Multiple Instance Learning from Multiple Cameras

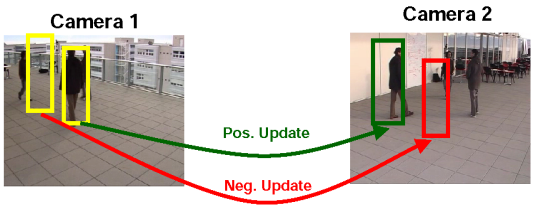
Peter M. Roth, Christian Leistner, Armin Berger, and Horst Bischof
Graz University of Technology
Institute for Computer Graphics and Vision
{pmroth,leistner,bischof}@icg.tugraz.at, armin.berger@student.tugraz.at

Abstract

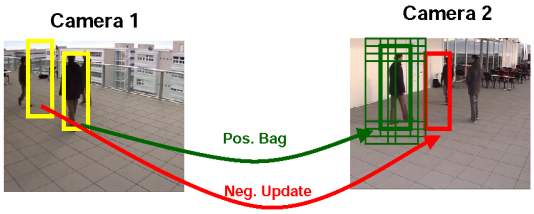
Recently, combining information from multiple cameras has shown to be very beneficial for object detection and tracking. In contrast, the goal of this work is to train detectors exploiting the vast amount of unlabeled data given by geometry information of a specific multiple camera setup. Starting from a small number of positive training samples, we apply a co-training strategy in order to generate new very valuable samples from unlabeled data that could not be obtained otherwise. To compensate for unreliable updates and to increase the detection power, we introduce a new online multiple instance co-training algorithm. The approach, although not limited to this application, is demonstrated for learning a person detector on different challenging scenarios. In particular, we give a detailed analysis of the learning process and show that by applying the proposed approach we can train state-of-the-art person detectors.

1. Introduction

Object detection and tracking are important tasks in computer vision typically run on a single camera. However, due to the increasing number of cameras mounted for security reasons it gets feasible in praxis to exploit the information of multiple cameras for detection and tracking. In fact, several real-world constraints such as the presence of a ground plane, a 3D scene model, or consistent appearance information across cameras provide beneficial information for these tasks [24]. Hence, there has been a considerable interest in object detection and tracking within multiple cameras networks [6, 10, 12, 14, 16, 17]. These methods mainly address the problem of occlusions that cannot be handled by single view approaches. For that purpose, they first apply change detection (*e.g.*, [12, 16, 17]) or a fixed pre-trained classifier (*e.g.*, [6]) to estimate the foreground likelihood of specific pixels. Then, this information is fused exploiting the common ground plane by either estimating a score map (*e.g.*, [6, 12, 16]) or by estimating axes intersections (*e.g.*, [17]). For an overview see also [16, 24].



(a) Standard approach.



(b) MIL approach.

Figure 1. Multiple Camera MIL Co-training: (a) The shared geometry of two cameras allows for co-training. (b) To compensate projection errors and to acquire more appropriate positive samples multiple instance learning is applied.

Most of the previous methods applied multiple-view geometry in order to improve the detection or tracking results of a priori given models (*e.g.*, a fixed off-line trained detector). However, this information can also be used to adaptively learn a detector as was shown by Leistner *et al.* [18]. The key idea is that each camera holds a separate classifier and that the cameras continuously co-train on unlabeled data from the shared scenario by exchanging foot-points of detections using camera-to-camera homographies.

Even though shown to be beneficial, in practice, this approach would easily fail due to projections errors. These are either the result of sub-optimal aligned (however correct) detections resulting from a sliding window detection approach or extreme camera angels inhibiting to accurately estimate corresponding points on the ground-plane. This is

not much of a problem if the detections are not used for further processing. However, as illustrated in Figure 1(a), if the thus estimated patches are used for learning, this leads to sub-optimal updates degrading the classification performance. Moreover, in an adaptive on-line learning scenario those errors are accumulated, which finally results in the total failure of the system.

To overcome these problems, in this paper, we introduce a robust multiple camera learning approach based on multiple-instance-learning (MIL) [9]. In MIL, training samples are provided in form of bags, where each bag consists of several instances. Labels are provided only for the bags, the labels of instances in positive bags are unknown and only constrained to that at least one instance has to be positive. For negative bags, all instances can be considered to be negative. The task of a multiple-instance learner is then to deliver a common instance classifier from this ambiguous labeled training data. For our multiple cameras learning problem, we first initialize several candidate positions around the foot-point of the estimated projection. In this way patches are collected which form a bag. Then, MIL is used to analyze this bag and to find the instance that is most likely the correct location of the object. This, is illustrated in Figure 1(b). In particular, due to its benefits for adaptive learning in this paper we introduce multiple camera MIL Boosting (MC-MILBoost) for learning the detectors.

