Dynamic attention priors: a new and efficient concept for improving object detection ☆

Alexander R.T. Gepperth[a], Michael Garcia Ortiz[b], Egor Sattarov[a], Bernd Heisele[c]

[a] *ENSTA ParisTech, 828 Boulevard des Marechaux, 91762 Palaiseau, France*
[b] *Aldebaran Robotics, 168 bis 170 rue Raymond Losserand, 75014 Paris, France*
[c] *Honda Research Institute USA, Inc., 425 National Avenue, Suite 100, Mountain View, CA-94043, USA*

Abstract

Recent psychophysical evidence in humans suggests that visual attention is a highly dynamic and predictive process involving precise models of object trajectories. This is in a certain contrast to computational models where visual attention is derived from static quantities, and used to modulate only the *present* perception. In this article, we present a proof-of-concept that predictive spatial attention can benefit a technical system solving a challenging visual object detection task. To this end, we introduce a Bayes-like modulation of dense detection likelihoods, derived from a sliding-window SVM detector, by dynamic attention priors (DAPs), which enhance detection likelihoods at positions predicted by the extrapolation of past object trajectories.

Using a set of real-world video sequences containing pedestrians in a parking lot setting, we show that predictive visual attention as realized by DAPs can improve detection performance significantly as compared to a baseline condition without DAPs, i.e., just relying on local visual patterns.

☆This work is supported by Honda Research Institute USA, Inc.

Email addresses: alexander.gepperth@ensta-paristech.fr (Alexander R.T. Gepperth), michael.garcia.ortiz@gmail.com (Michael Garcia Ortiz), egor.sattarov@u-psud.fr (Egor Sattarov), bheisele@hra.com (Bernd Heisele)

1. INTRODUCTION

There is an extensive body of biological insights on various aspects of visual attention, which is sometimes seen as guided by static local image properties[24], sometimes by static spatial context[39]. Even if non-static image features, such as local motion, are used[13, 27], such attention mechanisms are always reactive in the sense that they guide attention towards the detected features but do not anticipate future events.

Recent work[23] however reveals that humans learn highly precise dynamic models *predicting* the movement of objects, and that such predictions are used to guide eye movements to the predicted locations ahead of time. This predictive mechanism is shown to permit the visual pursuit of highly dynamic objects, such as squash balls, with the very limited amount of fixations per second that can be realized by the human visual system.

As even stronger restrictions usually apply in technical systems, we consider this mechanism of predictive visual attention to be a crucial ingredient in the analysis of dynamic scenes. In particular, learned high-level models of future object behavior may permit to keep track of complex object motion with a small number of measurements (fixations), and help to make detection more robust in case of simple or no motion.

This article proposes a predictive attention mechanism similar in spirit to [23] and presents a proof-of-concept for its added value by employing these so-called *dynamic attention priors* (DAPs) in a visual pedestrian detection task.

We chose pedestrian detection for this evaluation because it is considered to be a very challenging detection task[33] that is basically unsolved by state-of-the art methods. Therefore, any improvement DAPs can contribute to this difficult task can be considered significant, especially given that they come at a negligible computational cost. Nevertheless, this article should definitely be considered a proof-of-concept for the worthwhileness, efficiency and feasibility of DAPs, and not as a study on pedestrian detection which would require a much more extensive evaluation on much bigger and more challenging benchmark databases.

1.1. Motivation, system structure and novelty

Motivation. The motivation for the presented work is twofold: first of all, we wish to give a system-level realization of an important aspect of biological

visual perception. Secondly, we wish to show that technical systems can profit from it with little changes or performance overhead.

Overview. The overall structure of the presented system is shown in Fig. 1. We extend a visual pedestrian detector, which operates by analyzing local pixel patterns in a sliding-window fashion, by DAPs. The predictive aspect of DAPs is contributed by a module for trajectory extrapolation (often termed "tracking") which predicts imminent pedestrian positions by an analysis of past detections. In the manner of biological models[24], DAPs are applied in a multiplicative fashion to the dense array of detection likelihoods obtained from the detector, at locations where pedestrians are likely to occur in the near future. In this way, detections are stabilized and small deviations from learned appearance models (which always occur, and which lead to the typical on-off "flickering" of detections) are compensated for, at least as long as the attended locations are the correct ones. In case they are not, DAPs have little effect due to the modulatory nature of attention [21]: if there is little evidence to begin with, it will be enhanced by modulation but nevertheless remain insignificant. This mechanism is strikingly analogous to Bayesian inference, and indeed it has been speculated [44] that human perception is, to a large extent, a probabilistic inference process. We establish by quantitative evaluation that DAPs are beneficial for applied tasks, in our case pedestrian detection, and that they continue to have these properties even if the assumed (simple) motion model is locally violated. In addition, DAPs have the advantage of being extremely computationally efficient. In order to show that our results are not the artifacts of a particular detection or tracking method, we perform experiments twice, each time using a different combination of detection and tracking algorithms.

