Exploring Context to Learn Scene Specific Object Detectors

Severin Stalder Helmut Grabner Luc Van Gool
Computer Vision Laboratory
ETH-Zurich, Switzerland
{sstalder,grabner,vgool}@vision.ee.ethz.ch

Abstract

Generic person detection is an ill-posed problem as context is widely ignored. Local context can be used to split the generic detection task into easier sub-problems, which was recently explored by classifier grids. The detection problem gets simplified spatially by training separate classifiers for each possible location in the image. So far, adaptive grid based approaches only focused on exploring the specific background class. In contrast, we propose a method using different types of context in order to collect scene specific samples from both, the background and *the object class over time. These samples are used to update the specific object detectors. Due to limiting label noise and avoiding direct feedback loops our system can robustly adapt to the scene without drifting. Results on the PETS 2009 dataset show significantly improved person detections, especially, during static and dynamic occlusions (*e.g.*, lamp poles and crowded scenes).*

1. Introduction

Robust visual object detection and tracking under real-world conditions is still an unsolved problem and limits the use of state-of-the-art methods in commercial systems, *e.g.*, for video surveillance applications [4]. The task is inherently difficult due to the variability in appearance of persons (*e.g.*, clothing, pose, illumination), in backgrounds (*e.g.*, clutter, static occlusions, moving objects) and in dynamical occlusions (*e.g.*, crowds [14], other moving objects).

A generic person detector (*e.g.*, [3, 6]) is usually designed to be applicable in any scenario. Hence, a large training set is required to capture all variability of persons and backgrounds. Not surprisingly, the main limitation of these detectors lies in gathering a representative training set.

Context could help to limit appearance changes and thus scaling down the training set. It is well known that context plays a very important role, *e.g.*, exactly the same image patch can be interpreted very differently depending on its embedding in the world [18]. A generic object detector tries to solve just the ill-posed problem of detecting the object of a class in any context [12]. Hence, generic detectors often fail in real world scenarios. In many application scenarios the detection problem would be far simpler. For example, in a 24/7 surveillance scenario the camera is often static and focuses always on one and the same scene. Further, there is a continuous data stream providing a huge amount of (unlabeled) data which should be explored for (i) improving detection results as well as (ii) speeding up the detection process.

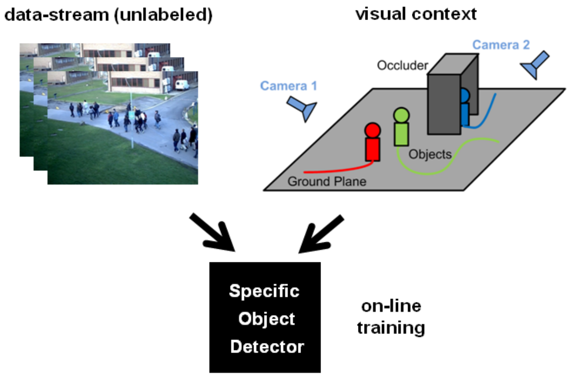


Figure 1. Having access to a large data-stream and using various types of context (*e.g.*, scene knowledge, tracking) our approach continuously updates an specific object detector.

One simple way to benefit from the static camera is to incorporate information about the particular scene (*e.g.*, using a ground plane to limit the size of persons). However, such information usually helps only to reduce the number of false alarms (*e.g.*, [13]). In order to increase also the detection rate, on-line methods adapting to a particular scene have been investigated (*e.g.*, [19]). These methods focus on solving the object detection task in the particular scene and take advantage from the continuous incoming data stream. In fact, these approaches use context (scene knowledge) already in the training process and not just as post-processing. Therefore, on-line unsupervised learning methods are usually used to continuously adapt the model. The main problem, however, is to robustly include the new unlabeled data.

If the data is wrongly interpreted, the performance of the detector will be reduced. In other words, the detector might drift and would end in an unreliable state.[1]

In most object detection systems a sliding window technique is used, *i.e.*, each patch of an image is tested if it is consistent with a previously estimated model or not. In contrast, Grabner and Roth et al. [10, 16] recently proposed to simplify the object detection task such that fixed update rules can be applied. In fact, a separate object detector is introduced at each image location. This approach can be applied to long term sequences for different object classes (*i.e.*, pedestrian and cars) without drifting. The limitations are due to (i) modeling the object class in a generic fashion (fixed object class distribution) and (ii) temporal drifting (similar to background modeling).

