Spatiotemporal Stacked Sequential Learning for Pedestrian Detection

Alejandro González[1,2], David Vázquez[1,2], Sebastian Ramos[2], Antonio M. López[1,2], and Jaume Amores[3]

[1] Autonomus University of Barcelona, Barcelona, Spain
[2] Computer Vision Center, Barcelona, Spain
[3] United Technologies Research Center, Cork, Ireland
{agalzate,dvazquez,sramosp,antonio}@cvc.uab.es
AmoresJ@utrc.utc.com

Abstract. Pedestrian classifiers decide which image windows contain a pedestrian. In practice, such classifiers provide a relatively high response at neighbor windows overlapping a pedestrian, while the responses around potential false positives are expected to be lower. An analogous reasoning applies for image sequences. If there is a pedestrian located within a frame, the same pedestrian is expected to appear close to the same location in neighbor frames. Therefore, such a location has chances of receiving high classification scores during several frames, while false positives are expected to be more spurious. In this paper we propose to exploit such correlations for improving the accuracy of base pedestrian classifiers. In particular, we propose to use two-stage classifiers which not only rely on the image descriptors required by the base classifiers but also on the response of such base classifiers in a given spatiotemporal neighborhood. More specifically, we train pedestrian classifiers using a stacked sequential learning (SSL) paradigm. We use a new pedestrian dataset we have acquired from a car to evaluate our proposal at different frame rates. We also test on well known dataset, Caltech. The obtained results show that our SSL proposal boosts detection accuracy significantly with a minimal impact on the computational cost. Interestingly, SSL improves more the accuracy at the most dangerous situations, *i.e.* when a pedestrian is close to the camera.

1 Introduction

Localizing humans in images is key for applications such as video surveillance, avoiding pedestrian-to-vehicle collisions, etc. Developing a reliable vision-based pedestrian detector is a very challenging task with more than a decade of history by now. As a result, a plethora of features, models, and learning algorithms, have been proposed to develop the pedestrian classifiers which are at the core of pedestrian detectors [10]. The research for boosting the accuracy of pedestrian classifiers has followed different lines. Some authors have researched image descriptors well-suited for pedestrians (*e.g.*, HOG [4], HOG+LBP [21], Haar+EOH [11], others have researched different image modalities (*e.g.*, appearance+depth+motion [22], [8]), others have focused on the pedestrian model (*e.g.*, DPM [9], [17], [14]. The outcome of each of the above mentioned proposals is a pedestrian classifier, termed here as *base classifier*, which provides a relatively high response at neighbor windows overlapping a pedestrian. In fact, non-maximum suppression (NMS) is usually performed as last detection stage in order to merge overlapped detections to a single one. An analogous reasoning applies for image sequences.

If there is a pedestrian located within a frame, the same pedestrian is expected to appear close to the same location in neighbor frames, while false positives are expected to be more spurious. In fact, this may allow removing such undesired spurious by the use of a tracker. In this paper we propose to exploit such expected *response correlations* for improving the accuracy of the classification stage itself. We propose to use a two-stage classification strategy which not only rely on the image descriptors required by the base classifiers, but also on the response of the own base classifiers in a given spatiotemporal neighborhood. More specifically, we train pedestrian classifiers using a stacked sequential learning (SSL) paradigm [2].