In the experiments, we demonstrate that using multiple cameras can dramatically increase the stability of co-training; even if only very few labeled samples are available. In fact, due to the combination of geometric constraints and MIL only very valuable samples are selected for updating the classifiers, which results in state-of-the-art (single view) classification results. In particular, since we target at large-camera network applications, where the camera views are usually only slightly overlapping, the cameras do not collaborate during evaluation. Thus, these slight overlaps can be used for training, but since most camera views are not shared with other cameras, the detectors can also be applied if an object is visible in a single view only.

The remainder of this paper is as follows. First, in Section 2, we review the theoretical background of our work. Next, in Section 3, we introduce our new multi-camera MIL co-training system. Experimental evaluations are given in Section 4. Finally, we summarize and conclude the paper in Section 5.

2. Preliminaries

In the following, we give an overview of the related theory building the basis for our multi-camera co-training system, *i.e.*, co-training and multiple instance learning.

2.1. Co-Training

In supervised learning one deals with a labeled dataset $\mathcal{D}^L \subseteq \mathcal{X} \times \mathcal{Y} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_{|\mathcal{D}^L|}, y_{|\mathcal{D}^L|})\}$, where $\mathbf{x}_i \in \mathcal{X} = \mathbf{R}^P$ and $y_i \in \mathcal{Y} = \{+1, -1\}$. In contrast, unsupervised methods aim to find an interesting (natural) structure in $\mathcal{X}$ using only unlabeled input data $\mathcal{D}^U \subseteq \mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_{|\mathcal{D}^U|}\}$. Since most of the time unlabeled data can be obtained significantly easier than labeled samples, the main goal is to take advantage of the statistical properties of both labeled and unlabeled samples.

One widely applied method exploiting both labeled $\mathcal{D}^L$ and unlabeled $\mathcal{D}^U$ data, is co-training [7]. The main idea is that first two initial classifiers h_1 and h_2 are trained on labeled data $(\mathbf{x}_i, y_i), (\mathbf{x}_j, y_j) \in \mathcal{D}^L$. Then, these classifiers update each other using the unlabeled data set $\mathcal{D}^U$, if one classifier is confident on a sample whereas the other one is not. Abney [1] showed that co-training classifiers minimize the error on the labeled samples while increasing the agreement on the unlabeled data. Thus, the unlabeled data helps to improve the margin of the classifiers and to decrease the generalization error.

The approach has proven to converge [7], if two assumptions hold: (a) the error rate of each classifier is low and (b) the views must be conditionally independent. However, the second condition, which is hard to fulfill in practice, was later relaxed (*e.g.*, [1, 5]). For practical usage, this means that co-training can even be applied, if the learners are slightly correlated. Especially, computer vision naturally offers many physical "real-world" views, which can be exploited by co-training. Existing co-training approaches used for learning visual classifiers combined different simple cues based on shape, appearance, or motion (*e.g.*, [15, 20–22]). Thus, starting with Levin *et al.* [20], who indented to train a car detector, co-training was applied for various different applications such as learning a person detector (*e.g.*, [18, 21, 22]), tracking (*e.g.*, [15]), estimating a background model (*e.g.*, [26]).

2.2. Multiple Instance Learning

Multiple instance learning (MIL) [9] is a popular machine learning paradigm that deals with ambiguously labeled data. Thus, there has been a considerable interest in multiple-instance-learning and various different approaches have been proposed. Most of these approaches are based on popular supervised learning algorithms such as SVM [3] or boosting [25], that are adapted in order to incorporate the MIL constraints.

In multiple-instance learning training samples are given in bags $B_i, i = 1, \ldots, N$, where each bag may consist of an arbitrary number of instances $B_i = \{x_i^1, x_i^2, \ldots, x_i^{n_i}\}$, where $x_i \subset \mathbb{R}^d$ and n_i is the number of instances in-

side B_i. Negative bags B_i^- consist of only negative instances. Ambiguity is introduced into learning by the constraint that for positive bags B_i^+ it is only guaranteed that there exist at least one positive instance (also called *witness* of the bag). There is no information about the non-positive instances in the bag. In fact, they might not even belong to the negative class. The task is to learn either a bag classifier $f : B \rightarrow \{-1, 1\}$ or an instance classifier $f : \mathbb{R}^d \rightarrow \{-1, 1\}$. However, bag classification can follow automatically from instance prediction, *e.g.*, by using the *max* operator $p_i = \max_j\{p_{ij}\}$ over posterior probabilities over the instances p_{ij} of the i^{th} bag.

