International Journal of Engineering Research in Computer Science and Engineering (IJERCSE) Vol 2, Issue 1, January 2015

Enhanced Active Feature Selection method for Object Tracking

[1]Sandeep G.S [2]Sunitha M R [3]Niranjan Murthy C

[1] Asst. Professor, Dept of CS&E,G M Institute of Technology, Davangere, Karnataka sandeepgsgagan@gmail.com,

[2] Associate Professor, Dept of CS&EAdichunchanagiri Institute of Technology, Chikmagalur, sunithamr2310@gmail.com

[3] Asst. Professor,Dept of CS&E,G M Institute of Technology, Davangere, Karnataka,nirucs190@gmail.com.

Abstract: **The key idea of Adaptive tracking is how to train an online discriminative classifier which can well separate object from its local background. The classifier is incrementally updated using positive and negative samples extracted from the current frame around the detected object location. If the detection is less accurate thereby leading to visual drift. Recently, the multiple instances learning (MIL) based tracker has been proposed to solve these problems to some degree. It puts samples into the positive and negative bags, and then selects some features with an online boosting method via maximizing the bag likelihood function. Finally, the selected features are combined for classification. In MIL tracker, the features are selected by a likelihood function, which can be less informative to tell the target from complex background. Motivated by the active learning method, in this paper we propose an enhanced active feature selection approach which is able to select more informative features than MIL tracker by using the Fisher information criterion to measure the uncertainty of classification model. More specifically, we propose an online boosting feature selection approach via optimizing the Fisher information criterion, which can yield more robust and efficient real-time object tracking performance.**

Experimental evaluations on challenging sequences demonstrate the efficiency, accuracy and robustness of the proposed tracker in comparison with state-of-the-arts.

Index Terms **Visual tracking, Multiple instance learning, fisher information, active learning.**

I. INTRODUCTION

the classifier, the selection of positive and negative samples affects much the performance of the tracker. Most trackers only choose one positive sample, i.e., the tracking result in the current frame. If the tracked target location is not accurate, the classifier will be updated based on a less effective positive sample, thereby leading to visual drift over time. To alleviate the drifting problem, multiple samples near the tracked target location can be used to train the classifier. A multiple instance learning (MIL) approach [2] was proposed to solve the ambiguity problem in tracking. The samples are put into bags and only the labels of the bags are provided. The bag is positive if one or more instances in it are positive while the bag is negative when all of the instances in it are negative. The samples near the tracking location are put into the positive bag while the samples far from the tracking location are put into the negative bag. To handle the appearance variations over time, an online MIL boosting algorithm is proposed to greedily select the discriminative features from a feature pool by maximizing the bag likelihood function. The strong classifier is then used to separate object from background in the next frame.

The MIL tracker [2] has the following shortcomings. First, the selected features may be less informative. In order to make the classifier discriminative enough, a relatively large number of features are selected from the feature pool. This enlarges the computational burden. Second, the more features are selected, the higher the probability that less discriminative features are included. These less discriminative features can degrade the performance of the classifier, and cause drift over time. To address the above problems, inspired by the active learning method [3] we propose a novel feature selection scheme to select the more informative features for visual tracking, namely, the active feature selection (AFS) based tracker. An online feature selection scheme is proposed by optimizing a bag Fisher information function instead of the bag likelihood function. Thus, the selected features are much more informative than those selected by the bag likelihood function in MIL tracker [2]. Consequently, we can use less features to design a classifier which is more efficient and robust than the classifier induced by the MIL tracker. Our experimental evaluations on challenging video clips validate the superior performance of AFS tracker to state-of-the-art trackers in terms of efficiency, accuracy and robustness.

II. Related Work

The recent algorithms can be mainly categorized into two classes according to how they deal with the appearance variations of target object and the background: the generative methods [4-12] and the discriminative methods [2][13-21]. The generative methods learn an appearance model for the target object by minimizing the difference between the each region and the reference object model. Black et al. [4] represented the object by learning a subspace model offline. To handle appearance variations of the object over time, some online appearance update models have been proposed. Jepson et al. [5] proposed a Gaussian mixture model which is updated by an online expectation maximization (EM) algorithm. Ho et al. [6] and Ross et al. [7] used the incremental subspace update schemes to adapt the appearance variation. Adam et al. [8] proposed a fragment-based appearance model to deal with the pose variation and partial occlusion.