Temporal SSL involves the analysis of window volumes, these volumes may change depending on the application, the target conditions and camera movement. In this paper, we are specially interested in on-board pedestrian detection within urban scenarios. Therefore, camera and targets are in movement. Accordingly, in this paper we test our SSL approach for a fixed neighborhood (*i.e.*, fixed spatial window coordinates across frames) and for an scheme relying on an ego-motion compensation approximation (*i.e.*, varying spatial window coordinates across frames). Moreover, in order to assess the dependency of the results with respect to the frame rate, we acquired our own pedestrian dataset at 30fps by normal driving in an urban scenario. This new dataset is used as main guide for our experiments, but we also complement our study with other challenging dataset publicly available, Caltech. In this paper we start by using a competitive baseline in pedestrian detection [7], namely a holistic base classifier based on HOG+LBP features and linear SVM. Note that HOG+LBP/linear-SVM is base of different more sophisticated detectors: deformable models (DPM) [9], occlusion handling [21], [15], node experts in random forest [16] and domain adaptation [18]. Altogether, we think that HOG+LBP/linear-SVM is a proper baseline to start assessing our proposal. Moreover we have extended this baseline with the HOF [20] motion descriptor that complements the appearance and texture features of the baseline. Overall, the obtained results show that our spatiotemporal SSL proposal boosts detection accuracy significantly. Especially, when the pedestrians are close to the camera, *i.e.* in the most critical situations. Therefore, encouraging to augment the study for other pedestrian base classifiers as well as other object categories.

The rest of the paper is organized as follows. In Sect. 2 we review some works related to our proposal. Section 3 briefly introduces the SSL. In Sect. 4 we develop our proposal. Section 5 presents the experiments carried out to assess our spatiotemporal SSL, and discuss the obtained results. Finally, Sect. 6 draws our main conclusions.

2 Related work

The use of motion patterns as image descriptors was already proposed as an extension of spatial Haar-like filters for video surveillance applications [19], [3], [12] and for detecting human visual events [13]. In these cases, original spatial Haar-like filters were extended with a temporal dimension. Popular HOG descriptor was also extended using optical flow [5], [20]. In all cases motion information was complemented with appearance information (*i.e.*, Haar/HOG for luminance and/or color channels). In contrast with these approaches, our proposal does not involve to compute new temporal image descriptors as new features for the classification process. As we will see, we use the responses of a given base classifier in neighbor frames as new features for our

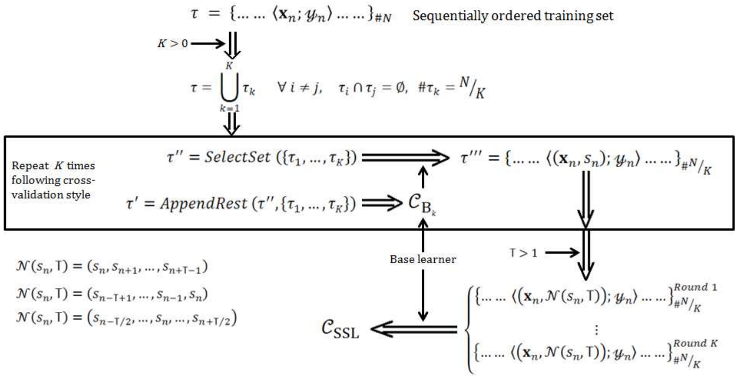

Fig. 1. SSL learning. See main text in Sect. 3 for details.

SSL classifier. In fact, our proposal can also be applied to base classifiers that already incorporate motion features. Therefore, the reviewed literature and our proposal are complementary strategies. Focusing on single frames, it has been recently shown how pedestrian detection accuracy can be boosted by analyzing the image area surrounding potential pedestrian detections. In particular, [6], [1] follow an iterative process that uses contextual features of several orders for progressively enhancing the response for true pedestrians and lowering it for hallucinatory ones. Our SSL proposal does not require new image descriptors and is not iterative, which makes it inherently faster. Moreover, we treat equally spatial and temporal response correlations.

Finally, we would like to clarify that our SSL proposal is not a substitute for NMS and tracking post-classification stages. What we expect is to allow these stages to produce more accurate results by increasing the accuracy of the classification stage.

3 Stacked sequential learning (SSL)

Stacked sequential learning (SSL) was introduced by Cohen *et al.* [2] with the aim of improving base classifiers when the data to be processed has some sort of sequential order. In particular, given a data sample to be classified, the core intuition is to consider not only the features describing the sample but also the response of the base classifier in its neighbor samples. Figure 1 summarizes the SSL learning process that we explain in more detail in the rest of this section.

