Classifier Grids for Robust Adaptive Object Detection

Peter M. Roth[†], Sabine Sternig[†], Helmut Grabner[‡], Horst Bischof[†]

[†]Institute for Computer Graphics and Vision, Graz University of Technology, Austria
[‡]Computer Vision Laboratory, ETH-Zurich, Switzerland

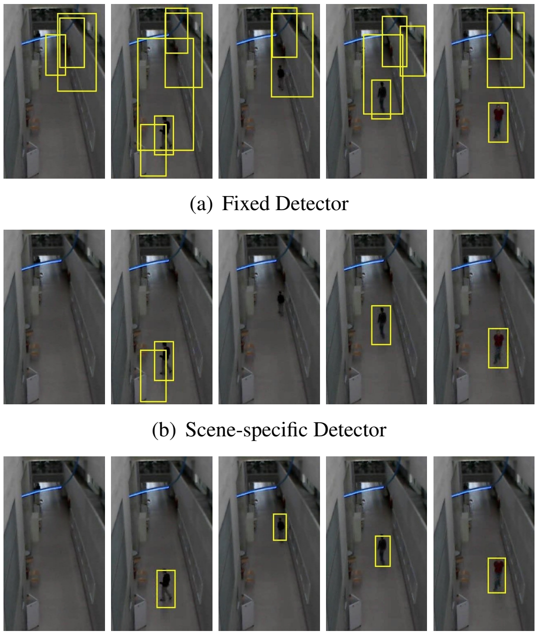
{pmroth,sternig,bischof}@icg.tugraz.at grabner@vision.ee.ethz.ch

Abstract

*In this paper we present an adaptive but robust object detector for static cameras by introducing classifier grids. Instead of using a sliding window for object detection we propose to train a separate classifier for each image location, obtaining a very specific object detector with a low false alarm rate. For each classifier corresponding to a grid element we estimate two generative representations in parallel, one describing the object's class and one describing the background. These are combined in order to obtain a discriminative model. To enable to adapt to changing environments these classifiers are learned on-line (*i.e.*, boosting). Continuously learning (24 hours a day, 7 days a week) requires a stable system. In our method this is ensured by a fixed object representation while updating only the representation of the background. We demonstrate the stability in a long-term experiment by running the system for a whole week, which shows a stable performance over time. In addition, we compare the proposed approach to state-of-the-art methods in the field of person and car detection. In both cases we obtain competitive results.*

1. Introduction

A very prominent approach for object detection is to use a sliding window technique (*e.g.*, [3,15,18,21]). Each patch of an image is tested if it is consistent with a previously estimated model or not. Finally, all consistent patches are reported. The model may either represent a specific object (*e.g.*, a specific cup, that is represented by different views) or a class of objects (*e.g.*, faces, pedestrians, cars, or bikes), where specific instances are not distinguished. These models are mostly based on local features described by a classifier, (*e.g.*, AdaBoost [5] or support vector machine [20]), which is typically obtained by learning. Hence, when discussing the problem of object detection, implicitly, the problem of visual learning is addressed.



(a) Fixed Detector

(b) Scene-specific Detector

(c) Adaptive Detector

Figure 1. Since changing environmental conditions (*e.g.*, lightning changes or changes of objects in the background) can not be handled by a fixed model an adaptive/scene specific system is required.

The goal of all of these approaches is to build a generic object model that is applicable for all possible scenarios and tasks (e.g., [3,4,11]). The drawback of these methods, however, is that the detectors are usually not very specific (*i.e.*, they return false alarms). As can be seen from Figure 1(a) even if general classifiers ("broad application") are trained from a very large number of training samples, they often fail for specific situations. This is caused by the main limitation of such approaches – a representative dataset is needed, which is not available for many applications. Since not all variability, especially for the negative class (*i.e.*, all possible backgrounds), can be captured this results in a low recall and an insufficient precision.

One way to overcome this problem is to use scene specific information to reduce the number of false alarms [9]. As can be seen from Figure 1(b) this can dramatically improve the overall performance of a generic detector. To further improve the classification results specific classifiers ("narrow applications") can be applied, which are designed to solve a specific task (*e.g.*, object detection for a specific setup). This is the common case, *e.g.*, for a surveillance application. In fact, to train such classifiers less training data is required and for the particular task they are usually better in terms of accuracy and efficiency [12, 18, 22].

To further improve the classification power and to further reduce the number of needed samples an adaptive classifier using an on-line learning algorithm can be applied [10, 15, 22]. Thus, the system can adapt to changing environments (*e.g.*, changing illumination conditions) and these variations need not to be handled by the model. In fact, in this way the complexity of the problem is reduced and a more efficient classifier can be trained. The main problem of adaptive methods is that they tent to drift when running over a long period of time.