In this paper, we focus on how to cope with these problems in a more principled manner. In fact, we propose to not "blindly" use machine learning techniques, but to explore various context cues (*i.e.*, 2D, 3D and temporal context [5]) in order to train and improve scene specific object detectors in a robust manner. The principle is depicted in Fig. 1.

The paper is organized as follows. In Sec. 2 we review the basic concepts of on-line learning using classifier grids and discuss their limitations. Next, in Sec. 3 we present our new approach and three particular implementations using an increasing amount of context. Detailed evaluations of the proposed approach as well as results on the PETS 2009 person counting task are given in Sec. 4. Finally, we conclude and summarize the paper in Sec. 5.

2. Grid-based Detectors

The idea of classifier grids [10] is to reduce the complexity of the learning problem. In fact, a highly overlapping grid of classifiers (detectors) is put on the image. This allows to focus on the specific context in which the object detector has to perform. More formally, looking at the Bayes-formula for a two class classification problem (*i.e.*, $\mathbf{x} \in \mathbb{R}^n$ is the feature vector and $y \in \{-1, +1\}$ the class label of positive and negative samples, *i.e.*, object or background)

$$P(y|\mathbf{x}, c) = \frac{P(\mathbf{x}|y, c)P(y|c)}{P(\mathbf{x}|c)} \tag{1}$$

where c can be seen as context. The goal is to use context such that its easier to separate the two distributions $P(\mathbf{x}|y = 1, c)$ and $P(\mathbf{x}|y = -1, c)$. For simplicity reasons we will omit the notation of c in the following. However, the reader should keep in mind that context allows us to solve an "easier" sub-problem, *i.e.* to reduce inconsistencies. The main contribution of grid-based detectors comes from the fact that each single grid cell has only to consider its specific background and distinguish it from the class of objects at exactly that position. In the following, we briefly review the recent grid-based approaches for object detection. Finally we discuss their limits which motivates this work for object detection in challenging scenes with occlusions.

2.1. Fixed Update Classifier Grid

The principle of the original grid based detection approach [10] is depicted in Fig. 2(a). The approach was essentially inspired by a classifier grid based background model [7, 8]. Actually, the unlabeled samples are included with a fixed update rule interpreting all the incoming samples as background patches. Additionally, in order to focus on a particular object class, the positively labeled samples come from a set of object images (in the extreme case only one averaged object image was used) which is kept fixed. The fixed yet simple update rules ensure that the system is drift free by design (see discussion). On-line boosting for feature selection [7] is used for learning and updating the individual classifiers. However, it is not crucial to use this classifier, any other classifier could be used.

2.2. Generative Models Classifier Grid

Very recently, Roth et al. [16] extended the above reviewed approach, see Fig. 2(b). In fact, they take a closer look on the specific training algorithm, namely on-line boosting for feature selection. A generative model is built for each feature for both, the positive and the negative class. The model for the positive class is kept constant, whereas the model for the negative class is continuously updated using unlabeled data. These two generative models are then used to form a discriminative classifier by selecting a good subset from all possible features. Summarizing, the unlabeled samples (interpreted as negative data) are used to guide the feature selection process *directly*. The fixed statistic obtained from positive examples, so the argumentation, ensures a drift-free system (see discussion).

2.3. Discussion

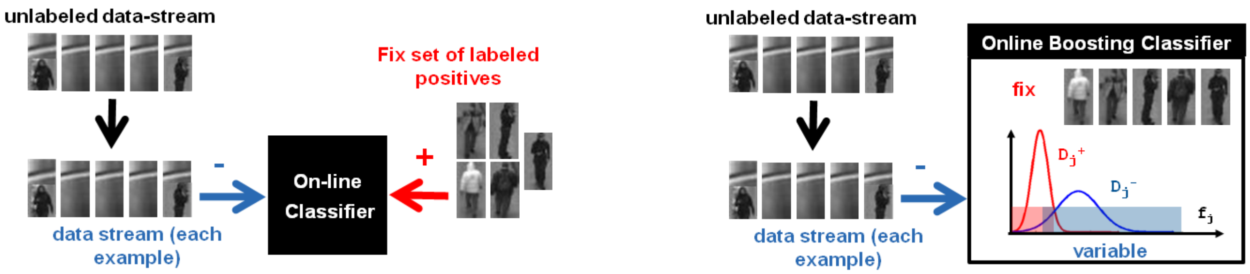
The underlying problem is to robustly integrate new unlabeled samples into the system during runtime. Actually, unlabeled data can only give us information about P(x) which is the sum over the marginal distributions:

$$P(\mathbf{x}) = P(\mathbf{x}|y = 1)P(y = 1) + P(\mathbf{x}|y = -1)P(y = -1). \tag{2}$$

As can be easily seen from Eq. (1), with no further assumptions one can not gain anything form unlabeled data![2] Obviously really interesting and useful examples are the ones

[1] The same problem appears in model-free visual object tracking [15, 9].