Recently, sparse representation methods have been proposed to handle the partial occlusion in visual tracking [9]. Kwon et al. [10] decomposed the observation model into multiple basic observation models which cover different kinds of features and motions to handle pose variations, illuminations and scale changes. Sun et al. [11] proposed an object appearance model which combines the local scale-invariant feature and the global incremental principle component analysis (PCA). The discriminative methods treat tracking as a binary classification problem by training a discriminative classifier to separate object from background. Avidan [13] trained an offline support vector machine (SVM) and combined it into an optic-flow based tracker. To adapt the appearance changes of the object and background over time, Avidan [14] proposed an online boosting method to train the classifier: some weak classifiers are updated in an online manner and then ensemble into a strong classifier. Collins et al. [15] proposed an online feature selection scheme to evaluate the multiple features and integrated this scheme into a mean-shift tracking system [12] to select the most discriminative features. In [16], the relationship between the object and the structured environments is exploited to improve the performance of tracking. Grabner et al. [17] developed an online boosting feature selection technique which demonstrates good performance to adaptively handle appearance changes.

To better handle visual drift, Grabner et al. [18] proposed an online semi-supervised tracker which only labels the samples in the first frame while leaving the samples in the sequent frames unlabeled. Babenko et al. [2] proposed to use an online MIL approach to handling the ambiguity in tracking location to reduce visual drift. Kalal et al. [19] proposed a semi-supervised learning approach to select the positive and negative samples via an online classifier with structural constraints. Recently, an efficient tracking algorithm [21] based on compressive sensing theory [22] was proposed, which demonstrates that the low dimensional features randomly extracted from the high dimensional multi-scale image feature space can preserve the discriminative capability, thereby facilitating object tracking.

2. **Tracking with Active Feature Selection**

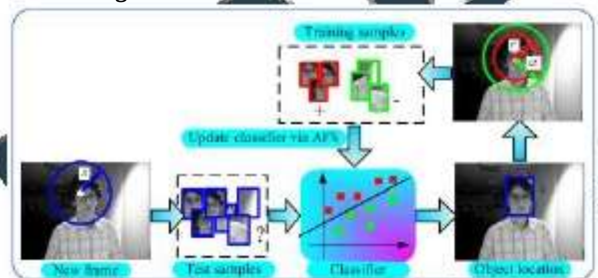

Figure1: Illustration of how our tracking system works

A. System overview

Fig. 1 illustrates the basic flow of our tracking system. There are two important components in our tracking system. One is how to detect the object location in the new frame, and the other is how to update the classifier. We represent the object location in the *t*-th frame as location are cropped as

$D^s = \{x \mid | l(x) - l^*_{t-1} | < s\}$,

Where *s* is a search radius and *x* denotes the image patch. Then, we compute the classifier response $H(x)$ for all $x \in D^s$, where the classifier $H(x) = \sum_k h_k(x)$ is a linear combination of some weak classifiers $h_k(x)$. Finally, we update the object location using a greedy strategy

$$l^*_t = l(\arg\max_{x \in D^s} H(x)) \qquad (1)$$

After the object location is updated, a set of samples $D^s = \{x \mid | l(x) - l^*_{t-1} | < r\}$, where *r* is a scalar radius, are cropped and put into a positive bag. For the negative samples, we take a small random set of samples from set $D^{r\text{-}\beta} = \{x \mid r < | l(x) - l^*_t | < \beta\}$, where β is a scalar radius, because $D^{r\text{-}\beta}$ contains a large number of samples. If the background between two consecutive frames do not changes much, the negative patches which are not from the boundary area around the target may be beneficial for classification because they will much correlate with each other.

B. MIL tracker

The MIL method was introduced by Dietterich et al. [23] to deal with the drug activity prediction. Suppose that we have a set of *N* bags $\{X1,\ldots..Xn\}$, where each bag $Xi = \{X_{i1}\ldots\ldots X_{inj}\}$ has n_i instances. Let $y_i \in \{0,1\}$ be the label of bag Xi and $y_{ij} \in \{0,1\}$ the label of instance x_{ij}. The MIL defines that if bag *Xi* is positive, then at least one of the instance labels in it is positive. If the bag label is zero, then all of the corresponding instance labels are zero. The MIL tracker seeks for the discriminative classifier *H*(*x*), which can return the conditional Probability $p(y=1|x)$. Since the discriminative classifier is an instance classifier that is related to the conditional probabilities of the instances, the Noisy-OR model is used to exploit the conditional probabilities of the instances to estimate the bag probability.