Let τ be an ordered training sequence of cardinality N. The SSL approach involves to select a sub-sequence for training a base classifier, $\mathscr{C}_B$, and the rest to apply $\mathscr{C}_B$ and so training the SSL classifier, $\mathscr{C}_{SSL}$. If this is done once, then the final classifier $\mathscr{C}_{SSL}$ would be trained with less than N samples. Thus, to avoid this, it is followed a cross-validation style were τ is divided in $K > 0$ disjoint sub-sequences, $\tau = \cup_{k=1}^{K} \tau_k \wedge i \neq j \Rightarrow \tau_i \cap \tau_j = \emptyset$, and K rounds are performed by using a different subset each round to test the $\mathscr{C}_{B_k}$ and the rest of subsets for training this $\mathscr{C}_{B_k}$. At the end of the process, joining the K sub-sequences processed by the corresponding $\mathscr{C}_{B_k}$, we can have N *augmented* training samples for learning $\mathscr{C}_{SSL}$. $k = 1$ means to train the C_B and C_{SSL} on the same training set, without actually doing partitions.

Let us explain what means *augmented* training samples. The elements of τ, *i.e.*, the initial training samples, are of the form $< \mathbf{x_n}; y_n >$, where $\mathbf{x_n}$ is a vector of features with

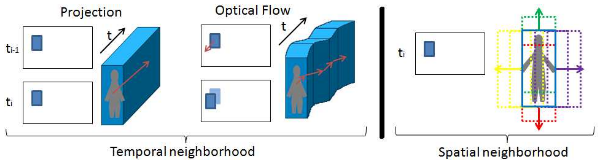

Fig. 2. Different types of neighborhood for SSL. See main text in Sect. 4.1 for details.

label y_n. Therefore, the elements of each sub-sequence τ_k are of the same form. As we have mentioned before, during each round k of the process, a sub-sequence τ'' is selected among $\{\tau_1, \ldots, \tau_K\}$, while the rest are appended together to form a sub-sequence τ'. From τ' it is learned $\mathscr{C}_{B_k}$ and applied to τ'' to obtain a new τ'''. The elements of τ''' are of the form $< (\mathbf{x_n}, s_n); y_n >$, where we have augmented the feature $\mathbf{x_n}$ with the classifier score $s_n = \mathscr{C}_{B_k}(\mathbf{x_n})$. Therefore, after the K rounds, we have a training set of N samples of the form $< (\mathbf{x_n}, s_n); y_n >$. It is at this point when we can introduce the concept of neighbor scores into the learning process. In particular, the final training samples are of the form $< (\mathbf{x_n}, \mathscr{N}(s_n, T)); y_n >$, where $\mathscr{N}(s_n, T)$ denotes a neighborhood of size $T > 1$ anchored to the sample n.

4 SSL for pedestrian detection

In this section, without loosing generality, we will assume the use of the *past neighborhood* (Sect. 3) to illustrate and explain our SSL approach (use previous images to do detection in the current one). Actually there is no need to save the previous images, saving the already computed scores is enough to compute the current SSL descriptor making the computation of SSL very computational efficient.

4.1 Spatiotemporal neighborhoods for SSL

For object detection in general and for pedestrian detection in particular, applying SSL starts by defining which are the neighbors of a given window under analysis. In learning time, such a window will correspond either to the bounding box of a labeled pedestrian or to a rectangular chunk of the background. In operation time (*i.e.*, testing), such a window will correspond to a candidate generated by a pyramidal sliding window scheme or any other candidate selection method. In this paper we assume the processing of image sequences and, consequently, we propose the use of a spatiotemporal neighborhood.