[2] This is strongly related to semi-supervised and transfer learning, (see [20] for an excellent survey), where one can think of build scene specific classifiers by transfering "general" knowledge to a particular task using unlabeled data. However, as noted above there are strong assumptions in a pure machine learning setting.


unlabeled data-stream
-
data stream (each example)
On-line Classifier
+
Fix set of labeled positives
unlabeled data-stream
-
data stream (each example)
Online Boosting Classifier
fix
D_j^+
D_j^-
f_j
variable

(a) Fixed update rules (b) Generative models

Figure 2. By using a classifier grid instead of a sliding windows approach the detection task gets simplified. This can be seen as using context to focus on an easier "subproblem" of the detection task. This enables specializing the detector using fixed, yet simple, update rules directly (a) or by using it inside a specific classifier (b).

for which the actual classifier predicts its label wrongly. These examples are the crucial ones which improve results for both, recall and precision of the detector. Without external supervision (assumptions about the data, scene/task knowledge or context) this can not be solved in a pure machine learning approach. In the following we take a look at the assumptions yielded by the grid-based approach and discuss their limitations.

Label noise. It was already mentioned in the original paper [10] that the negative updates are not always correct. Label noise, *e.g.* false negatives, are included with low probability. So if a person is standing at the same location for a certain time, it will get more and more difficult to detect the person. This is even more dramatically when using boosting for updating the classifier, since the method puts high weights on misclassified samples. This effect was limited by the generative models approach, as it is able to cope with some label noise on feature level. However, this could not be achieved in the (dominant) feature selection process itself. Therefore, weight limitation and fading memory was used (similar to [8]) in order to limit the effect. But, rigorously talking, *temporary*[3] drifting still existed. It was preferable to drift into the background class than into the object class. More formally, the main assumption is that the unlabeled data-stream corresponds to the negative class, *i.e.*,

$$P(\mathbf{x}) \sim P(\mathbf{x}|y=-1), \tag{3}$$

since it was assumed that the class prior $P(y=-1) \gg P(y=1)$. In fact, in both mentioned approaches *all* incoming samples are yielded as background. This is exactly what limits the approach since the information is even encoded at the likelihood term and not the prior term in Eq. (1).

Occlusion handling. Another problem of the aforementioned approaches comes from the fact that no scene specific positive samples are incorporated. The object class is always modeled in a generic fashion, *e.g.* by using a fixed/limited set or statistics obtained from it for the object class. Hence, no attention is paid to occlusions or appearance changes of the object which can not be handled by the generic detector. So the classifier can focus only on the scene specific negative class at the grid element, but not on the scene specific object appearance at that location.

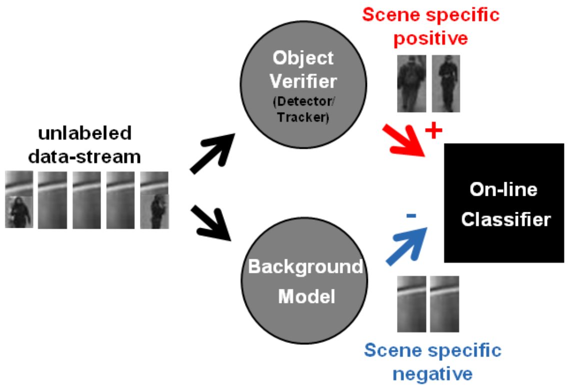


Figure 3. Proposed approach of context-based grid detector. The unlabeled data-stream is analyzed and scene specific positive and negative samples are collected for updating the classifier, *i.e.* a local grid detector.

3. Extended Context-based Grid Detectors

We make use of a static camera and the huge amount of unlabeled data to specialize the detector to the specific scene similar to the reviewed grid-based detection approaches. However, in our approach we would like to overcome the limitations discussed in the previous section.

The principle is depicted in Fig. 3, where the main aim of our approach is to split the unlabeled data stream $P(\mathbf{x})$ into the margin distributions, see Eq. (2). Further, we propose to not use every upcoming examples in any case (*i.e.*, either label it as positive or negative), since we can benefit from

[3] Note, due to the fixed update strategy the system was able to recover from failures after time, *i.e.*, the person has left and new updates fades away the old information.