$$p(y_i = 1 | X_i) = 1 - \prod_j (1 - p(y_{ij} = 1 | x_{ij})) \quad (2)$$

Where the instance probability $p(y_{ij} = 1 | x_{ij})$ *is modeled as* $p(y_{ij} = 1 | x_{ij}) = \sigma(H(x_{ij}))$ *(3)*

Where σ(z) is the sigmoid function, and the classifier $H(x)$ is learned by maximizing the following bag log likelihood loss function

$$\mathcal{L}(H) = \sum_i (y_i \log(p(y_i = 1 | X_i)) + (1 - y_i) \log(1 - p(y_i = 1 | X_i))) \quad (4)$$

To handle the appearance changes over time, an online MIL boosting approach is proposed to update the classifier $H(x)$. First, a weak classifier pool is maintained, and then a small number of weak classifiers are greedily selected from the pool by maximizing the log likelihood of the bag

$h_k = \text{argmax}_{h \in \varphi} L(H_{k-1} + h)$ (5)

where H_{k-1} is a strong classifier and φ is a weak classifier.

C. Principle of AFS

The selected features can be less informative than those selected by optimizing the Fisher information function in our method to be introduced below. Therefore, to ensure the enough discriminative information, in the MIL tracker [2] a relatively large number of features (K=50) are selected from a feature pool with a relatively large size (M=250), while in our AFS tracker only K=15 features are selected from a pool with M=50 features. Moreover, if too many features are selected, the discrimination between the object and background features can be reduced.

We take the classifier as the following form

$H(x) \square \alpha^T h(x)$ (6)

Where α is weight vector and h is a weak classifier vector.

Each element in h is a decision stump function that returns the binary labels (i.e., +1 or □1). In order to devise the classifier $H(x)$, we need to estimate its corresponding parameters α. The Cramer-Rao inequality [25] shows that for any unbiased estimator t_n of α based on n independent and identically distributed samples from the probability.

$I(\alpha)$ is the Fisher information matrix defined as

$$I(\boldsymbol{\alpha}) = -\int p(y | \boldsymbol{\alpha}) \frac{\partial^2}{\partial \boldsymbol{\alpha}^2} \log p(y | \boldsymbol{\alpha}) dy \quad (7)$$

In [26], for each query in active learning, an unlabeled sample that can decrease the Fisher information most is selected. To measure the uncertainty of the classification model in our AFS tracker, we use the Fisher information matrix based on the samples from the bag probability.

$$I(\boldsymbol{\alpha}) = \sum_i \left[y_i p(y_i | X_i, \boldsymbol{\alpha}) \frac{\partial^2}{\partial \boldsymbol{\alpha}^2} \log p(y_i | X_i, \boldsymbol{\alpha}) + (1 - y_i) p(y_i | X_i, \boldsymbol{\alpha}) \frac{\partial^2}{\partial \boldsymbol{\alpha}^2} \log p(y_i | X_i, \boldsymbol{\alpha}) \right] + \delta I_m$$

The inverse Fisher information matrix $I(\alpha)^{-1}$ is the lower bound of the covariance matrix of the estimated α[25]. As a particular case, $\det(I(\alpha)^{-1})$ is the lower bound of the product of the variances for the elements in α. since it is difficult to compute $\det(I(\alpha))$ in our objective function we relax it to *minimizing* the trace of matrix *I(α)* (denote by tr(I(α)) because the upper bound of det(I(α)).

For the positive bag, as learning proceeds and the bag probability approaches to the target, Thus, the component of the positive bag in tr(I (α)) can be simplified. In order to minimize this function, we need to maximize two terms $p(y_i = 1 | x_i, \alpha)$ and $p(y_{ij} = 1 | x_{ij}, \alpha | x_{ij}, \alpha)$. The first term is to maximize the conditional probability of the positive bag. The second term reaches its maximum value at $p(y_{ij} = 1 | x_{ij}, \alpha) = 0.5$.