3. Learning from Multiple Cameras

The goal of this work is to exploit geometric constraints to acquire training samples, which cannot be obtained otherwise. For instance, there are various data sets containing frontal and back views of persons, however, from a multiple camera setup also side/semi-side views can be generated. In particular, we aim to co-train classifiers using the information from different camera views to finally obtain a general object detector via on-line learning. In particular, having overlapping camera views and a common ground-plane, the local image coordinate systems can be mapped onto each other by using a homography based on identified points in the ground-plane. This, is illustrated in Figure 2.

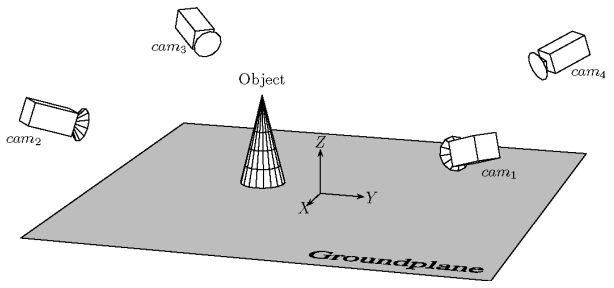


Figure 2. Geometric constraints, *i.e.*, homography information, can be used to exchange data between two camera views for learning object detectors.

However, such approaches have one main disadvantage: if the foot-points cannot be identified precisely enough the geometric information gets corrupted and the estimated projections are getting arbitrarily wrong (see Figure 1(a)). This might be the result of bad aligned detections or of extreme angles between the cameras. For instance, having two perfectly aligned detections, the foot-point in the first camera might be exactly at the toe-cap whereas the foot-point in the second camera is exactly on the heel. Hence, it is clear that even if having correct detections, the bilateral projections can fail, which is dangerous if those results are used for further processing, *i.e.*, learning.

These problems could be alleviated by using centralized fusion methods (*e.g.*, [6, 12], predicting the detections based on a probability map on the common top view of all cameras, or by running a motion-based verification step (*e.g.*, [21, 23]. However, centralized fusion methods are computational quite expensive and are limited to fixed object models (binary motion blobs or pre-trained detector). In contrast, motion-based verifiers are limited to simple scenarios (not containing occluded objects) and unnecessarily throw away too much information. Hence, both approaches are unsuitable in context of real-time learning from multiple cameras.

3.1. Multiple Camera Multiple-Instance Learning

To account the problems discussed above, in the following, we introduce an on-line multiple camera MIL boosting formulation (MC-MILBoost) allowing training from multiple cameras. The MIL approaches that are most similar to our methods are the off-line and on-line versions of MILBoost of Viola *et al.* [25] and Babenko *et al.* [4], respectively. MILBoost optimizes the binary log likelihood over bags in form of

$$log \quad \mathcal{L} = \sum_i \left(y_i \log p(y_i) + (1 - y_i) \log p(y_i)\right), \quad (1)$$

where the instance probability can be estimated using a sigmoid function: $p(y|x) = \sigma(H(x)) = \frac{1}{1+e^{-H(x)}}$. To solve this optimization problem over the bag, we apply gradient descent in function space. In order to identify the bag's witness this requires the max-operator $p_{ik} = \max_j \; p(c_k|s_{ij})$, which is, however, not differentiable. Therefore, we have to approximate the max-operator with a differentiable function. In our system, we apply the geometric mean

$$p_i = \left(\frac{1}{n_i}\sum_{j=1}^{n_i}(p_{ij})^{n_i}\right)^{\frac{1}{n_i}}. \quad (2)$$

