TransientBoost: On-line Boosting with Transient Data *

Sabine Sternig Martin Godec Peter M. Roth
Horst Bischof
Institute for Computer Graphics and Vision
Graz University of Technology
{sternig,godec,pmroth,bischof}@icg.tugraz.at

Abstract

For on-line learning algorithms, which are applied in many vision tasks such as detection or tracking, robust integration of unlabeled samples is a crucial point. Various strategies such as self-training, semi-supervised learning and multiple-instance learning have been proposed. However, these methods are either too adaptive, which causes drifting, or biased by a prior, which hinders incorporation of new (orthogonal) information. Therefore, we propose a new on-line learning algorithm (TransientBoost), which is highly adaptive but still robust. This is realized by using an internal multi-class representation and modeling reliable and unreliable data in separate classes. Unreliable data is considered transient, hence we use highly adaptive learning parameters to adapt to fast changes in the scene while errors fade out fast. In contrast, the reliable data is preserved completely and not harmed by wrong updates. We demonstrate our algorithm on two different tasks, i.e.*, object detection and object tracking showing that we can handle typical problems considerable better than existing approaches. To demonstrate the stability and the robustness, we show long-term experiments for both tasks.*

1. Introduction

Object detection or single target tracking are important tasks in computer vision, which are usually formulated as a binary classification problem. Hence, a discriminative classifier has to distinguish the object of interest from the background. For most applications the object of interest and/or the background are changing over time, which requires complex classifiers to cover all the variability in the data. To avoid complex classifiers on-line classifiers can be applied, which are capable to adapt the model to both, changing background and changing appearance of the object of interest. This, however, requires that new unlabeled or unreliable labeled data has to be used for updating the current classifier, where several learning strategies have been introduced.

In self-training (*e.g.*, [13]) the current classifier evaluates a sample and predicts a label, which is used to update itself. This strategy is applied in most trackers based on on-line learning (*e.g.*, [9]). Even though these samples often characterize correct data that may improve the classifier, this approach is highly prone to drifting, *i.e.*, slightly inaccurately labeled samples cause that over time the focus gets lost and that the classifier starts to learn something totally wrong. In contrast, by using an oracle (*e.g.*, [2, 15]) drifting can be avoided. An oracle can be considered a classifier, which has a high accuracy while the recall may be low. As a drawback, however, oracles are often too conservative neglecting informative data, which results in a reduction of the possible information gain.

A half-way between these extremal cases discussed above (very adaptive vs. very conservative) build semi-supervised learning [8] or transductive learning [18]. In both cases, the goal is to exploit the information given by labeled as well as unlabeled data. In the given context this can be interpreted as training a prior using a small amount of labeled data in an off-line stage, which can help to gain further information from unlabeled samples acquired during the on-line stage. There has been much research within this field showing that such methods are beneficial for many applications. However, due to strong prior knowledge the adaptivity is limited hindering to acquire new information.

An alternative way would be to use multiple classifiers that operate on different views and perform co-training [3]. In co-training, two independent classifiers, which were trained on a small amount of labeled data, are used to label the unlabeled samples for the other classifier if their own decision is confident. Highly unreliable samples are not used to train the second classifier. In practice, the required in-

*This work was supported by the FFG project HIMONI under the COMET programme in co-operation with FTW, the FFG project SECRECT under the Austrian Security Research Programme KIRAS, and the Austrian Science Fund (FWF) under the doctoral program Confluence of Vision and Graphics W1209.

dependent views are often not available, which violates the theoretical constraints for convergence, and the initial classifiers are too weak to allow for robust learning. In contrast, Multiple-Instance Learning (*e.g*., [5]) inherently copes with the problem of unreliable labeled samples. In particular, the single instances are organized in constrained bags, where a positive bag has to contain at least one positive sample and a negative bag consists only of negative samples. This solves the problem of inaccurately aligned samples typically occurring in, *e.g*., tracking, but they cannot handle unreliably labeled negative samples and occlusions.

