reweighting scheme in our method will lose its effect and can not promise a good classifier updating.

4.2. Speed

Although a heuristic weak classifier selection in the form of weighted sum is employed in instance-level boosting, much computation cost is saved while maintaining comparable performance. Let N_{in} and N_{wk} denote the number of instances

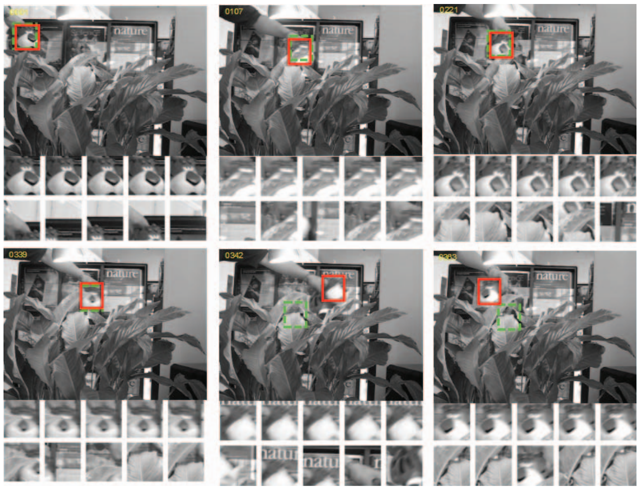

Fig. 2. Tracking results of "Tiger2". Results of MILBoost and HMBMBoost are shown in green dash and red solid rectangle respectively. In the bottom of each image 5 instances with highest weight in positive bag and negative bags are shown in the first row and the second row respectively.

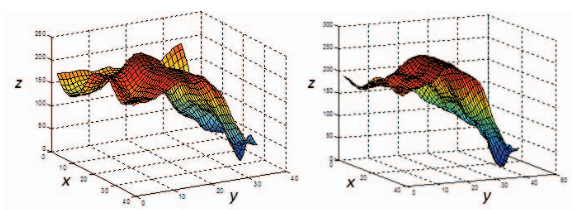

Fig. 3. Instance classifier response (denoted in z axis) in target search region (in $x-y$ plane) of frame 339. (Left): MILBoost Tracker; (Right):HMBMBoost Tracker.

and weak classifiers respectively. In our updating scheme, only $(N_{in} \cdot N_{wk})$ times adding operations are needed for weak classifier selection. As for MILBoost, exponential and logarithmic operations with the same calculation amount are involved. This also illustrates, our updating scheme can scale up well with a larger feature poor. On 2.53G Intel Core 2 processors with 4G memory, our method can run at about 20fps which is two times faster than that of MILBoost.

5. CONCLUSIONS

In this paper, we propose a novel Maximizing Bag's Margin criteria to solve the MIL problem in online updating of tracking. Experiments demonstrate our approach's effectiveness in handling object's appearance and surrounding variations. In addition our method is computationally efficient and realtime capable. However, long time occlusion and rapid drastic changes of object appearance can not be well handled. We will tackle these problems in the future.

Table 1. Summarization of testing results, in the form of (A,B). A denotes the average location error with standard variation. B denotes the average number of correctly tracked frames.

Video	MILBoost	HMBMBoost
David Indoor	(26±4.34 , 462)	(**21±6.39** , 462)
Sylvester	(13±1.39 , 1345)	(**12±2.53** , 1345)
Tiger 2	(11±1.54 , 202)	(**8±1.81** , **256**)
Girl	(**34±2.21** , **320**)	(36±3.79 , 314)
Coke11	(22±2.05 , 292)	(**19±0.32** , 292)
seq_dhb	(7±4.22 , 51)	(**5±1.45** , 51)
seq_dk	(**5±0.71** , 51)	(6±0.55 , 51)

6. REFERENCES

[1] B. Babenko, M.H. Yang, and S. Belongie, "Visual tracking with online multiple instance learning," in *Proc. CVPR*, 2009.

[2] Bernhard Zeisl, Christian Leistner, Amir Saffari, and Horst Bischof, "On-line semi-supervised multiple-instance boosting," in *Proc. CVPR*, 2010.

[3] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Trans.on PAMI*, vol. 27, no. 10, pp. 1631–1643, 2005.

[4] H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via online boosting," in *Proc. BMVC*, 2006, pp. 47–56.

[5] F Tang, Q. Zhao S Brennan, and H. Tao, "Co-tracking using semi-supervised support vector machines," in *Proc. ICCV*, 2007.

[6] T. Parag, F. Porikli, and A. Elgammal, "Boosting adaptive linear weak classifiers for online learning and tracking," in *Proc. CVPR*, 2008.

[7] Maron O and Lozano-Prez T, "A framework for multiple-instance learning," in *Proc. NIPS*, 1998, pp. 570–576.

[8] S. Andrews, I. Tsochantaridis, and T. Hofmann, "Support vector machines for multiple-instance learing," in *Proc. NIPS*, 2003, pp. 561–568.

[9] Y. Chen and Z. Wang, "Image categorization by learning and reasoning with regions," *Journal of Machine Learning Research 5*, pp. 913–939, 2004.

[10] T.Hastie J.Friedman and R.Tibshirani, "Additive logistic regression:a statistical view of boosting," *The Annals of Statistics*, vol. 38, no. 2, pp. 337–407, 2000.

[11] P.Viola, J. C. Platt, and C. Zhang, "Multiple instance boosting for object detection," in *Proc. NIPS*, 2005, pp. 1417–1426.

[12] http://vision.ucsd.edu/ bbabenko/project_miltrack.shtml

[13] http://vision.stanford.edu/ birch/headtracker/seq/