In contrast to previous MIL approaches such as [3, 4, 25], we perform multiple-instance learning in a co-training framework. Thus, we do not only have to optimize the labels of instances inside positive bags, but also to find the correct bag labels using co-training. Hence, in order to solve these ambiguities over the bags, we additionally introduce a *most-likely-cause estimator* over the bags:

$$\mathbf{Pr}(y = +1|B_i) = \arg\max_{y \in \mathcal{Y}} \frac{1}{N}\sum_{n=1}^{N} p_n(y|x), \quad (3)$$

where $p_n(y|x)$ is estimated from the n^{th} strong classifier C_n in a multi-view setting. Again, we map the confidences of the classifiers to probabilities using $\sigma(C_n)$. Eq. (3) is

inferred into the bags of the individual views to try to optimize their corresponding classifier. Another difference to [25] and [4] is that we perform weight updates according to the negative derivative of the loss-function, which allows for more accurate weight updates, *i.e.*, we select the weak learners as

$$c_t(x) = \arg\max_{c(x)} - \bigtriangledown \mathfrak{L}^T c(x) \ .$$

3.2. Co-training system

The overall idea of our MC-MIL co-training system, *i.e.*, to robustly exchange detections results, is illustrated in Figure 1. To start the training process, we first train an initial classifier C^0 from a fixed set of positive and negative samples. This initial classifier is cloned and used for all camera views, $C_1^0, \ldots, C_n^0$. Then, we co-train the cloned classifiers $C_1, \ldots, C_n$ using confidence-rated predictions provided by the boosted classifiers. Using their current confidences, they can update each-other on an unlabeled sample in case of disagreement, *i.e.*, one classifier yields high confidence whereas the other does not. In addition, since also agreement was proven to lead to better performance (*e.g.*, [19]), we take also advantage of agreement learning.

For co-training, considering two specific views V_i and V_j, at each time step t we exploit geometric information, *i.e.*, the homographies $\mathbf{H}_{ij}$ and $\mathbf{H}_{ji}$. By using the homography information a specific sample $\mathbf{x}_i \in V_i$ is projected onto the other view V_j: $\mathbf{x}_j = \mathbf{H}_{ij}\mathbf{x}_i$. Based on the response of the current classifier $C_j^{t-1}(\mathbf{x}_j)$, we decide whether the sample $\mathbf{x}_i$ is considered a true or a false positive. If $\mathbf{x}_j$ was classified as true positive (on a high confidence) a bag of prospective positive samples is generated: $\mathbf{B}_j^+$. As illustrated in Figure 1(b), $\mathbf{B}_j^+$ is created from samples in the surrounding of the projected foot-point of $\mathbf{x}_j$. In this way it can be ensured that the correct patch (perfectly encapsulating the object-of-interest) is a member of $\mathbf{B}_j^+$. Then both classifiers C_i^{t-1} and C_j^{t-1} are updated using $\mathbf{B}_j^+$. In contrast, if $\mathbf{x}_j$ was classified as false positive, a negative bag $\mathbf{B}_j^-$ is generated from this sample and used as negative update for both classifiers[1]. The whole update strategy is summarized more formally in Algorithm 1

Algorithm 1 MC MIL Co-Training

Input: classifiers $C_1^{t-1}, \ldots, C_n^{t-1}$
Output: classifiers $C_1^t, \ldots, C_n^t$

for $i = 1, \ldots, n$ **do**
  eval C_i^{t-1} on V_i: K_i detections $\mathbf{x}_{i,k}$
end for
for $i \neq j : \forall \mathbf{x}_{i,k}$ **do**
  if $C_i^{t-1}(\mathbf{x}_{i,k}) > \theta$ **then**
    project $\mathbf{x}_{i,k}$ onto other views: $\mathbf{x}_j = \mathbf{H}_{ij}\mathbf{x}_{i,k}$
    compute $C_j^{t-1}(\mathbf{x}_j)$
    $//agreement$
    if $C_j^{t-1}(\mathbf{x}_j) = C_i^{t-1}(\mathbf{x}_{i,k})$ **then**
      generate pos. bag $\mathbf{B}_j^+$
      $milupdate\left(C_i^{t-1}, \mathbf{B}_j^+, +\right)$
      $milupdate\left(C_j^{t-1}, \mathbf{B}_j^+, +\right)$
    end if
    $//disagreement$
    if $C_j^{t-1}(\mathbf{x}_j) \neq C_i^{t-1}(\mathbf{x}_{i,k})$ **then**
      if $\left|C_j^{t-1}(\mathbf{x}_j)\right| > \left|C_i^{t-1}(\mathbf{x}_{i,k})\right|$ **then**
        $milupdate\left(C_i^{t-1}, \mathbf{x}_j, -\right)$
      end if
    end if
  end if
end for

4. Experimental Results

To show the benefits of our approach, we demonstrate it for the task of person detection. Note that our approach is not limited to this task; however for this scenario exist various reference implementations and benchmark datasets, which allows for an extensive experimental comparison. As a low level representation we use simple Haar-features, however, more sophisticated features can be applied. But as can be seen from the results, even using this simple representation provides competitive results.