D. Online AFS boosting

Weak classifiers are enhanced for statistical view and are selected sequentially to optimize a specific objective function F as

$(h_k, \alpha_k) = \text{argmin}_{h \in \varphi} F(H_{k-1} + \alpha h)$ (9)

Where H_{k-1} is a strong classifier with first k-1 weak classifier and ϕ is the set of all possible weak classifier. For online learning, we always maintain a pool of M candidate weak classifiers. When updating the strong classifier, we first incrementally update the weak classifiers in the pool with the newly cropped samples, and then select sequentially $K \square M$ the most discriminative weak classifiers from the pool by minimizing the Fisher information criterion

$$(h_k, \alpha_k) = \arg\min_{h \in \{h_1, \ldots, h_M\}, \alpha \in \square} \mathcal{F}(H_{k-1} + \alpha h) \quad (10)$$

Algorithm 1: Online AFS Boosting

Input: Dataset $\{X_i, y_i\}_{i=1}^N$, where $X_i = \{x_{i1}, x_{i2}, \ldots\}$ is the i-th bag and $y_i \in \{0,1\}$.

1. Update all the M weak classifiers in the pool with data $\{x_{ij}, y_i\}$.
2. Initialize $H_0(x_{ij}) = 0$ for all i, j
3. **For** k=1 to K **do**
4. **for** m=1 to M **do**
5. $\mathcal{F}_m = \mathcal{F}(H_{k-1} + h_m)$
6. **end for**
7. $m^* = \arg\min_m(\mathcal{F}_m)$
8. $h_k \leftarrow h_{m^*}$
9. $H_k \leftarrow H_{k-1} + h_k$
10. **End for**

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

Algorithm shows the pseudo-code of online AFS Boosting, which is the key part of the tracking algorithm.

E. Advantages over the MIL tracker

Our Fisher information criterion (13) can select the features which are much more informative than those selected from the log likelihood criterion (5) in the MIL tracker [2], because our criterion maximizes classifiers which are more discriminative than those used in the MIL tracker. In our experiments, we select K=15 weak classifiers from a pool with M=50 candidate weak classifiers, which are much less than the MIL tracker where K=50 and M=250. Although our objective function (11) seems more complex than that used in MIL tracker (i.e., (4)), their computational complexities are comparative because only addition and multiplication are needed to compute bag and instance probabilities. Moreover, the MIL tracker needs to update more classifiers (*M*=250) than ours (*M*=50), and select more weak classifier (*K*=50) than our method (*K*=15). Thus, overall our tracker is more efficient than MIL tracker. In addition, because our selected weak classifiers are more informative than those selected by the MIL tracker, our appearance model (i.e., the strong classifier) is able to better handle visual drift. The uncertainty of the selected features. Thus, we only need to actively select a small number of weak

F. Implementation details

Haar-like image features as used by the MIL tracker [2] which can be efficiently computed using the integral image technique [24]. Each feature f_i is a Haar-like image feature computed by the sum of weighted pixels in 2 ~ 4 randomly selected rectangles. Each weak classifier h_i returns the log odds ratio

$$h_i = \log\left[\frac{p(y=1\mid f_i(x))}{p(y=0\mid f_i(x))}\right] = \log\left[\frac{p(f_i(x)\mid y=1)}{p(f_i(x)\mid y=0)}\right] \quad (11)$$

Where we assume uniform prior p(y=1) = p(y=0). The parameters μ_1, σ_1, t € {0,1} can be incrementally updated based on maximal likelihood estimation.

3. Experimental Results

As the proposed AFS tracker is developed to address several issues of MIL based tracking method we compare it with the MIL tracker [2] on 12 challenging video sequences. The other compared trackers are: fragment tracker (Frag) [8], online AdaBoost tracker (OAB) [17], Semi-supervised boosting tracker (SemiB) [18], incremental visual tracker (IVT) [7], L1 tracker [9], and visual tracking decomposition (VTD) method [10]. The default setting for the MIL tracker is to select *K*=50 weak classifiers from a pool with *M*=250 candidate weak classifiers. We also test the MIL tracker with setting *K*=15 and *M*=50 (we call it MIL_{15}).

Since all the competing trackers (except for [8]) involve randomness, we repeat each experiment 10 times and report the average results. Our tracker is implemented in MATLAB and runs at 15 frames per second on a Pentium Dual-Core 2.10 GHz CPU with 1.95 GB RAM.

TABLE I lists the speed of all trackers in terms of average frames per second (FPS). Note that the source code of the MIL tracker is written in C++ which runs at 10 FPS, while the MIL15 tracker runs at 25 FPS.