In this paper, we address the problem of adaptive scene-specific learning by further simplifying the problem. In particular, we use the ideas of grid-based object classification (*e.g.*, [7, 8]), where the main idea is to apply a separate classifier on each image location (grid element), to learn an adaptive but still robust scene specific object representation. We estimate two models in parallel. We learn a generative model for the background corresponding to the grid element as well as a generative model for the object-of-interest. Thus, we can keep the representation for the object fixed while we gain the adaptivity by adapting the model to the background. Finally, both generative classifiers are combined in order to get a discriminative model. To adapt the classifier to a specific scene an on-line learning method is applied for learning. In a long-term experiment we demonstrate that the classification performance is not decreased even if the system is running and learning for a whole week (*i.e.*, by taking one image per second, we processed approx. 580,000 frames). In addition, we demonstrate our method for person and car detection comparing it to state-of-the-art approaches obtaining excellent results. However, since the approach is quite general, it is not limited to these specific applications.

The paper is organized as follows. In Section 2 we review the basic concepts of on-line learning using classifier grids. Next, in Section 3 we present and discuss our new combined generative-discriminative grid-based classification method. Detailed evaluations of the proposed approach are given in Section 4. Finally, the paper is summarized and discussed in Section 5.

2. On-line Learning Classifier Grids

In the following, we briefly review the main ideas of classifier grids and how they can be applied for on-line learning using fixed update rules.

2.1. Classifier Grid

The idea of classifier grids is to sample an input image by using a fixed highly overlapping grid (both in location and scale), where each grid element $i = 1, \ldots, N$ corresponds to one classifier C_i. This is illustrated in Figure 2. Thus, the classification task that has to be handled by one classifier C_i is reduced to discriminate the background of the specific grid element from the object-of-interest. Due to this simplification less complex classifiers can be applied. In particular, the grid-based representation is well suited for compact smart on-line classifier, which can be evaluated and updated very efficiently.

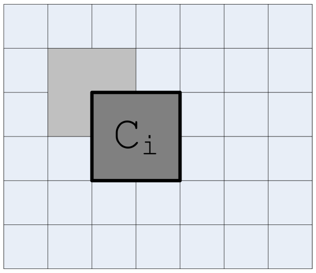


Figure 2. Concept of grid-based classification: a highly overlapping grid is placed over the image, where each grid element corresponds to a single classifier.

2.2. Fixed Update Rules

However, on-line systems have one main disadvantage: new unlabeled data has to be robustly included into an already built model. Typical update schemes (i.e., label generators) such as self-training (e.g., [13, 17]) and co-training (e.g., [2, 12]) rely on a direct feedback of the current classifier. Thus, they tend to drift, *i.e.*, the classifier starts to learn something completely wrong and would yield arbitrarily wrong results.

To overcome these problems, we recently proposed a grid-based object detection system that is based on fixed updates [7]. Given a set of representative positive (hand) labeled examples $\mathcal{X}^+$. Then, using

$$\langle \mathbf{x}, +1 \rangle, \qquad \mathbf{x} \in \mathcal{X}^+ \tag{1}$$

to update the classifier is a correct positive update by definition. The probability that an object is present on patch $\mathbf{x}_i$ is given by

$$P(\mathbf{x}_i = \text{object}) = \frac{\#p_i}{\Delta t} \ , \tag{2}$$

where $\#p_i$ is the number of objects entirely present in a particular patch within the time interval Δt. Thus, the negative update with the current patch

$$\langle \mathbf{x}_{i,t}, -1 \rangle \tag{3}$$

is correct most of the time (wrong with probability $P(\mathbf{x}_i = \mathrm{object})$). The probability of a wrong update for this particular image patch is indeed very low.

Since the positive updates are correct by definition the remaining problem is some low amount of label noise from the negative class. Hence, the applied on-line learning method must be able to cope with this problem.

2.3. On-line Learning

This can be ensured by using on-line boosting for feature selection [6]. In general, Boosting [19] forms a strong classifier

$$H(\mathbf{x}) = \sum_{j=1}^{N} \alpha_j h_j(\mathbf{x}) \tag{4}$$

by a linear combination of N weak classifiers $h_j(\mathbf{x})$, which have only to perform better than random guessing. These weak classifiers are trained by re-weighing the training samples, *i.e.*, more emphasis is given to still misclassified examples. In order to do feature selection, each weak classifier h_j corresponds to one feature f_j.