Hence, existing methods to include new (unreliable labeled) samples are either too firm hindering to acquire new information or too adaptive tending to drift. Moreover, even using a strong prior more sophisticated semi-supervised methods can fail if false positives (fitting to the prior) are used for updating the classifiers. In contrast, in this paper we combine reliable knowledge (gathered off-line) with unreliable information (acquired on-line). The main idea is to model certain and uncertain samples within different classes in a multi-class representation, while still preserving binary update and evaluation strategies. Since the uncertain data can be considered as transient information, we refer to the method as *On-line TransientBoost*. Thus, we can assure robustness (*i.e*., avoid long-term drifting), but in contrast to existing approaches, we are able to include new (totally orthogonal) information, especially increasing the recall.

In the experiments, we demonstrate our approach for two typical tasks requiring an adaptive system: object detection in video streams and visual tracking-by-detection. For both applications we give a comparison to existing approaches on standard benchmark data sets showing the benefits of the approach, in particular, increasing the recall while preserving the accuracy. Moreover, to demonstrate the stability and the robustness of the presented approach, we run longterm experiments (600,000 and 3,000 frames for detection and tracking task, respectively) with continuously updating classifiers.

2. On-line TransientBoost

In the following, we introduce a new on-line boosting algorithm, which is built on on-line GradientBoost [11]. Similar to Saffari *et al*. [17] we introduce an on-line multi-class booster, however, adding the capability to cope with reliable and unreliable (transient) information in parallel.

Given a loss function $\ell(\cdot)$ and a labeled dataset, $\mathcal{X} = \{(x_1, y_1), \cdots, (x_N, y_N)\}, x_n \in \mathbb{R}^D, y_n \in \{-1, +1\}$ the goal of GradientBoost is to estimate a strong classifier $F(x)$ as a linear combination of M weak learners $f_m(x)$ minimizing this loss. Hence, at stage t, we searching a base function f_t which maximizes the correlation with negative direction of the loss function:

$$f_t(x) = \arg\max_{f(x)} \; -\nabla\mathcal{L}^T f(x), \tag{1}$$

where $\nabla\mathcal{L}$ is the gradient vector of the loss at $F_{t-1}(x) = \sum_{m=1}^{t-1} f_m(x)$. This can be simplified to

$$f_t(x) = \arg\max_{f(x)} \; -\sum_{n=1}^{N} y_n \underbrace{\ell'(y_n F_{t-1}(x_n))}_{-w_n} f(x_n), \tag{2}$$

where $\ell'(\cdot)$ are the derivatives of the loss with respect to F_{t-1} and w_n are the sample's weights. Optimizing Eq. (2) is independent of the applied loss function.

This formulation can simply be adopted for the on-line domain by using selectors as introduced in [9], where each selector $s_m(x)$ consists of N weak classifiers $\{f_{m,1}(x), \cdots, f_{m,N}(x)\}$ and is represented by its best weak classifier $f_{m,k}(x)$. The optimization step in Eq. (2) is then performed iteratively by propagating the samples through the selectors and updating the weight estimate w_n according to the negative derivative of the loss function.

On-line GradientBoost was designed for a binary classification problem. However, by introducing weak learners that are able to handle more than two classes, we can extend it to the multi-class domain. In general, any weak learner providing confidence-rated responses can be applied, however, we use histogram-based classifier. This choice is based on Friedman *et al*. [7], who used symmetric multiple logistic transformation as weak learner for a J-class problem:

$$f_j(x) = \log \; p_j(x) - \frac{1}{J}\sum_{l=1}^{J} \log p_l(x) \; , \tag{3}$$

where $p_j(x) = P(y_j = 1|x)$. In particular, they showed that if the sum over the weak classifier responses over all classes is normalized to zero, *i.e*., $\sum_{j=1}^{J} f_j(x) = 0$, the probability $p_j(x)$ can be estimated by using histograms. Moreover, histograms are highly appropriate for on-line learning since they can easily be updated.