Temporal SSL involves the analysis of window volumes. Therefore, there are several possibilities to consider (see Fig. 2). Let us term as W_f the set of coordinates defining an image window in frame f, and $\mathbf{V}_f = \mathrm{vol}(\cup_{t=0}^{T-1} W_{f-t})$ the window volume defined by a temporal neighbor of T frames. The simplest volume is obtained by assuming fixed locations across frames, which we term as *projection* approach. In other words, $W_f = W_{f-1} = \ldots = W_{f-(T-1)}$. Another possibility consists in building volumes taking into account motion information. For instance, $W_f = W_{f-1} + t_{OF(W_{f-1})}$, where $t_{OF(W_{f-1})}$ is a 2D translation defined by considering the *optical flow* contained in W_{f-1}, and $'+'$ stands for summation to all coordinates defining W_{f-1}.

Spatial SSL involves the analysis of windows spatially overlapping the window of interest (see Fig. 2). For instance, we can fix a 2D displacement $\Delta = (\delta_x, \delta_y)$ and n_x displacements in the x axis, to the left and to the right, an analogously for the y axis given a n_y number of up and down displacements.

Our proposal combines both ideas, the temporal volumes and the spatial overlapping windows, in order to define the spatiotemporal neighborhood required by SSL (Sect. 3).

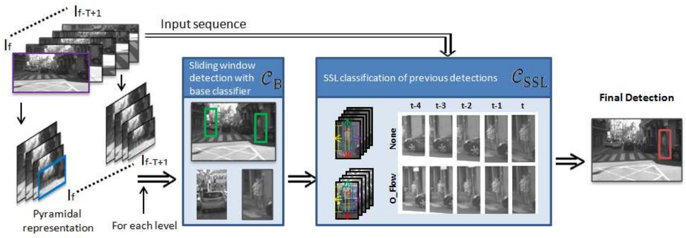

Fig. 3. Two-stage pedestrian detection based on SSL. See main text in Sect. 4.3 for details.

4.2 SSL training

As usual, we assume an image sequence with labeled pedestrians (*i.e.*, using bounding boxes) for training. Negative samples for training are obtained by random sampling of the same images, of course, these samples cannot highly overlap labeled pedestrians. The cross-validation-style rounds of SSL (Sect. 3) are performer with respect to the images of the sequence, not with respected to the set of labeled pedestrians and negative samples as it may suggest the straightforward application of SSL (note that pedestrian/negative labels are for individual windows not for full images). Moreover, as we have seen in Sect. 4.1, the neighborhood relationship is not only temporal but spatial too. The training process is divided in two stages. First, we train the auxiliary classifiers ($\mathscr{C}_{B_k}$) as usual using three bootstraping rounds. Then we train the SSL classifier (using final $\mathscr{C}_{B_k}$ as auxiliary), again we run three bootstrapping rounds for obtaining the final classifier ($\mathscr{C}_{SSL}$). Using the full training dataset, we also assume the training of a base classifier $\mathscr{C}_B$. Another possibility is to understand the different $\mathscr{C}_{B_k}$ as the result of a bagging procedure and ensemble them to obtain $\mathscr{C}_B$. Without loosing generality, in this paper we have focused on the former approach.

4.3 SSL detector

The proposed pedestrian detection pipeline is shown in Fig. 3. As we can see there are two main stages. The first stage basically consists in a classical pedestrian detection method relying on the learned base classifier $\mathscr{C}_B$. In Fig. 3 we have illustrated the idea for a pyramidal sliding window approach, but using other candidate selection approaches is also possible. Detections at this stage are just considered as potential ones. Then, the second stage applies the spatiotemporal SSL classifier, $\mathscr{C}_{SSL}$, to such potential detections in order to reject or keep them as final detections.