Method		Positive updates	Negative updates
sliding window	general detector (*e.g.*, [6])	no	
	adaptive detector (*e.g.*, [19])	some sort of supervision	
classifier grid	background model [7]	natural image statistics	current patch
	object detection [10]	pre-defined positive set	current patch
	object detection ext. [16]	no (pre-calculated model)	current patch
classifier grid	our approach	verified patches	background image

Table 1. Comparison of the training stage of different person detection approaches. Our proposed method (last row) focuses on the context of the detection problem and explores both, scene specific positive and negative samples.

a lot of data which is accumulated during runtime. In other words, we do not use it directly in an naive version. In contrast, we introduce local pools for positive $\mathcal{X}_i^+$ and negative $\mathcal{X}_i^-$ samples at each grid location i. The pools are filled using a defined update strategy, which is described in general in the following and in more detail in the later subsections.

Positive Samples. Modeling the scene specific positives improves recall, which however is not obvious to incorporate robustly. Because the method has to (i) focus on the right object of interest and further (ii) the object also needs to be well aligned. Hence, we propose, to fill the pool of positive samples only by verified samples from the unlabeled data stream. The verifier has to approved the current image patch.

Negative Samples. Since the negative class usually dominates, negative examples are collected by applying a very conservative (long term) background model.

An update of the classifier is done with a positive sample and a negative sample from its local (specific) pools $\mathcal{X}_i^+$ and $\mathcal{X}_i^-$, respectively. In contrast to the former proposed methods (see Tab. 1), we (i) switch from a fixed set of positive samples to an adaptive (scene/location specific) set; and (ii) considerably reduce label noise via the background modeling for the negative class.

In the following we show three particular methods of the proposed system, exploring more and more context (or visual information) for the individual classifier grid elements and discuss strategies to acquire samples.

3.1. Classifier Grid and Detector

We take advantage of a generic detector and a simple background image to put the detections into context with the background at the specific scene and position.

Positive samples. A generic detector is applied to the image at each time instant. The eventual detections are used to update the grid classifier at the respective location. Note, the detector is trained on a fixed set, however, the samples for updating our local grid classifier come from the scene. Therefore, the grid classifier will not rely on generic models, but focus on the scene specific object appearances.

Negative samples. The pools of background patches are filled with a conservative background model and not from the data directly. Hence, we can reduce label noise considerably, *i.e.*, without any positive examples in the negative pool. Therefore, drifting into the negative class is limited.

3.2. Classifier Grid and Tracking

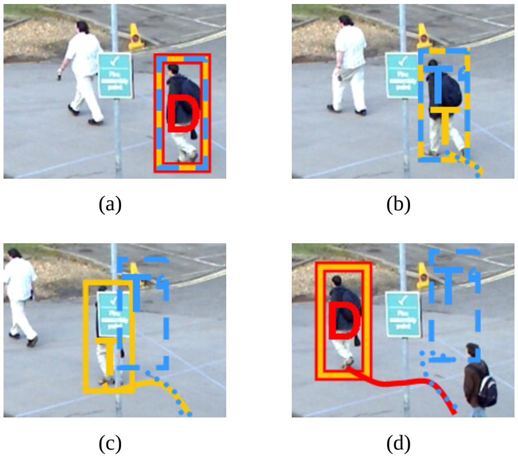
The object detector used previously is generic and probably fails in difficult situations (*e.g.*, occlusions or object appearances different from learned ones). One way to solve the occlusion problem is to take advantage of the temporal context, *e.g.* using object tracking to collect these informative positive examples where the generic detector fails.

Positive samples. Tracking results are accumulated for updating the grid detectors. Therefore, the same initial object detections are given to a tracker. The tracker tries to track the objects also under occlusions. This allows us to include information about the objects under occlusion in that specific scene instead of relying on generic models. With generic models it is hard to detect these objects. If the tracker drifts away from the object, wrong positive examples are gathered. To prevent such cases, classifiers only get updated if the tracklet is approved by another detection later on (Fig. 4). If no verification can be done for a user defined time, the tracker is stopped (yielded as drifted). Hence no samples are wrongly collected.

Negative samples. Static occluding objects and other tracked objects are present in both classes, foreground and background. Therefore, these image regions do not carry information for discrimination and the classifier will implicitly ignore them.