For on-line boosting for feature selections selectors were introduced, where the actual boosting step is then performed on these selectors. Each selector j holds a holds a set of M weak classifiers $\{h_1^j, ..., h_M^j\}$ and is represented by its best weak classifier h_m^j, *i.e.*, the weak classifier with the lowest estimated error

$$\epsilon_{m,j} = \frac{\lambda_{m,j}^w}{\lambda_{m,j}^c + \lambda_{t,j}^w} \ , \tag{5}$$

where $\lambda_{m,j}^w$ and $\lambda_{m,j}^c$ are the weights of the samples that were classified correctly and incorrectly up to now.

In our case the weak classifiers h_j are built based on two distributions D_j^+ and D_j^-, which are estimated from the feature responses of negative and positive samples, respectively. In particular, we assume that these are Gaussian distributions, which can easily be updated (*e.g.* by using a Kalman filtering technique). Based on these a simple decision stump

$$h_j(\mathbf{x}) = p_j \cdot \mathrm{sign}(f_j(\mathbf{x}) - \theta_j) \tag{6}$$

can be estimated, where the threshold θ_j and parity p_j are calculated using Bayesian rule with respect to D_j^+ and D_j^-.

3. Robust Adaptive Grid-based Classification

When training an on-line classifier we have to robustly include new unlabeled samples into the system. More formally, at time t given a classifier C_{t-1} and an unlabeled example $\mathbf{x}_t$, the goal is to robustly estimate a label $y_t \in \{+1, -1\}$ for $\mathbf{x}_t$. According to the findings of Section 2.2 we can take advantage of classifier grids in both cases, positive and negative updates.

Since the appearance of the object-of-interest is known, it can be described in advance by a finite set $\mathcal{X}^+$ of positive samples. Thus, for each feature $f_j \in \mathcal{F}$, where $\mathcal{F}$ is the full feature space, a generative model can be estimated using the distributions D_j^+. Since $\mathcal{X}^+$ is fixed, the distributions D_j^+ can also be kept fixed and can be calculated in an off-line pre-training stage. In contrast, the negative class is changing over time and can therefore not be described by a finite set of samples. However, this information can be extracted on-line from the scene, *i.e.*, from the patch corresponding to the grid-element. According to Eq. (2) the probability that a patch related to a grid-element does not describe the background is quite low and a representative set of negative samples can be gathered. Thus, for each feature f_j we can estimate a distributions D_j^- on-line over time.

The thus obtained generative models describe the positive and negative samples by the distributions of the feature responses. Hence, we finally can estimate a discriminative classifier by combining them on feature level. This can efficiently be realized by using on-line boosting for feature selection (see Section 2.3). The overall idea is illustrated in Figure 3.

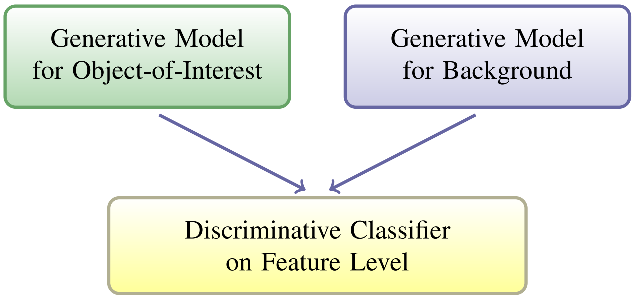

Generative Model for Object-of-Interest
Generative Model for Background
Discriminative Classifier on Feature Level

Figure 3. The grid-based detector can be interpreted as a combination of generative models describing the background and the object-of-interest, which are combined to a discriminative model at feature level.

In the following we summarize the three steps that are required to estimate an adaptive discriminative model by combining the generative models described above:

Off-line pre-training: Given the fixed set of positive training samples $\mathcal{X}^+$, in the first step we train a classifier applying off-line boosting. In this way for the selected features f_j we can estimate the corresponding positive distributions D_j^+ as well as positive error $\epsilon_+^{\text{off-line}}$, which is kept fixed during the on-line learning.

On-line weak classifier update: The negative distribution D_j^- is updated using the current patch whereas the fixed positive distribution D_j^+ is unchanged. In order to reduce the amount of label noise, the generative information of the positive and negative distribution D_j^+ and D_j^- can be used to decide whether the current image patch should be used for a negative update or not. Based on these two distributions we can build a discriminative model for each weak classifier corresponding to a feature (see Figure 4 and compare to Eq. 6). Since the generative representations can be adapted on-line this allows for discriminative on-line learning.