There are some details worth to mention. First, the usual non-maximum suppression (NMS) is only done for the output of the second stage. Second, for ensuring that true pedestrians reach the second stage, we apply a threshold on $\mathscr{C}_B$ such that it guarantees a very high detection rate even having a very high rate of false positives. In our experiments this usually implies that while the $\mathscr{C}_B$ processes hundred of thousands windows (for pyramidal sliding window), $\mathscr{C}_{SSL}$ only process a few thousands. Third, although in Fig. 3 we show pyramids of images for a temporal neighborhood of T frames, what we actually keep from frame to frame are the already computed features, so that we compute them only once. However, this depends on the type of temporal neighborhood we use (Sect. 4.1). For instance, using projection style no feature are needed to keep (*i.e.*, keeping the classification scores is enough). However, if we use optical flow we may need to compute features in previous frames if the window under consideration does not map to a location where they were already computed.

5 Experimental results

Protocol. As evaluation methodology we follow the de-facto Caltech standard for pedestrian detection [7], *i.e.* we plot curves of false positives per image (FPPI) *vs* miss rate. The miss rate average in the range of 10^{-2} to 10^{0} FPPI is taken as indicative of each detector accuracy, *i.e.*. the lower the better. Moreover, during testing we consider three different subset: *Near* subset include pedestrians with height equal or higher than 75 pixels, *medium* subset include pedestrian between 50 and 75 pixel height. Finally we group the two previous subset in the *reasonable* subset (height $>=$ 50 pixels).

Table 1. Evaluation of SSL over different datasets, frame rates and pedestrian sizes. For FPPI $\in [0.01, 1]$, the miss rate average % is indicated.

Dataset	FPS	Experiment	Near	Medium	Reasonable
CVC08	Any	Base: HOG+LBP	39.71	50.83	45.91
	3	SSL(Base) Proj. - OptFl.	36.03 - 36.72	50.01 - 50.04	44.40 - 44.02
		Base+HOF	47.98	56.65	50.88
		SSL(Base+HOF) Proj.	37.62	52.21	45.47
	10	SSL(Base) Proj. - OptFl.	35.49 - 34.79	50.22 - 49.42	43.56 - 42.10
		Base+HOF	39.24	52.37	42.43
		SSL(Base+HOF) Proj.	29.42	44.62	37.13
	30	SSL(Base) Proj. - OptFl.	34.18 - 34.01	49.84 - 48.04	42.90 - 41.73
		Base+HOF	37.81	53.39	38.78
		SSL(Base+HOF) Proj.	27.37	46.53	35.85
Caltech	25	Base	45.4	82.3	59.4
		SSL(Base) Proj. - OptFl.	40.6 - 38.9	81.2 - 80.4	59.4 - 57.6
		Base+HOF	33.8	78.4	52.9
		SSL(Base+HOF) Proj.	32.0	77.1	51.6

CVC08 On-board Sequence (CVC08). Since the temporal axis is important for the SSL classifier, we acquired our own dataset to be sure we have stable 30 fps sequences. The sequences were acquired on-board under normal urban driving conditions. The images are monochrome and of 480×960 pixels. We used a 4mm focal length lens, so providing a wide field of view. We drove during 30 minutes approximately, giving rise to a sequence of around 60,000 frames. Then, using steps of 10 frames we annotated all the pedestrians[4]. This turns out in 7,900 annotated pedestrians, 5,400 reasonable and non occluded. We have divided the video sequence into three sequential parts, the first one for training (3,600 pedestrians), the last one for testing (1300 pedestrians), in the middle we have leaved a gap for avoiding testing and training with the same persons.

Caltech dataset. We have also used other popular dataset acquired on-board. The Caltech dataset [7], which contain 3,700 reasonable pedestrians for training. It is worth to mention that the images were acquired at 25 fps.

Base detectors. For the experiments presented in this section we use our own implementation of HOG and LBP features, using TV-L1 [23] for computing optical flow, we obtain HOF features [20] as well. We call Base to the HOG+LBP/Linear-SVM and Base+HOF to the HOG+LBP+HOF/Linear-SVM.