3.3. Classifier Grids and 3D-Context

Finally, we extend the detector grid with the 3D context to benefit especially in situations of occlusion. Fig. 5 depicts the principle. From one view point to another, we assume a common ground plane and map the foot-point of a detector or a tracker bounding box using a given homography [11], see Fig. 5. Unfortunately, the bounding box is



(a) (b) (c) (d)

Figure 4. Temporal context to verify tracklets: a person detector initializes 2 trackers (a), both trackers perform correctly (b), the orange tracker remains on the person, the blue one drifts away (c), but only the orange tracker gets verified by a detection (d). All intermediate samples will be considered as correct and will be included in the positive sample bag.

not always precisely aligned with the object. Therefore, the corresponding patch might not be located at the precise location of the person in the second image. We can deal with such inaccuracy by looking at a larger region in the second view. Both image patches are concatenated to form the new training sample. In fact, we are now coping with a 3D classifier grid in regions with overlap.

Positive samples: Tracking results in one view and their expected corresponding regions in the other view form together the patch used for to train the grid detectors.

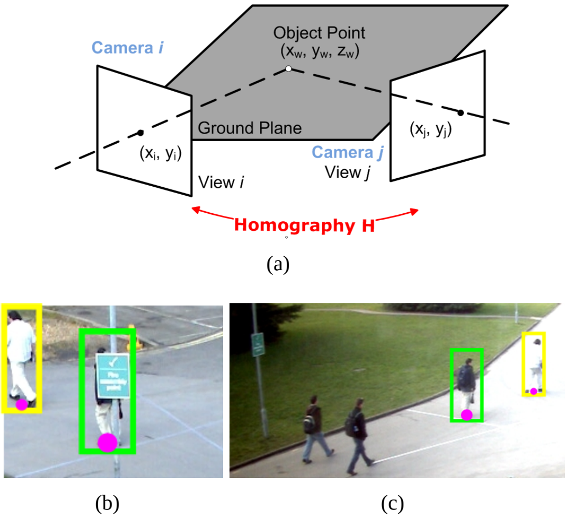
Negative samples: Negative updates are formed by concatenating the corresponding background images from the two views.

4. Experimental Results

We compare our proposed scene specific detectors to a state-of-the-art person detector and to a former proposed classifier grid method. In fact, the evaluation has only been carried out on this years PETS dataset, however the approach and results are valid for other object classes and scenes as well. Our approach is especially designed to perform better in difficult scenes with dynamical and static occlusions.

4.1. Implementation Details

We presented a very general framework for specific object detection. Principally, any background model, object detector, tracker or on-line classifier can be used. However, we used the following setting:



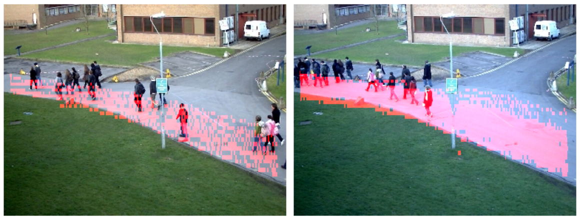
(a)

(b) (c)

Figure 5. 3D Context: A homography (a) maps a point on the ground plane from one view (b) to another (c).

Scene Geometry. The ground plane was manually estimated. Of course, the issue of ground plane estimation can also be addressed automatically, *e.g.* [2].

On-line Classifier. At each grid cell we compute a classifier using on-line boosting for feature selection [7]. Each classifier consists of 40 selectors containing a set of 30 weak classifiers. Haar-like features and histograms of oriented gradients are used as weak classifiers. The classifier grid has an overlap of 90 percent between single grid cells. Further, a classifier cell is only created if there is existence of a scene specific positive sample. This allows to save memory which can be used to increase the overlap or number of features. The grid is constantly being filled up with detection or tracking results (as shown in Fig. 6). The classifier grid is actually quite dense in interesting regions such as the street. As post-processing, we use a simple non-maxima suppression taking care of the relations between the cells. Note, non-maxima suppression is used to group very nearby classifier grid cells together, however it is only performing on a very tight local scale similar to the occupancy of a single person, in order to detect individual people in crowds and not “fuse” them into one detection.



(a) t_1 (b) $t_2 > t_1$

Figure 6. Temporal evolution of local classifier occupancy (local classifier existence is colored in red).