TABLE I: Average frames per second (FPS) of AFS and other state-of-the-art trackers

Tracker	Frag	OAB	MIL	MIL15	SemiB	IVT	L1	VTD	AFS
Average FPS	3	8	10	25	6	11	0.1	0.01	15/31

A. Experimental setup

We set the radius *r*=4 for cropping the samples in the positive bag which generates 45 samples. The out radius for the set $D^{r,\beta}$ that generates negative samples is set to β = 35. Then, we randomly select 45 negative samples from $D^{r,\beta}$ to construct the negative bag. The radius for searching the new object location in the next frame is set to *s*=25 and about 2000 samples are drawn, which is the same as that in the MIL tracker [2]. We tested different values of parameter *s* and found the tracking results are stable when we set 20< *s*<30. Hence in all our experiments, we set *s*=25. Therefore, this procedure is time-consuming if too many weak classifiers are used to design the strong classifier. Our tracker uses *K*=15 weak classifiers and thus is much more efficient than the MIL tracker [2] which sets *K*=50. Moreover, in AFS the number of candidate weak classifiers in the pool is set to *M*=50, which is also less than that of the MIL tracker (*M*=250). The learning parameter is set to □ =0.85.

Figure 2: sampled tracking results on the *David indoor* sequence.

*B. Qualitative evaluation:(*Scale and pose changes) Our tracker only estimates the translational motion, it can also handle scale and orientation changes because of the Haar-like features. In the *David indoor* sequence, the target has big scale and poses changes. Note that the IVT, MIL, VTD and our AFS trackers perform well on this sequence while the Frag, OAB, SemiB, L1, and MIL15 have severe drifts. The Haar-like features make MIL and AFS trackers able to handle the scale and pose changes well. So our AFS tracker yields much more accurate results.

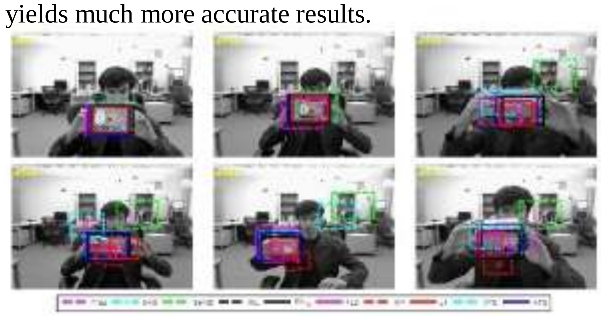

Figure3: sampled tracking results on the *Twinings* sequence.

Background clutter and pose variation: We use four sequence *Tiger-2*, to demonstrate the superior performance of our tracker in handling background clutter and pose variation. In the *Tiger 2* sequence, there are also partial occlusion, rotation, which makes object tracking more difficult.

Occlusion and motion blur: AFS tracker targets undergo occlusion and motion blur. In this sequence, there is pose variation besides partial occlusion. Although the Frag tracker is specially designed to handle partial occlusion by a part-based model, it cannot perform well on this sequence because of the large scale appearance changes due to the severe pose variation and occlusion. The OAB and SemiB trackers drift to the background when the heavy occlusion occurs. The OAB and SemiB trackers are unable to re-detect the target. Although the IVT and L1 methods are able to track the object throughout the sequence, their results are inaccurate and both the two trackers are snapped to cap area. The reason is that they are generative models which do not take into account the useful information from the background.

Both AFS and MIL trackers achieve good results because of the following two reasons. First, the localized Haar-like features are robust to partial occlusion [2]. Second, both trackers use an online update criterion which takes into account the appearance changes of the target and the background.

C. Quantitative evaluation:

The two commonly used criteria to quantitatively assess the performance of the trackers. The tracking success rate and the center location error using the manually labeled ground truth. The tracking results are measured in terms of center location error, which is defined as the Euclidian distance between the center locations of the tracked target and the ground truth. Overall, our AFS tracker performs favorably against the other state-of-the-art trackers.

4. Conclusion

In this paper, we proposed a robust tracker based on an online discriminative appearance model. In order to design a robust appearance model, we developed an online active feature selection (AFS) approach via minimizing a Fishier information criterion. We showed that the features selected by our proposed online AFS boosting algorithm are much more informative and discriminative than those selected by online MIL boosting algorithm which maximizes a likelihood loss function. The AFS appearance model can well handle large appearance changes. Numerous experimental results and evaluations on challenging video sequences demonstrated that our AFS tracker outperforms other state-of-the-art algorithms in terms of efficiency, accuracy and robustness.