The experiments are split into three main parts. First, we give an analysis of the learning behavior during MC-MIL co-training. Second, we compare our approach to existing adaptive and fixed person detectors. Third, we show that with our method a classifier trained on a specific multi-camera setup is also generalizing to setups not observed during training. In fact, in both cases we finally obtain state-of-the-art detection results; even if only a small number of labeled positive samples was used! We compare our method to (a) state-of-the-art person detectors, *i.e.*, the HOG-Detector of Dalal and Triggs [8] and the deformable part model of Felzenszwalb *et al.* [11], (b) to single view co-training methods, *i.e.*, the motion-based bootstrapping approach of Nair and Clark [21], the co-training approach of Levin *et al.* [20], and the conservative learning approach of Roth *et al.* [22], and (c) to the multi-camera approach of Leistner *et al.* [18].

[1] In general, exchanging the positive and negative updates between the classifiers is not necessarily required, but in this way more information can be gained.

4.1. Experimental Setup

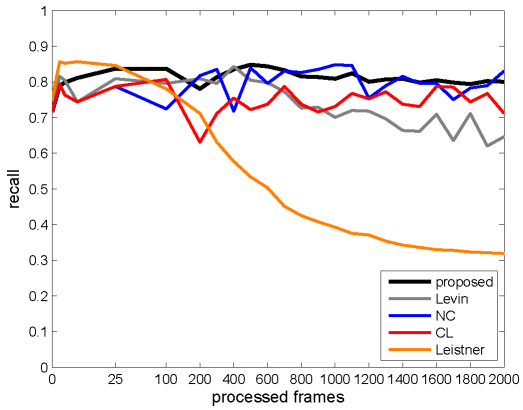
The experiments were performed on two data sets differing in complexity, view point/angle, and geometry. For both scenarios a scene is observed by three static cameras with partly overlapping views. The first data set (*Lab Scenario*) we have generated in our laboratory showing crowds of persons. The second data set (*Forecourt Scenario*) was thank worthy provided by the authors of [6]. For both datasets, we generated independent training and test sequences and estimated the homographies for all camera views sharing a view-point area according to [13]. In addition, the test sequences were annotated to enable an automatic evaluation.

4.2. On-line Learning in MC Networks

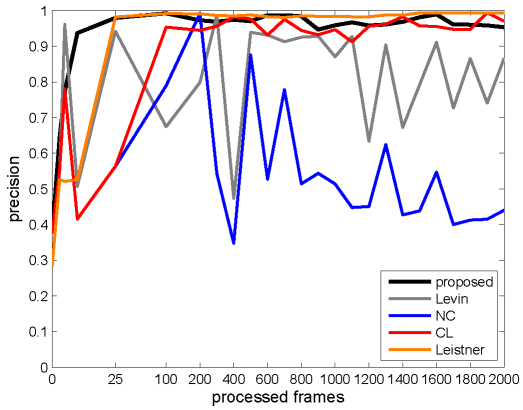
First of all, we demonstrate the on-line learning behavior of the proposed approach based on experiments that we carried out on the *Lab Scenario*. For that purpose, we trained an initial classifier using a fixed training set of 25 positive and 25 negative samples, which were randomly chosen from a larger dataset (*i.e.*, MIT-CMU). The reason for selecting only such a small amount of labeled samples, though, especially for the task of person detection, huge databases are available, is to demonstrate that our method can also be applied if only a small amount of labeled data is available. The classifier was cloned and used to initialize the co-training process for each camera. Later these initial classifiers were updated by co-training.

To demonstrate the learning progress, after a pre-defined number of processed training frames we saved the corresponding classifier, which was then evaluated on an independent test sequence (*i.e.*, the current classifier was evaluated but no updates were performed.). Please note, there is no collaboration between different cameras during the evaluation. From these results for each classifier we computed the *precision*, the *recall*, and the *F-measure* as proposed in [2]. The thus obtained results for recall and precision over time are shown in Figure 3. In addition, we give a comparison to single view learning methods, *i.e.*, Nair and Clark [21] (NC), Levin *et al.* [20] (Levin), and conservative learning of Roth *et al.* [22] (CL), and the multi-view method of Leistner *et al.* [18] (Leistner).