Novelty. The presented architecture, which is generic and in no way limited to pedestrian detection, proposes a previously unexplored way of boosting object detection accuracy by predictive spatial attention, making use only of components that any real-world object detection system needs to include in any case, i.e., detection and tracking. In this highly dynamic approach, detection and trajectory analysis (tracking) mutually influence on each other instead of being arranged in a linear processing chain, while retaining robust and stable dynamics. Apart from their conceptual novelty in real-world object detection, we additionally propose an extremely efficient calculation scheme for DAPs which makes their application appealing especially in resource-constrained real-world systems.

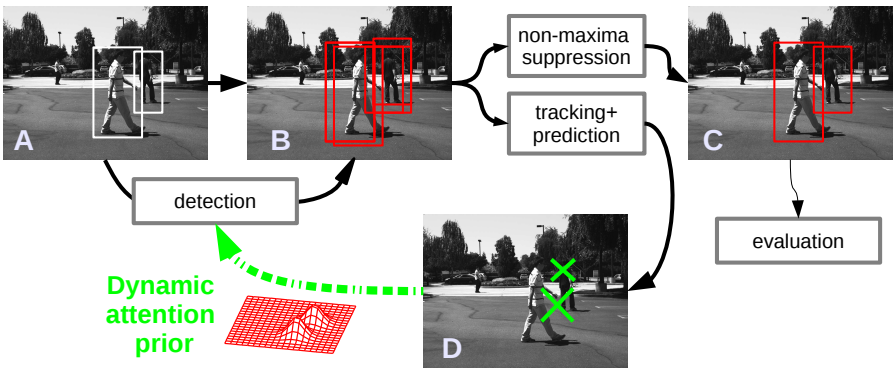

Figure 1: Block structure of the real-time pedestrian detection system. **A** Original image, white boxes show pedestrians that are to be found. The pedestrian left of the center is too small to be detected and is thus excluded here **B** Detections, indicted by red boxes, resulting from sliding window classification **C** Results of non-maxima suppression (NMS) removing overlapping detections. These detections can be considered the final detection result and are passed to evaluation **D** Predictions generated from past detections. Prediction centers and sizes are indicated by green crosses of varying size, and serve as the sources for dynamic attention priors (green dashed arrow) modulating the detection process.

1.2. Related work

There is extensive literature about the computational modeling of the various aspects of visual attention[22, 31, 13, 42, 30, 6]. In many contributions, scene context is used to derive static spatial attention [36, 34, 14, 28, 20, 7, 13, 25]. This body of work shows that visual attention is a potentially powerful tool to improve real-world object detection, but also that the effort to make it work in real problems is a considerable one. In particular, several authors treat visual attention as a kind of Bayesian inference process where the "attention prior" is combined with a likelihood term arising from a detection module[39, 36, 34]. Whereas a spatial attention prior can be easily expressed in a probabilistic form, the detection scores coming from a real-world object detector generally need to be "converted" to probabilities which is not always straightforward and involves a complicated calibration process[34, 35]. In this contribution we show that attentional modulation can work very well even without such calibration, which we avoid as it is

computationally costly. Furthermore, our approach uses dynamic quantities such as the object's own motion in order to guide attention to the correct locations. An aspect which our contribution shows explicitly, and which is often neglected in conceptual works on visual attention, is that by reducing the number of search locations, visual ambiguity is reduced as well, leading to an overall increase in detection accuracy.