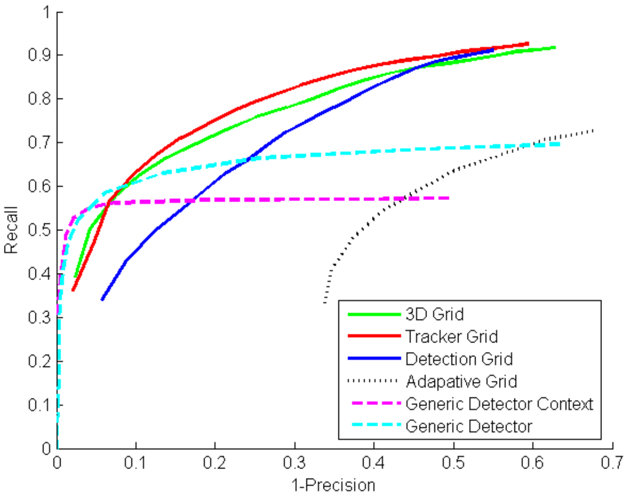


Figure 7. RPC for PETS 2009 dataset.

Method	R.	Pr.	f-M.
3D Grid	0.73	0.79	0.76
Tracker Grid	0.74	0.81	0.78
Detector Grid	0.72	0.71	0.72
Adaptive Grid [10]	0.59	0.55	0.57
Generic Detector Context [6]	0.56	0.93	0.70
Generic Detector [6]	0.63	0.87	0.73

Table 2. Results for PETS 2009 dataset at maximized f-Measure.

Background Model. As background model we use the approximated median method [17] updated every frame with pixel increment/decrement by 3 if no object is detected.

Object Detector. As an initial person detector we choose [6]. The resulting detection were filtered using a ground plane, *i.e.*, the detections must have an overlap of 20 percent in height with the grids.

Object Tracker. As tracker we used [9] with a one-shot learned prior which delivers robust tracking results even under partial occlusions.

4.2. Evaluation on PETS 2009 dataset

For a quantitative evaluation, we use recall-precision curves (RPCs) [1]. We manually labeled all the sequence frames, results are depicted in Fig. 7 and Tab. 2. Additionally, we show qualitative result in Fig. 8 on three typical scenes (with one zoomed) for the different approaches.

The generic detector shows accurate detection results of fully visible persons. However, in crowed scenes the recall is low compared to the proposed classifier grid approaches. There are no generic detections of partly occluded persons though, *e.g.* behind the lamp pole (static occlusion) or behind persons (dynamic occlusion). Especially, our approaches outperform the former adaptive grid approach [10], which indicates that a fixed set of positive samples can not be sufficient to detect persons in difficult crowded scenes. As expected, our approach using tracking performs better than using detections only. So temporal context introduced through tracking clearly helps in difficult scenes to gather occluded samples. There are only slight differences in the performance of the 2D tracker grid and the 3D grid. In general, 3D context information seems to help, however, the PETS dataset might be not long enough to fully explore it.

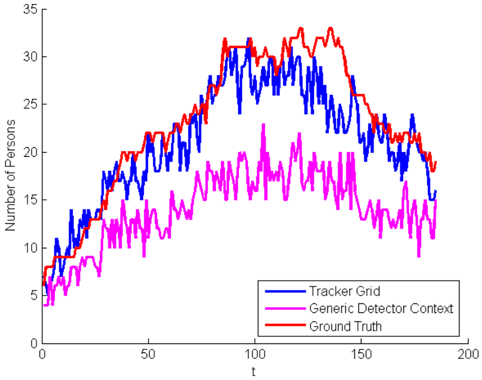


Figure 9. Person counts on PETS 2009 sequence *S1.L1* at maximized f-Measure.

To solve the person count task proposed by PETS we simply count the number of detections per frame at maximized f-Measure. Fig. 9 depicts the counted persons, showing improved performance compared to a generic detector using context as post-processing. However, please note that person detection is a more general and harder task than person count. In some cases, false detections compensate for missing detections. This fact is not reflected in the evaluation of person count. In this paper, no attempt has been made to improve the person count score.

4.3. Typical Updates

Fig. 10 shows typical samples collected in the local bags, reflecting the appearance of persons as well as the typical background at that position.

The task of the classifier is to select discriminant features to distinguish the classes. In contrast to the former gird based approach we benefit from a scene specific positive set as well as reduced label noise in the background class. Further, it can be noted that the positive class of the tracker grid contains samples with partially occluded persons. Hence, focus is implicitly put on relevant image regions as features will only be selected in regions which are different than the background, *i.e.*, no image feature will be placed on the common occluding object. The 3D grid patches consist of a bigger patch from the first viewpoint and a smaller extended patch from a second viewpoint. Mostly, there is more than one person appearing in the extended patch as the exact correspondences are not available.