References

[1]. A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Comput. Sur.*, vol. 38, no. 4, Dec. 2006.

[2]. B. Babenko, M.-H. Yang, and S. Belongie, "Robust Object Tracking with Online Multiple Instance Learning," *IEEE Trans. Pattern Anal. Mach Intell.*, vol. 33, no. 8, pp. 1619–1632, 2011.

[3]. Burr Settles, "Active learning literature survey," *Technical Report 1648*, University of Wisconsin Madison, 2009.

[4]. M. Black, A. Jepson, and Eigentracking, "Robust matching and tracking of articulated objects using a view-based representation," in *Proc. Eur. Conf Comput. Vision.*, 1996, pp. 329–342.

[5]. A. Jepson, D. Fleet, and T. Maraghi, "Robust online appearance models for visual tracking," *IEEE Trans. Pattern. Anal. Mach. Intell.*, vol. 25, no. 10, pp. 1296–1311, 2003.

[6]. J. Ho, K. Lee, M. Yang, and D. Kriegman, "Visual tracking using learned linear subspace," in *Proc.IEEE Conf. Comput. Vis. Pattern Recog.*, 2004, pp. 782–789.

[7]. D. Ross, J. Lim, R. Lin, and M. Yang, "Incremental learning for robust visual tracking," *Int. J.Comput. Vis.*, vol. 77, no. 1, pp. 125–141, 2008.

[8]. A. Adam, E. Rivlin and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 798–805.

[9]. X. Mei and H. Ling, "Robust visual tracking using l1 minimization," in *Proc. IEEE Conf. Comput. Vision.*, 2009, pp. 1436–1443.

[10]. J. Kwon and K. Lee, "Visual tracking decomposition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 1269–1276.

[11]. L. Sun and G. Liu, "Visual object tracking based on combination of local description and global representation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 4, pp. 408–420, 2011.

[12]. D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Trans. Pattern Anal.Mach. Intell.*, vol. 25, no. 5, pp. 564–575, 2003.

[13]. S. Avidan, "Support vector tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 8, pp.1064–1072, 2004.

[14]. S. Avidan, "Ensemble tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no.2, pp. 261–271,2007.

[15]. R. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 10, pp. 1631–1643, 2005.

[16]. J. Zhu, Y. Lao, and Y. Zheng, "Object tracking in structured environments for video surveillance applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 2, pp. 223–235, 2010.

[17]. H. Grabner, M. Grabner, and H. Bischof, "Real-time tracking via online boosting," in *British Machine Vision Conference*, 2006, pp. 47–56.

[18]. H. Grabner, C. Leistner, and H. Bischof, "Semi-supervised on-line boosting for robust tracking," in *Proc. Eur. Conf. Comput. Vision.*, 2008, pp. 234–247.
[19]. Z. Kalal, J. Matas, and K. Mikolajczyk, "P-N learning: bootstrapping binary classifiers by structural constraints," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 49–56.
[20]. K. Zhang, L. Zhang, and M.-H. Yang, "Real-time compressive tracking," In *Proc. Eur. Conf. Comput.Vis.*, 2012.
[21]. E. Candes and T. Tao, "Near optimal signal recovery from random projections and universal encoding strategies.," IEEE Trans. Inform. Theory, vol. 52, no. 12, pp. 5406–5425, 2006.
[22]. T. Dietterich, R. Lathrop, and T. Perez, "Solving the multiple instance problem with axis-parallel rectangles," *Artificial Intelligence*, pp. 31–71, 1997.
[23]. P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proc.IEEE Conf. Comput. Vis. Pattern Recog.*, 2001, pp. 511–518.
[24]. M. Cover and J. Thomas, *elements of information theory*, New York: Wiley, 1991.
[25]. T. Zhang and F. Oles, "A probability analysis on the value of unlabeled data for classification problems," in *Pro. Int. Conf. on Machine Learning.*, 2000, pp. 1191–1198.
[26]. B. Settles, M. Craven, and S. Ray, "Multiple-instance active learning," In *Proc. Adv. Neural Inf.Process. Syst.*, 2008, pp. 1289–1296.
[27]. D. Zhang, F. Wang, Z. Shi, and C. Zhang, "Interaction localized content based image retrieval with multiple-instance active learning," *Pattern Recognit.*, vol. 43, pp. 478–484, 2010.