Now having a multi-class formulation reliable and unreliable data can be modeled using different classes, *i.e*., $y = [+1, -1]$ for the reliable data and $y = [+2, -2]$ for the unreliable data. Thus, during an update the classifier is provided a sample x_t and a label $y_t \in \{-2, -1, +1, +2\}$ and depending on the label the corresponding histograms are updated. Moreover, for the reliable samples the histogram updates are performed incrementally whereas for the unreliable transient samples an iir-like filtering of the histogram bins is applied (*i.e*., the knowledge is scaled down according to its age). In this way the reliable information is accumulated whereas the unreliable information allows higher adaptivity, but is fading out quickly (depending on the forgetting rate f), thus, avoiding drifting.

The next, crucial step is to include the uncertainty of the sample $\langle x_t, y_t \rangle$ into the feature selection procedure. In each update step, similar to the binary case, the best weak classifier $f_{m,k}$ within a selector s_m is estimated according to its error. The error is updated depending on the weight of the correct classified samples $\lambda^c_{m,n}$ and the misclassified samples $\lambda^c_{m,n}$ within each weak classifier.

However, the error updates must be adapted according to the multi-class formulation. If the prediction was correct, *i.e.*, the signum of the classifier response $f_{m,n}$ equals the signum of class label used to update the classifier y_t ($sign(f_{m,n}(x)) = sign(y_t)$), the weight $\lambda^c_{m,n}$ is updated:

$$\lambda^c_{m,n} = \lambda^c_{m,n} + w_n \ ; \tag{4}$$

otherwise the weight $\lambda^w_{m,n}$ is updated:

$$\lambda^w_{m,n} = \lambda^w_{m,n} + w_n \ , \tag{5}$$

where w_n is the current estimated weight of the current sample. In the original GradientBoost algorithm any differentiable loss function ℓ can be used to update the weight by $w_n = -\ell'(y_t F_m(x_t))$, where $F_m(x) = \sum_{t=1}^{m} s_t(x)$ is the combination of the first m weak classifiers and y_t is the label of the current sample. In our case, however, we have to re-formulate the weight update according to our multi-class model. Otherwise the classifier would try to distinguish between the reliable and the unreliable classes and would penalize samples that are already classified correctly. Hence, since we are interested in discrimination of positive and negative classes, we have to change the weight update to

$$w_n = -\ell' \left(sign(y_t) F_m(x_t) \right) \ . \tag{6}$$

The derived update procedure for TransientBoost is summarized more formally in Algorithm 1. To finally obtain a binary classification result, during evaluation a sample is classified based on the signum of the classifier's prediction.

Algorithm 1 On-line TransientBoost Update

Require: sample x_t, label $y_t \in \{\pm 1, \pm 2\}$, model F^{t-1}
Output: updated model F^t

Set initial weight $w_0 = -\ell'(0)$
for $m = 1$ to M **do**
  for $n = 1$ to N **do**
    Train multi-class weak learner $f_{m,n}(x)$ with sample (x_t, y_t, w_n)
    if $sign(f_{m,n}(x_t)) = sign(y_t)$ **then**
      $\lambda^c_{m,n} = \lambda^c_{m,n} + w_n$
    else
      $\lambda^w_{m,n} = \lambda^w_{m,n} + w_n$
    end if
  end for
  Find best weak learner: $k = \arg\min_n \dfrac{\lambda^w_{m,n}}{\lambda^c_{m,n} + \lambda^w_{m,n}}$
  Set $s_m(x_t) = f_{m,k}(x_t)$
  Set $F^t_m(x_t) = F^t_{m-1}(x_t) + s_m(x_t)$
  Set $w_n = -\ell'(sign(y_t) F^t_m(x_t))$
end for

3. Applications

We demonstrate our algorithm on two different applications, *i.e.*, object detection and object tracking. For both problems we compare our algorithm to state-of-the-art approaches. In particular, we run experiments on common benchmark datasets and additionally show results obtained for long-term scenarios, where we demonstrate that TransientBoost is ideally suited to combine reliably gathered labeled data with unreliably gained scene specific data. For all experiments we use Haar-like features and classifiers with a size of at most 50 selectors, each of it containing 30 weak classifiers.