Pedestrian detection has been researched for decades, and thus there is a very large body of previous work [9, 11, 15, 32, 5, 8, 12, 1, 14], not all of it capable of real-time operation. System approaches (see Sec. 1.2) are based on feature extraction, detection and object tracking stages, the latter mostly realized by Kalman or particle filtering[26, 29, 37, 38, 3, 43, 16, 41]. Although tracking is included into most pedestrian detection systems, it is always considered as a post-processing step to detection and thus as the final point in a linear processing chain. There is no work we know of that makes use of object-centered dynamic attention mechanisms as we propose them here to influence its detections, except potentially [16]. As suggested by a recent comprehensive comparison study, it seems that state-of-the-art approaches for pedestrian detection are currently reaching a boundary that is not easy to break[33].

1.3. Messages and structure of the article

This article uses a bio-inspired attention mechanism to facilitate a strongly application-oriented problem. It should be noted that we consider this approach promising not simply because of the biological analogy, because of the practical advantages that can be gained in this way. In particular, this article intends to demonstrate the following things:

- **Dynamic attention priors are feasible and efficient** Here, it will be shown that detection likelihoods in general have a form that allows multiplicative modulation by attention priors, and that the application of the latter does not incur significant computational cost.

- **Dynamic attention priors improve the overall performance of object detectors** In particular, they compensate slight deviations from the learned appearance models, which is a case that often occurs.

- **Dynamic attention priors do not cause incorrect detections if predictions are incorrect** As predictions of future detections are based on a motion model, they can become incorrect when this model

is violated. In this case, it must be shown that actual detections are not, or at best slightly, affected.

To deliver these messages, the article proceeds as follows: in Sec. 2, the training, evaluation and the component parts of the real-time system are described in detail. Subsequently, we will present experiments validating the previous points in Sec. 3 and discuss the significance of the results in Sec. 4. In Sec. 5, we will conclude this contribution by providing an outlook of our future works.

2. Methods

In order to ensure that results do not depend on a specific choice of detection or tracking method, we verify the feasibility of DAPs for two combinations of different detection and tracking algorithms. On the one hand, we combine a cascaded HOG+SVM based detection method[17] with a simple, self-created tracking method based on a linear trajectory assumption (system **I**), and on the other hand we employ a standard HOG+SVM detector [10] in combination with a state-of-the-art particle filter-based PHD tracker [26] (system **II**).

Experiments are conducted for systems I and II in an identical fashion on the same data.

2.1. Object detection algorithms

Each algorithm, independently of the used model, provides at time t a list of detections $D_{j,t}$. Due to the multi-scale detection approach described in Sec. 2.1, the size of a detected pedestrian is a multiple of w_0, h_0 (see Sec. 2.1) which are constants. The spatial scales are numbered in descending order, the one with highest resolution being assigned the index $i = 0$. Each subsequent scale is subsampled along both dimensions by a factor of $\sqrt{2}$ using bicubic interpolation. It is the most practical solution to identify each scale by its downsampling factor σ w.r.t. to the original scale of highest resolution, which will therefore have the form $\sigma(i) = 2^{i/2}, i \in \mathbb{N}_0^+$. We can thus characterize any detection D_j by its center coordinates $\vec{c}(D_j)$, by its associated scale $d(D_j)$ and by its score $s(D_j)$:

$$D_j(t) = [\vec{c}(D_j), d(D_j), s(D_j)] \tag{1}$$

The variable $d(D_j)$ takes its values in powers of $\sqrt{2}$: $[1, \sqrt{(}2), 2...]$ depending on the used spatial scales during the detection process. The center coordinates $\vec{c}(D_j)$ take values only at the locations of the grid used for the sliding window detection, see Sec. 2.1. We shall denote a particular detection score at spatial grid position $\vec{x}$, spatial scale σ and time t as $s(\vec{x}, \sigma, t)$.

Cascaded HOG+SVM detection. The global structure of this pedestrian detection method is given in [17]. Due to GPU acceleration, the whole system can process 15 color images (800x600 pixels) per second on an off-the-shelf PC (2.0GHz) equipped with a nVidia geForce GTX 580 graphics card. The method is based on the computation of Histograms of Oriented Gradients (HOG) features [5] using the "GPU" module of the free OpenCV library[2]. Adopting the terms presented in [5], we use the following parametrization for HOG:

- a cell size of 8x8 pixels
- a block size of 16x16 pixels
- a border of 0 pixels
- a window size of $w_0 \times h_0$=32x64 pixels
- a window stride of 4x4 pixels
- a factor of $\sqrt{2}$ between scales