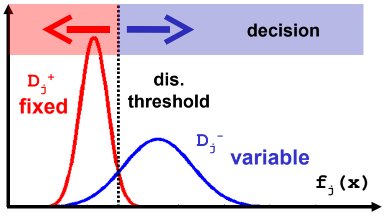


Figure 4. For each feature f_j the discriminative threshold θ is calculated using the *fixed* (pre-trained) positive distribution D_j^+ and the *variable* negative distribution D_j^-.

On-line feature selection: For the feature selection process, the errors of the features (weak classifiers) have to be calculated. In particular, as described in Section 2.3, the best weak classifier in a selector is chosen according to its error. Since only negative updates are performed during on-line learning, only the error for negative samples, *i.e.*, the false positive rate, can be estimated. However, the fixed distributions D_j^+ were estimated in the pre-training stage. Thus, instead of Eq. (5) we can use the combined error

$$\epsilon = \frac{1}{2}(\epsilon_+^{\text{off-line}} + \epsilon_-^{\text{on-line}}) \tag{7}$$

to select the best weak classifier within the selector.

Finally, the thus trained classifiers are evaluated on each new frame, where a detection is reported, if the classifier's response $H(\mathbf{x})$ is above a certain threshold (*e.g.*, zero). In order to avoid overlapping detections a post-processing is applied. Since there is no direct feedback within the learning process and possible errors are not accumulated but are fading out, the system runs stable even for a long period of time.

In general, any feature type for which the required distributions can be estimated (*e.g.*, [3]) may be applied within this framework. However, to have an efficient system (memory requirements and speed) that can be applied for real-world scenarios, we use only Haar-like wavelets [21]. In fact, for the scenarios discussed in Section 4 we can ensure a frame-rate of up to $30 fps$ on a standard PC! Moreover, these results show that even using this very simple feature type competitive results can be obtained.

4. Experimental Results

In the following we will demonstrate the benefits of the presented approach. Therefore, we split the experiments into three parts. First, we give a detailed analysis for the task of person detection. Second, to show that the approach is not limited to this specific task, it is applied for a completely different application, *i.e.*, car detection. Finally, to show the stability over time, we give results for a long-term experiment, which ran for one week!

4.1. Experimental Setup

If not specified otherwise all experiments were performed and evaluated as described below. First, to generate the classifier grid the approximate size of the object-of-interest in the scene is needed. For reasons of simplicity we estimated the ground-plane for our experiments manually (this could also be done automatically, *e.g.*, [16]). Based on this estimate a grid of classifiers is initialized using an overlap of $85\% - 90\%$. Each of these classifiers, which are evaluated and updated whenever a new frame arises, consists of 50 selectors, each of them holding a set of 10 weak classifiers. Hence, in total only 500 weak classifiers, each of it corresponding to one Haar-like feature, have to be stored. This is the result of the off-line pre-training stage, which allows to pre-select a smaller number of valuable features.

For a quantitative evaluation, we use recall-precision curves (RPC) [1]. Therefore, we have to estimate the precision $Pr = TP/(TP + FP)$ and the recall $R = TP/P$, where TP is the number of true positives, FP is the number of false positives, and P is the number of positives in the test data represented by the given ground-truth. In particular, a detection is accepted as true positive, if it fulfills the overlap criterion [1], where a minimal overlap of 50% is needed. Once we have estimated these parameters we can plot the recall R against $1 - Pr$. Additionally, we use the $F - measure$ [1], which is the harmonic mean between $recall$ and $precision$ and is defined by $FM = (2 \cdot R \cdot Pr)/(R + Pr)$. In particular, the characteristics given in this section were generated such that the $F - measure$ was maximized.

4.2. Person Detection

First of all, we give a detailed evaluation of the proposed approach for the task of person detection. For that purpose, we generated a challenging test set showing a corridor of a public building (*Corridor sequence*) consisting of 300 frames, which contains 296 persons. The sequence, which was taken at a resolution of 320 × 240, shows typical difficulties such as various moving objects (*e.g.*, a ball, chairs, and an umbrella) and partly overlapping persons.

In particular, we compared our new method to generic state-of-the-art detectors, which can be downloaded from the Internet[1] [2] (*i.e.*, the Dalal and Triggs (D&T) person detector [3] and the person detector trained using the deformable part model of Felzenszwalb *et al.* (FS) [4]), which do not use any prior knowledge, to adaptive scene specific detectors (*i.e.* the grid-based approach (GB) of Grabner *et al.* [7] and the Conservative Learning (CL) of Roth *et al.* [18], as well as to low level methods (*i.e.* a simple background model (BGM) [14], template matching (TM), and a combination of both (TM+BGM), which might be considered a simple pendent to our method).