Experiments. Experiments are based on the ST-SSL with $(\Delta x, \Delta y, \Delta f) = (3, 3, 5)$. In preliminary experiments we tested several values of K (Fig. 1), obtaining very similar results, thus we set $K = 1$ for speed up the training. In table 1 we show the results for the SSL experiments. As baseline detectors we use the Base and Base+HOF. The experiments are run over the different datasets, and frame rates (CVC08). We tested them

[4] Publicly available in: http://www.cvc.uab.es/adas/site/?q=node/7

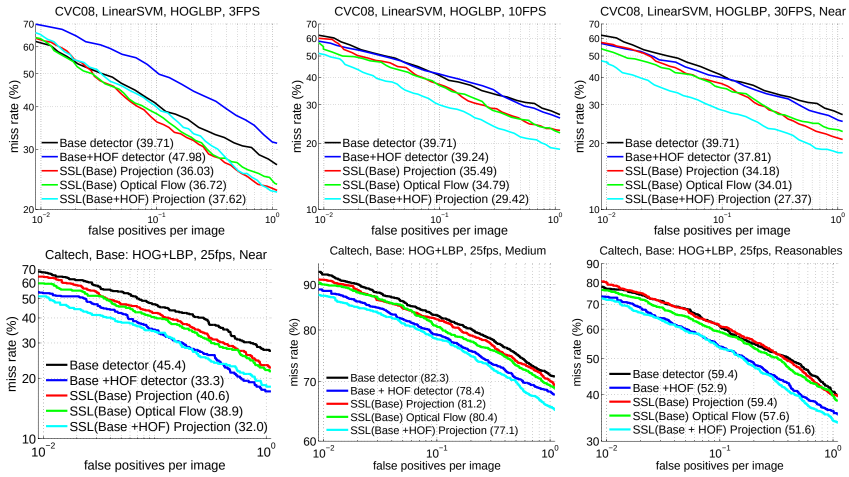

Fig. 4. Results for CVC08 and Caltech datasets. At the top row there are the 30fps, 10fps and 3fps cases of CVC08 using the *near* testing subset. The last two cases are obtained by sub-sampling the video sequence, but always keeping the same training and testing pedestrians. At the bottom row there are the experiments over the *near*, *medium* and *reasonable* testing of Caltech dataset.

for different ranges of pedestrian sizes. We observe significant accuracy improvements for all the tested datasets comparing the baseline detector and its SSL counterpart. For instance, in CVC08 near with SSL(Base+HOF) we obtain an accuracy improvement of ten points approximately. Also, significant accuracy improvements are obtained for all the tested frame rates (30 fps, 10 fps, 3 fps) of CVC08 dataset. Besides, we observe an improvement due to the optical flow in the volume generation at high frame rates. However, no significant difference is observed at low frame rates. The SSL accuracy improvement is more clear for the near pedestrians. In Fig. 4 we plot the accuracy curves obtained for some representative experiments.

Discussion. SSL approach outperforms its baseline in almost all the tested configurations. However, the improvement is more clear for near pedestrians at high frame rates. If we generate the *past neighborhood* over the far away pedestrians, we should expect a *past neighborhood* with pedestrians smaller than the minimum pedestrian size that the base detector can detect. That is why the SSL improvement is not so clear for the medium subset. However, in near pedestrians *past neighborhood* is more probable to find a history of confident responses. This is a very relevant improvement since for close pedestrians the detection system has less time to take decisions like braking or doing any other manoeuvre. Regarding the neighborhood generation approaches, the optical flow slightly improves the projection one as it captures the movement of the pedestrians in the temporal neighborhood.