It clearly can be seen that even if starting from a rather bad classifier finally a competitive classifier can be obtained. In fact, by using the geometric constraint very valuable samples can be generated whereas the MIL constraint assures the robustness during learning. This results in a very fast convergence and a stable learning behavior, *i.e.*, the recall can be increased while the precision stays stable over time.



(a)



(b)

Figure 3. Performance characteristics over time for different online (co-training) methods: (a) recall and (b) precision.

In contrast, the other methods (*i.e.*, [20, 21]) suffer from an instable behavior, especially concerning the precision. Since these methods are based on a background model, this results from too much and often unreliable and wrong updates. Thus, such methods are not applicable in practice if the complexity of the scene becomes to high. In contrast, for [22] the precision is high, but due to the conservative update strategy less positive updates are performed and the recall cannot be increased. For the basic multi-camera approach [18] as a result of sub-optimal updates, it can be seen that even though the precision is excellent the recall is decreasing over time. This clearly shows the benefits of the proposed approach in term of information gain and stability.

For reasons of completeness, in Table 1 we further give the detections characteristics for all three camera views obtained on the test sequences. Comparing the initial and final characteristics, it clearly can be seen that the detection performance can be improved. This especially applies for the precision!

	rec.	prec.	F-M
v1	0.76	0.35	0.48
v2	0.77	0.43	0.55
v3	0.77	0.45	0.57

(a)

	rec.	prec.	F-M
v1	0.83	0.97	0.89
v2	0.86	0.93	0.89
v3	0.86	0.92	0.89

(b)

Table 1. Performance characteristics for the *Lab Scenario* for all three camera views: (a) initial classifiers and (b) final classifiers.

4.3. Scene-specific Detection Task

After analyzing the stability of the proposed method, we give a competitive study compared to state-of-the-art person detectors for the *Lab Scenario* as well as for the *Forecourt Scenario*. In addition to the adaptive methods described above, we compared the results to fixed persons detectors, *i.e.*, the Dalal & Triggs [8] person detector and the person detector trained by using the deformable part model of Felzenszwalb *et al.* [11]. To compare the different approaches we use precision-recall curves. For the adaptive methods the classifiers were trained on the same training data as the proposed method and the finally obtained classifiers were evaluated on the test sequences. Again for multiple-camera co-training during the evaluation the classifiers did not collaborate. The obtained precision-recall curves for the test sequences, respectively, are shown in Figures 4 and 5 (solid lines indicate adaptive methods, doted lines fixed detectors).

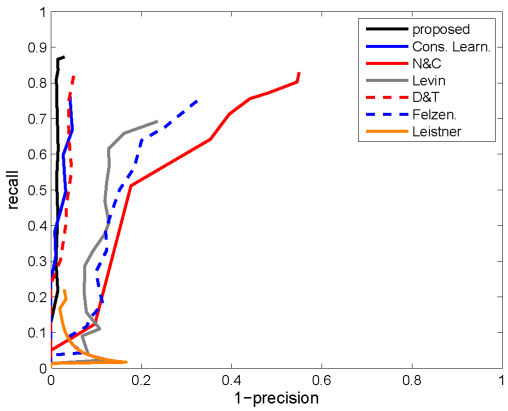


Figure 4. Precision-recall curves for different approaches obtained on the *Lab Scenario*.

For both datasets it can be seen that competitive results can be obtained and that other (single-view and multiple-view learning) approaches trained on the same data can be outperformed. This clearly shows that the training samples acquired during the MC-MIL co-training are highly valuable. Finally, some illustrative detection results of the final classifiers obtained for both test sets are given in Figure 6.

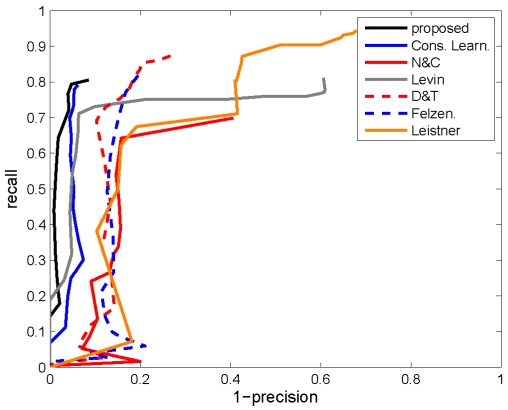
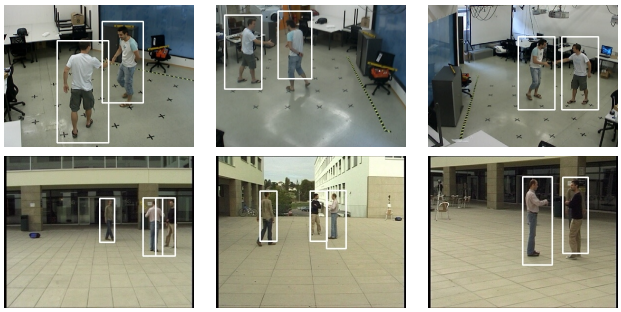