In order to allow for a fair comparison, in a post-processing step we removed all detections that do not fit to the estimated ground-plane. In fact, a detection was removed if the scale was smaller than 75% or greater than 125% of the expected patch-height. Please note, this post-processing does not reduce the recall since these detections would be counted as false positives otherwise. Moreover, to ensure satisfactory results for the Dalal and Triggs detector and the deformable part model, we resized the input images to 640 × 480 for these methods.

The thus obtained results obtained by a given ground-truth are summarized in Figure 5 and in Table 1, where we show the recall-precision curves for all methods and the corresponding detection characteristics.

	R	Pr	FM
Cons. Learning [18]	0.84	0.94	0.89
Proposed approach	0.88	0.83	0.85
Dalal and Triggs [3]	0.63	0.97	0.76
Grabner *et al.* [7]	0.61	0.87	0.72
Felzenszwalb *et al.* [4]	0.61	0.83	0.70
BGM	0.76	0.63	0.69
TM+BGM	0.44	0.73	0.55
TM	0.44	0.37	0.40

Table 1. Detection characteristics of the *Corridor Sequence* for different methods sorted by the F-measure.

From these results, it can be seen that the general detectors show an excellent precision, but the recall is too low for practical applications. Moreover, the low level cues totally fail, *i.e.* for BM and TM either the recall or the precision is very poor such that even a combination of both (TM+BGM) yields insufficient results. In contrast, the adaptive methods provide sufficient results. In fact, the grid-based approach of Grabner *et al.* shows a comparable performance to the generic detectors. The best results are obtained by Conservative Learning, a high sophisticated method. However, our proposed approach yields competitive detection results, even using only very simple update rules and compact classifiers. Finally, typical results of our approach are depicted in Figure 6.

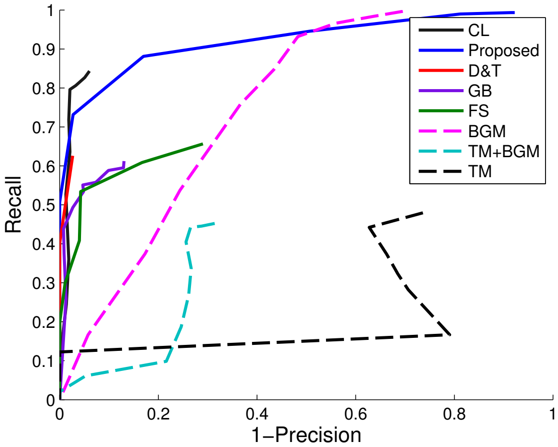


Figure 5. RPC for the *Corridor Sequence*.

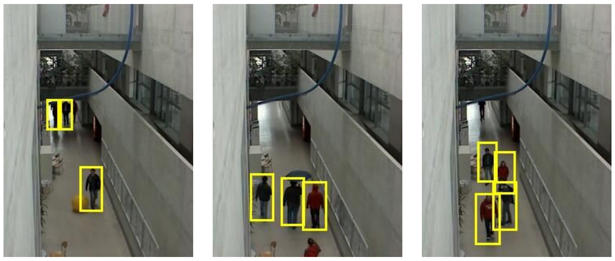


Figure 6. Illustrative detection results of the grid-based person detector for the *Corridor Sequence*.

In addition, we evaluated our approach on two publicly available datasets, *i.e.*, the *Caviar* dataset [3] and the *PETS 2006* dataset[4] and compared it to the general detectors, which might be considered a fair baseline. In particular, from both data sets we selected one sequence containing a lot of people. The results for both data sets are shown in Figure 7 and Figure 9, respectively. Again it can be seen that the adaptive grid-based detector outperforms the general detector; especially, in terms of recall. Finally, we show some illustrative detection results in Figure 8 and Figure 10, respectively.

[1] http://pascal.inrialpes.fr/soft/olt

[2] http://people.cs.uchicago.edu/˜pff/latent

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

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

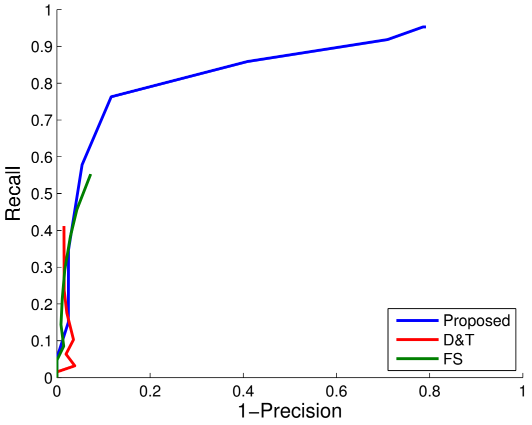


Figure 7. RPC for the *Caviar Sequence*.