6 Conclusion

In this paper we have presented a new method for improving pedestrian detection based on spatiotemporal SSL. We have shown how even simple projection windows can boost the detection accuracy in different datasets acquired on-board. We have shown that our approach is effective for different frame rates. In this paper we have focused on HOG+LBP/Linear-SVM and HOG+LBP+HOF/Linear-SVM pedestrian base classifiers, thus, our immediate future work will focus on testing the same approach for other base classifiers of the pedestrian detection state-of-the-art. Regarding the improvement obtained using optical flow neighborhood, we want to further explore different approaches for dealing with the neighborhood generation for moving pedestrians.

Acknowledgments

This work is supported by the Spanish MICINN projects TRA2011-29454-C03-01 and TIN2011-29494-C03-02.

References

1. G. Chen, Y. Ding, J. Xiao, and T. Han. Detection evolution with multi-order contextual co-occurrence. In *CVPR*, Portland, Oregon, USA, 2013.
2. W. Cohen and V. de Carvalho. Stacked sequential learning. In *IJCAI*, Scotland, 2005.
3. X. Cui, Y. Liu, S. Shan, X. Chen, and W. Gao. 3d haar-like features for pedestrian detection. In *ICME*, Bejing, China, 2007.
4. N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *CVPR*, San Diego, CA, USA, 2005.
5. N. Dalal, B. Triggs, and C. Schmid. Human detection using oriented histograms of flow and appearance. In *ECCV*, Graz, Austria, 2006.
6. Y. Ding and J. Xiao. Contextual boost for pedestrian detection. In *CVPR*, USA, 2012.
7. P. Dollár, C. Wojek, B. Schiele, and P. Perona. Pedestrian detection: an evaluation of the state of the art. *T-PAMI*, 34(4):743–761, 2012.
8. M. Enzweiler and D.M. Gavrila. A multi-level mixture-of-experts framework for pedestrian classification. *T-IP*, 20(10):2967–2979, 2011.
9. P. Felzenszwalb, R. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part based models. *T-PAMI*, 32(9):1627–1645, 2010.
10. D. Gerónimo and A. López. *Vision-based Pedestrian Protection Systems for Intelligent Vehicles*. Springer, 2013.
11. D. Gerónimo, A. Sappa, D. Ponsa, and A. López. 2D-3D based on-board pedestrian detection system. *CVIU*, 114(5):583–595, 2010.
12. M. Jones and D. Snow. Pedestrian detection using boosted features over many frames. In *CVPR*, Anchorage, AK, USA, 2008.
13. Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. In *ICCV*, Beijing, China, 2005.
14. J. Lafferty, A. McCallum, and F. Pereira. Real-time pedestrian detection with deformable part models. In *IV*, Madrid, Spain, 2012.
15. J. Marin, D. Vázquez, A. López, J. Amores, and L. Kuncheva. Occlusion handling via random subspace classifiers for human detection. *Cyber*, 2013.
16. J. Marin, D. Vázquez, A. López, J. Amores, and B. Leibe. Random forests of local experts for pedestrian detection. In *ICCV*, Sydney, Australia, 2013.
17. D. Ramanan. *Part-based Models for Finding People and Estimating Their Pose*. 2009.
18. D. Vázquez, A. López, J. Marín, D. Ponsa, and D. Gerónimo. Virtual and real world adaptation for pedestrian detection. *T-PAMI*, 2013.
19. P. Viola, M. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *ICCV*, Nice, France, 2003.
20. S. Walk, N. Majer, K. Schindler, and B. Schiele. New features and insights for pedestrian detection. In *CVPR*, San Francisco, CA, USA, 2010.
21. X. Wang, T.X. Han, and S. Yan. An HOG-LBP human detector with partial occlusion handling. In *ICCV*, Kyoto, Japan, 2009.
22. C. Wojek, S. Walk, and B. Schiele. Multi-cue onboard pedestrian detection. In *CVPR*, Miami Beach, FL, USA, 2009.
23. C. Zach, T. Pock, and H. Bischof. A duality based approach for realtime tv-l1 optical flow. In *DAGM*, Heidelberg, Germany, 2007.