3.1. Scene-specific Object Detection

First, we demonstrate the proposed algorithm for learning an adaptive pedestrian detector for stationary cameras. In particular, we build on the scene specific classifier grid approach (CG) [16], where the key idea is to simplify the problem of object detection by dividing the whole image into small, highly overlapping grid elements, each holding a separate classifier. To ensure robustness the positive information is pre-trained and kept fixed and only negative updates are performed.

In the following, however, we show that robustly incorporating also positive scene-specific samples can be beneficial. To generate these samples, we use a labeler which is initialized by co-training. Similar to Levin *et al.* [12] two classifiers are co-trained on background-subtracted images and the current gray-value images. In contrast, after an initial stage this co-trained classifier is kept fixed and used to generate the updates for the classifier grid (co-grid). Since the labeler still provides a small number of wrong or inaccurate updates a robust learning method would be beneficial.

Thus, in the following we compare the proposed TransientBoost with GradientBoost, where the updates are generated by using the co-grid labeler. In particular, for TransientBoost the reliable classes $+1$ and -1 are initialized using a small amount of labeled samples whereas the samples generated by the labeler are modeled by the unreliable transient classes $+2$ and -2. In contrast, GradientBoost is initialized in the same way (*i.e.*, the weak learners contain the same features and the same statistics for reliable classes), however, these models are updated later on. Moreover, since for both approaches the loss functions can be changed on the fly, to increase the robustness for the negative updates and positive updates an exponential and a logit

loss function are applied, respectively. In addition, to have a baseline, we also run two generic state-of-the-art object detectors: the Dalal and Triggs [1] (DT) pedestrian detector [4] and the deformable part model of Felzenszwalb *et al.*[2](FS) [6]. Both detectors do not use any scene specific specific knowledge.

Longterm Benchmark

First, to show the robustness over time, we run experiments on our publicly available long-term dataset[3]. The dataset, showing a corridor in a public building, consists of 580.000 frames (*i.e.*, 7 days, 1fps) with a resolution of 320x240. For evaluation purposes we selected three specific sequences: *Sequence 1* starts after frame 3.390, right at the beginning of the sequence, and consists of 2.500 frames containing 201 pedestrians. *Sequence 2* is in the middle of the video, starting at frame 105.000. This sequence consisting of 5000 frames, containing 670 pedestrians, was selected, since it is very challenging (*i.e.* shadows and highlights are moving through the scene), which typically hampers on-line learning. *Sequence 3* is at the end of the dataset starting with frame 575.000. This sequence contains 2.500 frames containing 316 pedestrians. To demonstrate the longterm behavior of the different methods all on-line methods are updated throughout all 580.000 frames. The thus obtained results are presented in form of recall-precision curves (RPC) in Figures 1, 2, and 3.

From Figures 1 and 3 it can be seen that TransientBoost provides more or less the same performance as the CG approach, which does not use any positive updates. Even though the recall is not increased (this can be explained by complexity of the scenes) the precision is not decreased even when running positive updates. In contrast, Figure 2 clearly shows that TransientBoost provides excellent results whereas, even though recovering later on as can be seen from Figure 3, CG totally fails. Since Sequence 2 is characterized by heavily changes in the environment (moving shadows), this demonstrates the robustness our approach (this also explains the overall worse results compared to the other sequences). In general, Figures 1, 2, and 3 shows that TransientBoost clearly outperforms GradientBoost, especially in terms of precision. Moreover, on all three sequences the static detectors can be outperformed.