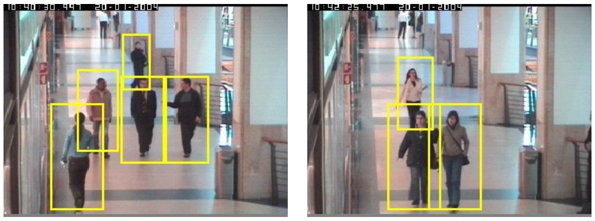


Figure 8. Illustrative detection results of the grid-based person detector for the *Caviar Sequence*.

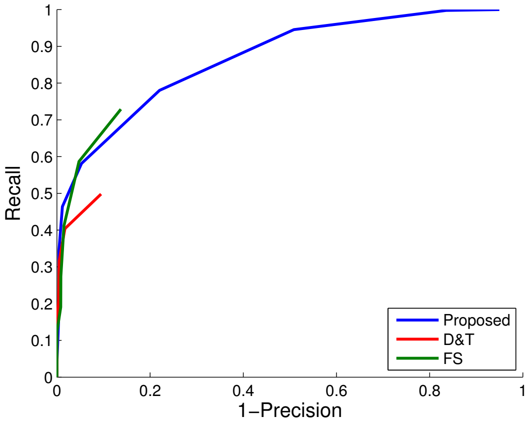


Figure 9. RPC for the *PETS 2006 Sequence*.

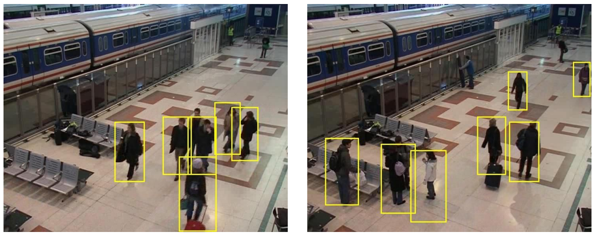


Figure 10. Illustrative detection results of the grid-based person detector for the *PETS 2006 Sequence*.

4.3. Car Detection

To show that the proposed approach is not limited to detecting persons, we additionally demonstrate it for car detection. We compare our method to existing established methods: Implicit Shape Models (ISM) of Leibe *et al*. [11][5] and a car detector trained using the deformable part model of Felzenszwalb *et al*. (FS) [4]. The methods were evaluated on a sequence showing one lane of a public highway. The whole scene consists of 1000 frames and contains 1283 cars from the rear view. Again for the ISM method and the FS detector the original images (380×324) were re-sized to the double size. In order to obtain a sufficient number of detections from the FS detector the detection threshold was set to -0.5.

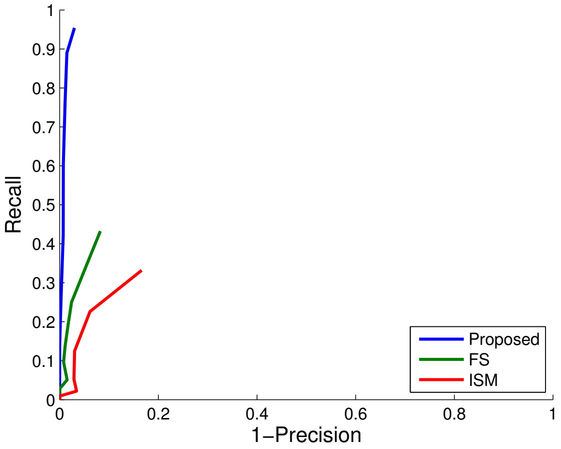


Figure 11. RPC for the *Highway sequence*.

From the results shown in Figure 11 it can be seen that the proposed method clearly outperforms the fixed car detectors. In particular, we get a recall of more than 90% while still having a very high precision! Moreover, illustrative detection results obtained by the proposed approach are shown in Figure 12.

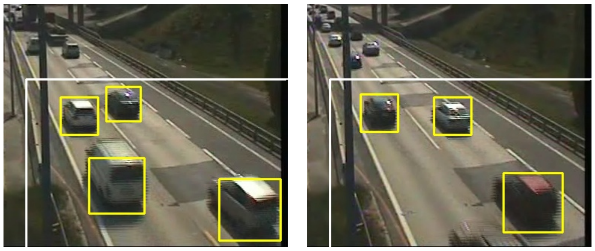


Figure 12. Illustrative detection results of the grid-based person detector for the *Highway sequence*. The white line put a ceiling on the detection region.