Figure 5. Precision-recall curves for different approaches obtained on the *Forecourt Scenario*.



view 1 view 2 view 3

Figure 6. Illustrative examples of detection results obtained by the final detectors on the *Lab Scenario* (first row) and the *Forecourt Scenario* (second row).

4.4. General Detection Task

Finally, we show that using the proposed approach not only scene/view specific classifiers can be trained, but that these classifiers are also generalizing to different views/scenarios. For that purpose, performed MC-MIL co-training on the *Lab Scenario* data set and applied the finally obtained classifier on two publicly available standard benchmark datasets, *i.e.*, the *PETS 2006*[2] and the *CAVIAR* dataset[3].

From the illustrative examples shown in Figure 9, it can be seen that the two scenarios are quite different to our training setup illustrated in Figure 6. This, clearly shows that the classifiers trained by our system are also generalizing to different detection setups. The thus obtained results, compared to fixed person detectors, *i.e.*, Felzenswalb *et al.* and Dalal & Triggs, are shown in form of PR curves in Figures 7 and 8. In particular, the higher recall, even though simpler features were used, can be explained by the fact that

[2]http://www.pets2006.net

[3]http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1

during the training also side and half-side views of persons are captured, which are typically not included in data sets used for training person detectors.

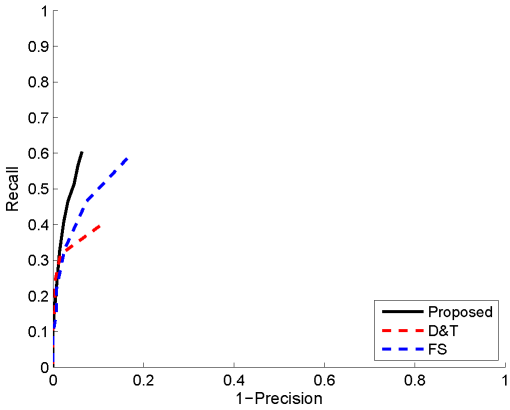


Figure 7. Precision-recall curves for the proposed approach and fixed detectors obtained on the *PETS 2006* data set.

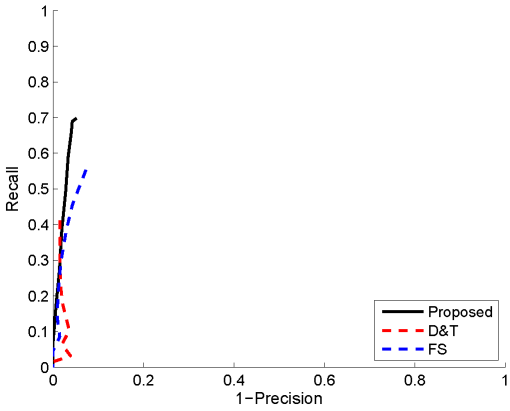


Figure 8. Precision-recall curves for the proposed approach and fixed detectors obtained on the *CAVIAR* data set.

5. Conclusion

In this paper, we introduced an approach for autonomously on-line learning of classifiers from multiple cameras, exploiting geometric constraints to exchange information between cameras. In particular, we use homography information between cameras to guide a co-training process. However, in contrast to existing methods the cameras collaborate only during training; during the evaluation the classifiers are run independently. To compensate projection errors arising from false or badly aligned detections, we introduce an on-line multiple instance co-training method. In this way, very valuable updates are generated, which allows for efficient and robust co-training. In the experiments,

(a)

(b)

Figure 9. Generalizing classifiers: (a) Illustrative detections results for (a) *CAVIAR* and (b) *PETS 2006* data sets.

we gave a detailed analysis on the learning behavior and show that we finally obtain state-of-the-art classification results. To show the generality of the approach, we run the experiments on various scenarios differing in complexity, view angle, etc. In addition, we show that by training within a multi-camera setup highly valuable positive samples are generated, which allows for training general detectors. Future work, will include to extend the proposed approach for a larger number of cameras and to apply it for more general detection tasks.