PETS 2006

In addition, we evaluated our approach on the *PETS 2006* dataset[4]. We compared it to the two generic object detectors (FS and DT) as a baseline and the standard GradientBoost approach, which was initialized using the same clas-

[1]http://pascal.inrialpes.fr/soft/olt
[2]http://people.cs.uchicago.edu/~pff/latent
[3]http://lrs.icg.tugraz.at/datasets/longterm
[4]http://www.pets2006.net

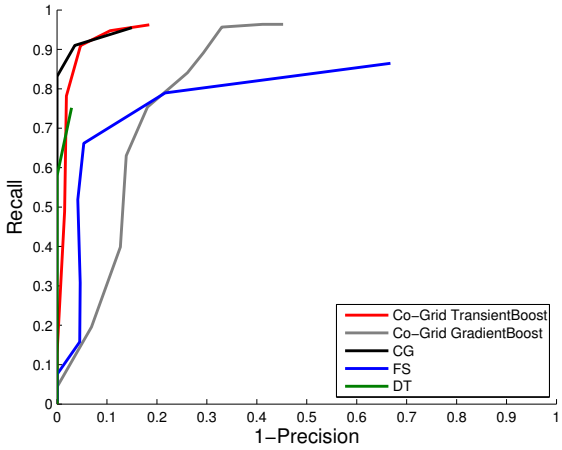

Figure 1. RPC for first evaluation part of the corridor sequence starting at 3.390 frames.

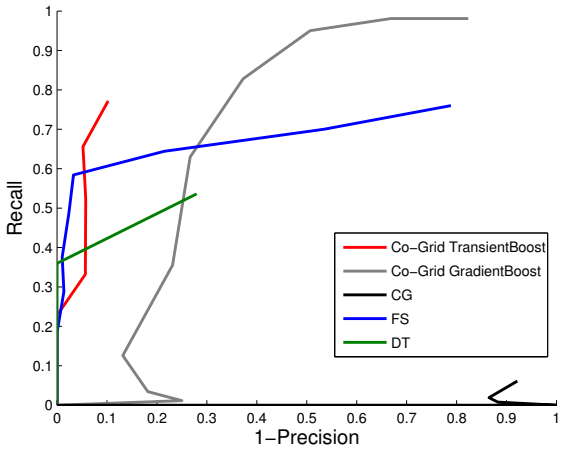

Figure 2. RPC for challenging second part of the corridor sequence containing moving shadows starting at frame 105.000.

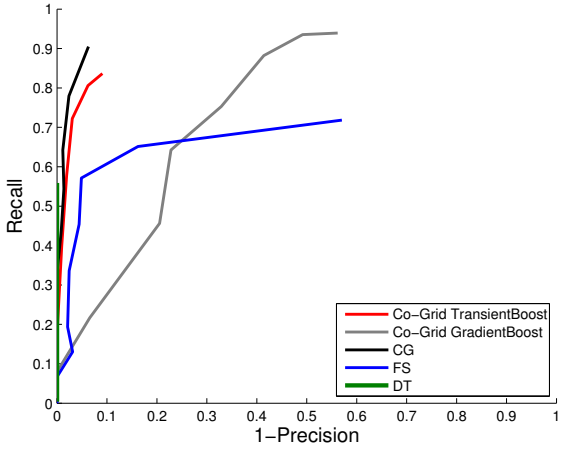

Figure 3. RPC for third part of the corridor sequence at the end of this longterm dataset starting of frame 575.000.

sifier. Again it can be seen from Figure 4 that the generic detectors can be clearly outperformed. However, in this case the additional positive updates drastically increase the recall (+40%). Since this sequence is pretty short, containing only 306 frames, the difference in performance between TransientBoost and GradientBoost shown is smaller. Due to the small number of sub-optimal updates the classifier is not totally degenerated but, as can be seen, there is a high number of false positives at high confidence. Illustrative results for this data set are shown in Figure 5.

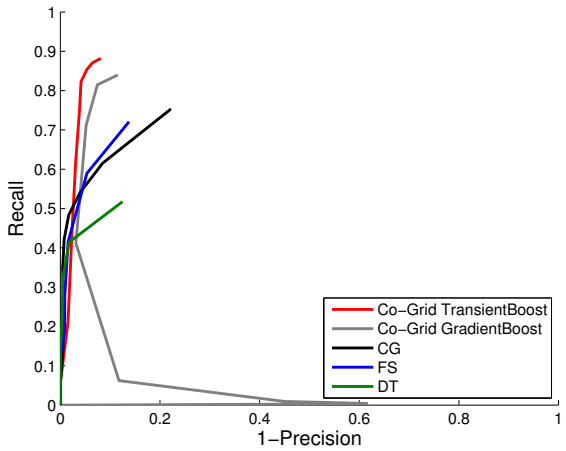

Figure 4. RPC for the *PETS 2006 Sequence*.