[5] http://www.vision.ee.ethz.ch/~bleibe/code/ism.html

4.4. Long-term behavior

Since the main goal in this paper was to develop a robust adaptive system that is learning 24 hours a day and 7 days a week, in the following we demonstrate the long-term behavior of the proposed method. During 7 days we updated our system with $580,000$ frames (*i.e.*, we took $1fps$). To show that the systems' performance is unchanged over time, we selected four different points in time and extracted sequences of $2,500$ frames (which corresponds to approx. 40 minutes of video data):

	# updates yet performed	# persons
1st day	3,390	201
3rd day	179,412	222
6th day	484,891	454
7th day	577,500	316

Table 2. Description of the selected sequences of the long-term experiment.

From the results shown in Figure 13 and in Table 3 it can be seen that the method is stable over time. The slightly variations in the curves can be explained by the different levels of complexity for the four sequences (*i.e.*, number of persons, density of persons, etc.). But as can be seen from Table 3 the F-measure is unchanged over time.

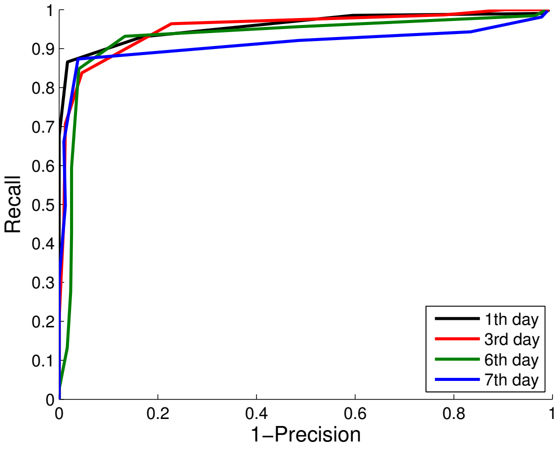
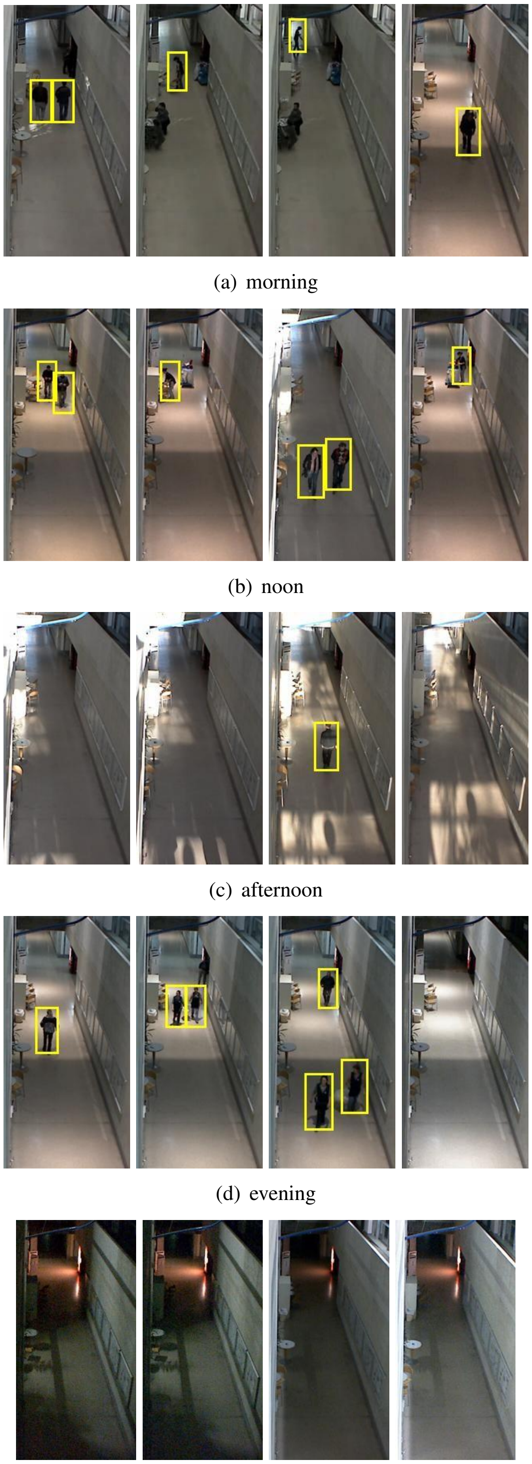


Figure 13. RPC for the *long-term experiment*

	R	Pr	FM
1st day	0.87	0.98	0.92
3rd day	0.84	0.95	0.89
6th day	0.85	0.96	0.90
7th day	0.87	0.96	0.92

Table 3. Results of the *long-term experiment*.