Acknowledgments

This work was supported by the FFG project EVis (813399) under the FIT-IT programme, by the FFG project SECRECT (821690) under the Austrian Security Research programme KIRAS, and the Austrian Science Fund (FWF) under the doctoral program Confluence of Vision and Graphics W1209.

References

[1] S. Abney. Bootstrapping. In *Proc. Annual Meeting of the Association for Computational Linguistics*, pages 360–367, 2002.

[2] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(11):1475–1490, 2004.

[3] S. Andrews, I. Tsochantaridis, and T. Hofmann. Support vector machines for multiple-instance learning. In

Advances in Neural Information Processing Systems, pages 561–568, 2003.

[4] B. Babenko, M.-H. Yang, and S. Belongie. Visual tracking with online mulitple instance learning. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2009.

[5] M.-F. Balcan, A. Blum, and K. Yang. Co-training and expansion: Towards bridging theory and practice. In *Advances in Neural Information Processing Systems*, pages 89–96, 2004.

[6] J. Berclaz, F. Fleuret, and P. Fua. Principled detection-by-classification from multiple views. In *Proc. Int'l Conf. on Computer Vision Theory and Applications*, pages 375–382, 2008.

[7] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. COLT*, pages 92–100, 1998.

[8] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume I, pages 886–893, 2005.

[9] T. G. Dietterich, R. H. Lathrop, and T. Lozano-Pérez. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1–2):31–71, 1997.

[10] R. Eshel and Y. Moses. Tracking in a dense crowd using multiple cameras. *Intern. Journal of Computer Vision*, 2010. (online first).

[11] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2008.

[12] F. Fleuret, J. Berclaz, R. Lengagne, and P. Fua. Multi-camera people tracking with a probabilistic occupancy map. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 30(2):267–282, 2008.

[13] R. Hartley and A. Zisserman. *Multiple View Geometry*. Cambridge University Press, 2003.

[14] M. Hu, J. Lou, W. Hu, and T. Tan. Multicamera correspondence based on principal axis of human body. In *Proc. IEEE Intern. Conf. on Image Processing*, volume II, pages 1057–1060, 2004.

[15] O. Javed, S. Ali, and M. Shah. Online detection and classification of moving objects using progressively improving detectors. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume I, pages 696–701, 2005.

[16] S. M. Khan and M. Shah. Tracking multiple occluding people by localizing on multiple scene planes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 31(3):505–519, 2009.

[17] K. Kim and L. Davis. Multi-camera tracking and segmentation of occluded people on ground plane using search-guided particle filtering. In *Proc. European Conf. on Computer Vision*, volume III, pages 98–109, 2006.

[18] C. Leistner, P. M. Roth, H. Grabner, A. Starzacher, H. Bischof, and B. Rinner. Visual on-line learning in distributed camera networks. In *Int'l Conf. on Distributed Smart Cameras*, 2008.

[19] B. Leskes and L. Torenvliet. The value of agreement a new boosting algorithm. *Journal of Computer and System Sciences*, 74(4):557–586, 2008.

[20] A. Levin, P. Viola, and Y. Freund. Unsupervised improvement of visual detectors using co-training. In *Proc. IEEE Intern. Conf. on Computer Vision*, volume I, pages 626–633, 2003.

[21] V. Nair and J. J. Clark. An unsupervised, online learning framework for moving object detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume II, pages 317–324, 2004.

[22] P. M. Roth, H. Grabner, D. Skočaj, H. Bischof, and A. Leonardis. On-line conservative learning for person detection. In *Proc. IEEE Intern. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pages 223–230, 2005.

[23] P. M. Roth, C. Leistner, H. Grabner, and H. Bischof. *Multi-Camera Networks, Principles and Applications*, chapter Online Learning of Person Detectors by Co-Training from Multiple Cameras, pages 313–334. Academic Press, 2009.

[24] A. C. Sankaranarayanan, A. Veeraraghavan, and R. Chellappa. Object detection, tracking and recognition for multiple smart cameras. *Proc. of the IEEE*, 96(10):1606–1624, 2008.

[25] P. Viola, J. C. Platt, and C. Zhang. Multiple instance boosting for object detection. In *Advances in Neural Information Processing Systems*, pages 1417–1426, 2005.

[26] Q. Zhu, S. Avidan, and K.-T. Cheng. Learning a sparse, corner-based representation for background modelling. In *Proc. IEEE Intern. Conf. on Computer Vision*, volume I, pages 678–685, 2005.