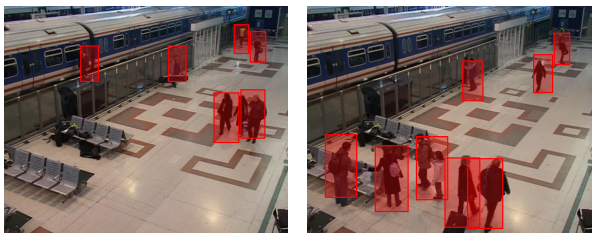

Figure 5. Illustrative detection results of our approach for the *PETS 2006* sequence.

3.2. Object Tracking-by-Detection

Second, we demonstrate the performance of our approach on the tracking-by-detection task. To start the tracking process, we initialize the classifier (*i.e.*, classes ± 1) with labeled data (virtual object samples) from the first frame. After this initial training, the continuous updates of the classifier are performed by self-learning, *i.e.*, the correctness of those samples cannot be guaranteed. Therefore, we define all samples as unreliable, and during runtime we update only the transient classes $y = [-2, +2]$ for the background as well as the foreground. In particular, to ensure the required adaptivity, we apply a forgetting rate $f = 0.1$. In the following, we show two experiments: (a) comparing our approach to state-of-the-art methods on publicly available benchmark data sets and (b) demonstrating the stability of our approach for a long-term tracking scenario.

Standard Benchmark Data Sets

For the first experiment, we use 8 publicly available sequences [1, 14] [5] covering different problems of tracking. The proposed approach (TRB) is compared to on-line SemiBoost (SEMI) [10], Multiple-Instance Boosting (MIL) [1], and on-line AdaBoost (OAB) [9]. The obtained results showing the average center location error in pixels compared to the ground-truth are depicted in Table 1. The results for MIL, OAB and SEMI are taken from [1]. Since the classifiers are initialized randomly, we give the results averaged over 5 runs.

Sequence	TRB	SEMI [10]	MIL [1]	OAB [9]
David	**19**	59	23	49
Sylvester	**10**	22	11	25
Face Occlusion	**8**	41	27	44
Face Occlusion 2	**12**	43	20	21
Girl	**20**	52	32	48
Tiger 1	21	46	**15**	35
Tiger 2	33	53	**17**	34
Coke	**19**	85	21	25

Table 1. Average location error: bold-face shows the best method (the lower the better).

It can be seen, that on-line AdaBoost [9] is not robust to noisy updates, which leads to drifting. On-line SemiBoost [10] performs well on sequences with low object variance (*e.g.*, *Sylvester*), but fails on highly diverse data (*e.g.*, *David*, *Girl*). Likewise on-line MILBoost [1] has limited adaptivity (*e.g.*, *Girl*), but in general it reaches good performance. In contrast, our approach performs best in 6 out of 8 sequences, which clearly show that we can cope with both, static appearance and dynamic changes in the scene. Moreover, due to reoccurring appearances which are covered by the reliable model, only the transient variations of the object have to be modeled, which allows to be highly adaptive considering the transient classes.

Longterm Tracking

To demonstrate the stability of our approach, we perform a longterm tracking experiment. Therefore, we evaluate the different approaches on a sequence with a length of 3000 frames[6]. The sequence contains variations of the object appearance and partial and full occlusions. As can be seen in Figure 6, trackers based on MILBoost and AdaBoost already drifted before frame 450, whereas TransientBoost is able to handle the variations of the object without drifting. Our method is also able to recover after full occlusions of

[5] Downloaded from http://vision.ucsd.edu/~bbabenko/project_miltrack.shtml

[6] Typical tracking sequences do not consist of more than 800 frames.

the object. Also SemiBoost is able to recover, but only supports limited adaptivity, resulting in temporary failures.