Finally, in Figure 14 we illustrate the significantly changing conditions we had to deal with during these 7 days (*i.e.*, natural light, artificial lighting, inadequate lighting, etc.). Thus, these drastically changing conditions, which can be handled by our system considerable better than by other methods, arise the need for an adaptive system!



(a) morning

(b) noon

(c) afternoon

(d) evening

(e) night

Figure 14. Illustrative detection results of the grid-based person detector obtained during to long-term experiment.

5. Conclusion

In this paper we presented a new robust adaptive object detection system for static cameras, which should run 24 hours a day, 7 days a week. The main idea is to apply classifier grids, *i.e.*, we train a separate classifier for each image location. Since the complexity of the detection task is considerably reduced (*i.e.*, a single classifier has only to discriminate between the object-of-interest and the background of the specific grid element) more compact and thus very efficient classifiers can be applied. To ensure an adaptive but still stable system, we apply two representations (on feature level) in parallel, one for the background, which is adapted, and one for the objects, which is kept fixed. In particular, this is realized by using on-line boosting for feature selection. Thus, the system can adapt to changing environment conditions avoiding drifting (*i.e.*, corrupting the classifier). This stability over time is demonstrated in a long-term experiment. In fact, the results show that even after performing 580,000 updates (!) our system is still running stable. Moreover, a comparative study for person and car detection shows that the proposed approach yields competitive results and even outperforms state-of-the-art methods on different (publicly available) datasets.

Acknowledgment

This work was supported by the FFG project AUTOVISTA (813395) under the FIT-IT programme, by the FFG project HIMONI under the COMET programme in co-operation with FTW, and by the Austrian Joint Research Project Cognitive Vision under projects S9103-N04 and S9104-N04. In addition, this work was supported by the EU-project SCOVIS under grant agreement no 216465.

References

[1] 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.

[2] A. Blum and T. Mitchell. Combining labeled and unlabeled data with co-training. In *Proc. Conf. on Computational Learning Theory*, pages 92–100, 1998.

[3] 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.

[4] 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.

[5] Y. Freund and R. E. Shapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55:119–139, 1997.

[6] H. Grabner and H. Bischof. On-line boosting and vision. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume I, pages 260–267, 2006.

[7] H. Grabner, P. M. Roth, and H. Bischof. Is pedestrian detection really a hard task? In *Proc. IEEE Intern. Workshop on Performance Evaluation of Tracking and Surveillance*, pages 1–8, 2007.

[8] M. Heikkilä, M. Pietikäinen, and J. Heikkilä. A texture-based method for detecting moving objects. In *Proc. British Machine Vision Conf.*, pages 187–196, 2004.

[9] D. Hoiem, A. A. Efros, and M. Hebert. Putting objects in perspective. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, volume II, pages 2137–2144, 2006.

[10] 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.

[11] B. Leibe, A. Leonardis, and B. Schiele. Robust object detection with interleaved categorization and segmentation. *Intern. Journal of Computer Vision*, (1–3):259–289, 2008.

[12] 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.

[13] L.-J. Li, G. Wang, and L. Fei-Fei. Optimol: automatic online picture collection via incremental model learning. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[14] N. J. B. McFarlane and C. P. Schofield. Segmentation and tracking of piglets. *Machine Vision and Applications*, 8(3):187–193, 1995.

[15] 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.

[16] R. Pflugfelder and H. Bischof. Online auto-calibration in man-made worlds. In *Digital Image Computing: Technqiues and Applications*, pages 519 – 526, 2005.

[17] C. Rosenberg, M. Hebert, and H. Schneiderman. Semi-supervised self-training of object detection models. In *IEEE Workshop on Applications of Computer Vision*, pages 29–36, 2005.

[18] 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.

[19] R. E. Schapire. The strength of weak learnability. *Machine Learning*, 5(2):197–227, 1990.

[20] V. N. Vapnik. *The Nature of Statistical Learning Theory*. Springer, 1995.

[21] P. Viola, M. J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *Proc. IEEE Intern. Conf. on Computer Vision*, volume II, pages 734–741, 2003.

[22] B. Wu and R. Nevatia. Improving part based object detection by unsupervised, online boosting. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, pages 1–8, 2007.