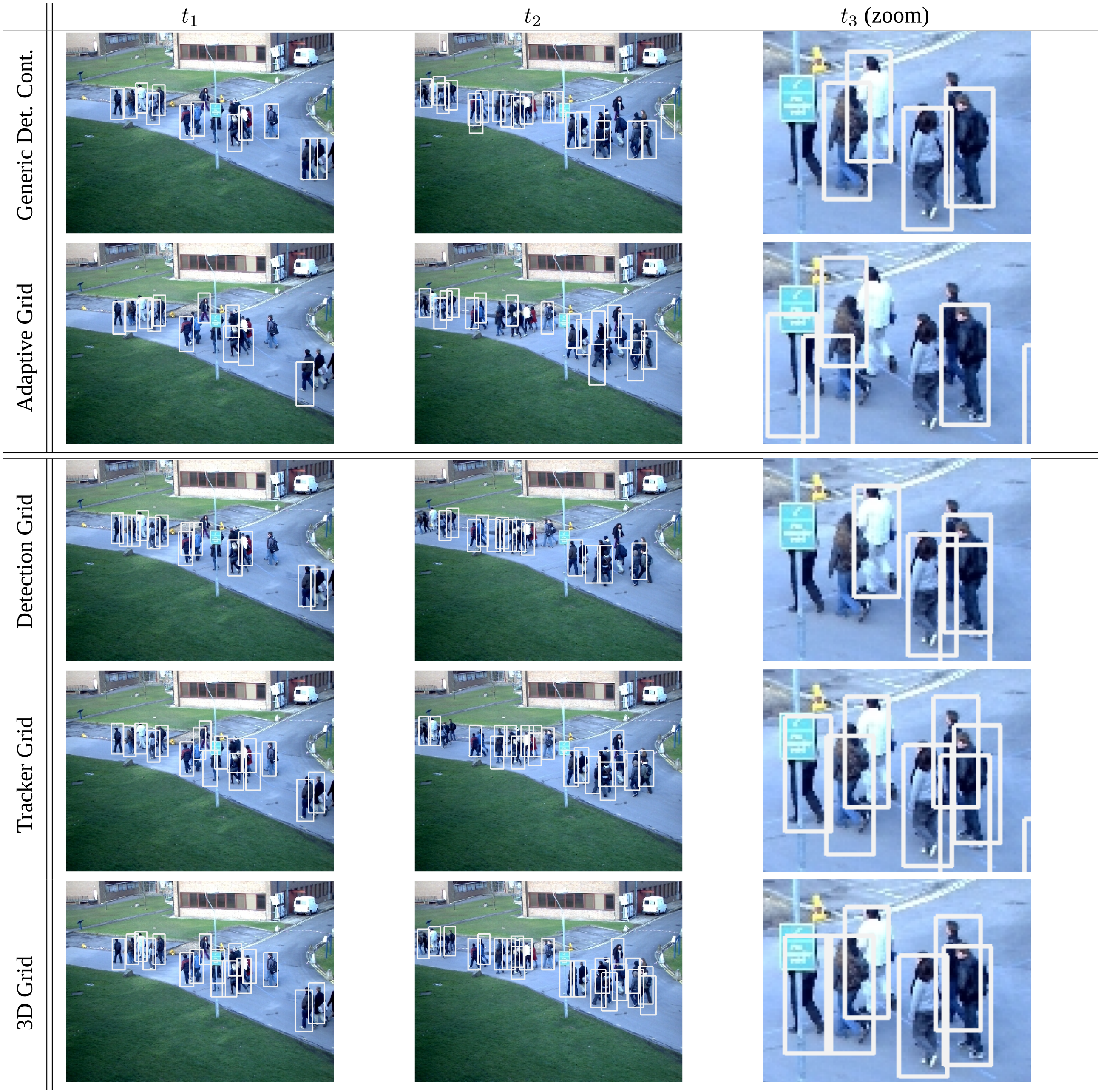


Figure 8. Typical results for PETS 2009 dataset. The approaches perform very differently in the region of the lamp pole (static occlusion) and in crowds (dynamic occlusions).