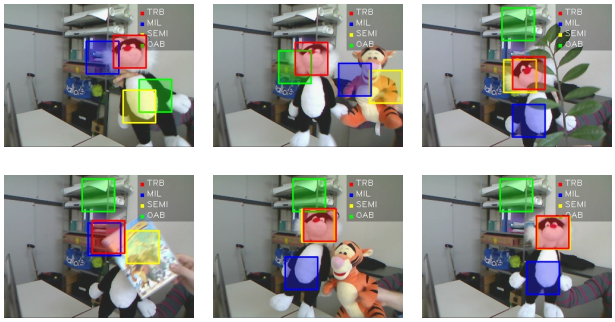

Figure 6. Frames 450, 950, 1400, 1800, 2400, and 3000 of the longterm tracking experiment (red: our approach; blue: MIL [1]; yellow: SEMI [10]; green: OAB [9]).

4. Conclusion

Existing approaches to include new samples into an on-line learning process are either too firm (*e.g.*, semi-supervised algorithms which are biased by a prior) or too unstable (*e.g.*, self-learning which is predicting its own labels). Thus, the goal of this work was to robustly incorporate unreliably labeled samples while still preserving the required adaptivity. In particular, we introduced on-line TransientBoost, which allows to inherently combine reliable (labeled) data and unreliable (on-line) information within one model. This is realized by using an internal multi-class representation allowing to deal with transient changes within the data whereas preserving the reliable (pre-trained) data. In fact, since the (possible unreliable) transient information is fading out very quickly, we get a highly adaptive system but can assure robustness due to the firm model trained from the reliable data. To demonstrate the benefits of the proposed approach we applied it for learning an adaptive object detector and for tracking. In both cases unreliable data has to be included, either by using a co-trained oracle or by self-training, where we showed that we can cope with this data considerable better than existing methods. In addition, to demonstrate the stability we showed two long-term experiments.

References

[1] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online mulitple instance learning. In *Proc. CVPR*, 2009.

[2] A. Baumberg and D. Hogg. Learning flexible models from image sequences. In *Proc. ECCV*, 1994.

[3] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. COLT*, 1998.

[4] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, 2005.

[5] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 1997.

[6] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Proc. CVPR*, 2008.

[7] J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: a statistical view of boosting. *The Annals of Statistics*, 2000.

[8] A. B. Goldberg, M. Li, and X. Zhu. Online manifold regularization: A new learning setting and empirical study. In *Proc. European Conf. on Machine Learning and Knowledge Discovery in Databases*, 2008.

[9] H. Grabner and H. Bischof. On-line boosting and vision. In *Proc. CVPR*, 2006.

[10] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *Proc. ECCV*, 2008.

[11] C. Leistner, A. Saffari A. A., P. M. Roth, and H. Bischof. On robustness of on-line boosting - a competitive study. In *Proc. IEEE On-line Learning for Computer Vision Workshop*, 2009.

[12] A. Levin, P. Viola, and Y. Freund. Unsupervised improvement of visual detectors using co-training. In *Proc. ICCV*, 2003.

[13] L.-J. Li, G. Wang, and L. Fei-Fei. Optimol: automatic online picture collection via incremental model learning. In *Proc. CVPR*, 2007.

[14] J. Lim, D. Ross, R. Lin, and M. Yang. Incremental learning for visual tracking. In *NIPS*. 2005.

[15] V. Nair and J. J. Clark. An unsupervised, online learning framework for moving object detection. In *Proc. CVPR*, 2004.

[16] P. M. Roth, S. Sternig, H. Grabner, and H. Bischof. Classifier grids for robust adaptive object detection. In *Proc. CVPR*, 2009.

[17] A. Saffari, M. Godec, T. Pock, C. Leistner, and H. Bischof. Online Multi-Class LPBoost. In *Proc. CVPR*, 2010.

[18] V. Vapnik. *Statistical Learning Theory*. Wiley, 1998.