5. Conclusion

We explore context to robustly improve object detectors using unlabeled data. Compared to former approaches, the focus of the proposed approach was put on conservative gathering of scene-specific samples for both, specific objects and backgrounds. The experimental results evaluated on the PETS 2009 dataset have shown the necessity of including context to improve object detection, especially in the case of static occlusions and crowed scenes.

Acknowledgment The research leading to these results has received funding from the European Community's Seventh Framework Programme FP7/2007-2013 - under grant agreement no. 216465-SCOVIS.

References

[1] S. Agarwal, A. Awan, and D. Roth. Learning to detect objects in images via a sparse, part-based representation. *PAMI*, 26(11):1475–1490, 2004. 6

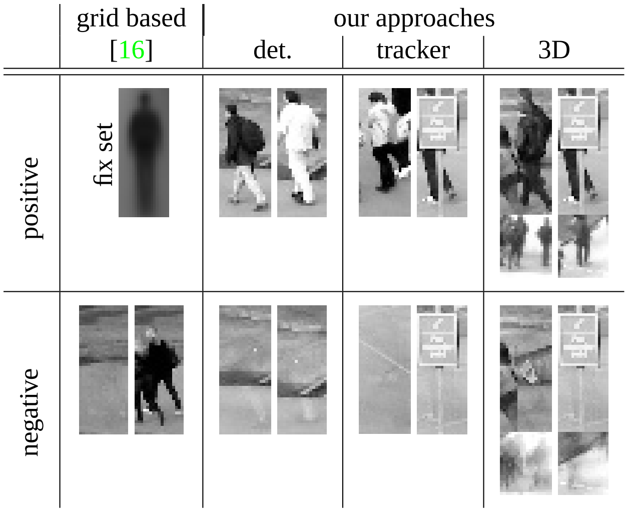


Figure 10. Typical collected samples for updating a grid classifiers.

[2] M. Breitenstein, E. Sommerlade, B. Leibe, L. van Gool, and I. Reid. Probabilistic parameter selection for learning scene structure from video. In *Proc. BMVC*, 2008. 5

[3] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proc. CVPR*, volume 1, pages 886–893, 2005. 1

[4] H. Dee and S. Velastin. How close are we to solving the problem of automated visual surveillance? *Machine Vision and Applications*, 19(5-6):329–343, 2008. 1

[5] S. Divvala, D. Hoiem, J. Hays, A. Efros, and M. Hebert. An empirical study of context in object detection. In *Proc. CVPR*, 2009. 2

[6] P. Felzenszwalb, D. McAllester, and D. Ramanan. A discriminatively trained, multiscale, deformable part model. In *Proc. CVPR*, 2008. 1, 4, 6

[7] H. Grabner and H. Bischof. On-line boosting and vision. In *Proc. CVPR*, volume 1, pages 260–267, 2006. 2, 4, 5

[8] H. Grabner, C. Leistner, and H. Bischof. Time dependent online boosting for robust backgroundmodeling. In *Proc. Int. Conf. on Comp. Vision Theory and App.*, 2007. 2, 3

[9] H. Grabner, C. Leistner, and H. Bischof. Semi-supervised on-line boosting for robust tracking. In *Proc. ECCV*, 2008. 2, 6

[10] H. Grabner, P. Roth, and H. Bischof. Is pedestrian detection realy a hard task? In *Proc. PETS*, 2007. 2, 3, 4, 6

[11] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004. 4

[12] G. Heitz and D. Koller. Learning spatial context: Using stuff to find things. In *Proc. ECCV*, 2008. 1

[13] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. In *Proc. CVPR*, volume 2, pages 2137–2144, June 2006. 1

[14] B. Leibe, E. Seemann, and B. Schiele. Pedestrian detection in crowded scenes. In *Proc. CVPR*, volume 1, pages 878–885, 2005. 1

[15] I. Matthews, T. Ishikawa, and S. Baker. The template update problem. *PAMI*, 26:810 – 815, 2004. 2

[16] P. Roth, S. Sternig, H. Grabner, and H. Bischof. Classifier grids for robust adaptive object detection. In *Proc. CVPR*, 2009. 2, 4, 8

[17] C. Stauffer and W. Grimson. Adaptive background mixture models for real-time tracking. In *Proc. CVPR*, number 2, pages 246–252, 1999. 6

[18] A. Torralba. Contextual priming for object detection. *IJCV*, 53(2):169–191, 2003. 1

[19] B. Wu and R. Nevatia. Improving part based object detection by unsupervised, online boosting. In *Proc. CVPR*, pages 1–8, 2007. 1, 4

[20] X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005. 2