The pedestrian detection system consists of a cascade of linear and non-linear support vector machines that are applied in a sliding-window fashion at S^{det} spatial scales to the computed HOG features. This cascade approach allows us to circumvent the speed disadvantage of non-linear SVMs as they are only applied to the (few) detections that survive the linear SVM stage. We therefore consider a detection window at time t, with center point $\vec{x}$ at scale σ, to contain a pedestrian if and only if the corresponding scores from both the linear and the nonlinear-SVMs, $s_{\text{lin}}(\vec{x}, \sigma, t)$ and $s_{\text{rbf}}(\vec{x}, \sigma, t)$ exceed their respective thresholds, $\theta^{\text{det}}_{\text{lin}}$ and $\theta^{\text{det}}_{\text{rbf}}$. To save computation time, we apply the non-linear SVMs only to windows for which $s_{\text{lin}} > \theta^{\text{det}}_{\text{lin}}$. For training the linear and non-linear SVMs for pedestrian detection, we used the training sets from the Daimler Monocular Pedestrian Detection Benchmark (DM-PDB, [10]), as well as from the Daimler Stereo Pedestrian Benchmark [9].

All training is performed using the libSVM library and tools [4]. We resize all training images to a common size of size 32x64 pixels prior to training. From these resized images, we compute HOG features according to [5] and store the resulting feature vectors, along with suitably assigned class memberships, in a libSVM training file. Linear and RBF kernel C-SVC training is subsequently conducted using this libSVM training file to obtain linear and non-linear pedestrian detectors that are able to distinguish pedestrians from background. Further details on training can be found in [17].

Standard HOG+SVM detection. This method corresponds exactly to the linear stage of the cascaded HOG+SVM method, except that the window size w_0, h_0 is 48x96 pixels instead of 32x64. In order to avoid training issues, we use the trained SVM already available in the OpenCV library, implementing the pedestrian detector whose training is described in detail in [10]. Due to the excellent speed/accuracy trade-off this method offers even without GPU acceleration, it can still be considered a state-of-the-art architecture as other detectors that yield better performance[12] are much more demanding in terms of computation time or much more complex to implement [17].

2.2. Non-maxima suppression

There will usually be clusters of overlapping detections due to positional invariance of the basic HOG features used in both detection algorithms. To obtain the final results that are passed on to other, possibly security-relevant applications, we therefore perform a simple non-maxima suppression (NMS) step that selects detections whose score exhibits a local maximum. NMS is a standard post-processing method in object detection which expects a set of bounding boxes with associated scores $\{D_j\}$, and produces a thinned out list of boxes/scores $\{\tilde{D}_j\}$ where only the locally most confident detections survive. In detail, the algorithm runs as follows, relying on the overlap measure

$$o(D_i, D_j) = \frac{\text{area}(D_i \cap D_j)}{\text{area}(D_i \cup D_j)} \tag{2}$$

Algorithm *Simple NMS*$(\{D_j\})$
1. Sort $\{D_j = (\vec{c}(D_j), d(D_j), s(D_j))\}$ in descending order of score $s(D_j)$
2. **for** $a \leftarrow 1$ **to** N
3. **for** $b \leftarrow a + 1$ **to** N
4. **if** D_a not marked for deletion

5. **then**
6. **if** $o(D_a, D_b) \geq \theta_{\text{nms}}$
7. **then** mark D_b for deletion
8. Erase marked detections and return list

2.3. Tracking and prediction

We implement two tracking algorithms in order to show that DAPs are feasible independently of the concrete tracking model that is used. One method is a self-created algorithm optimized for execution speed called LRT("linear regression tracker"), the other is a state-of-the-art particle-filter based PhD tracker.

2.3.1. Simple multi-object tracking with linear trackers

This method, which we shall term "linear regression tracker" (LRT) operates on the results of the pedestrian detection *before* applying NMS as described in Sec. 2.2.

LRT is represented by a time-variable number of tracks T_k, $K(t) > k \geq 0$. Each track has an associated track state allowing to predict the quantity $P_{T_k,t} = [\vec{c}_{T_k,t}, d_{T_k,t}, \vec{v}_{T_k,t}]$ which contains, respectively, the center coordinates, the scale and the speed of a pedestrian. A measure of the prediction error $\epsilon_{T_k,t}$ is also computed for each track. The internal variables of a track are the linear regression coefficients for position and scale, $\vec{\alpha}_{T_k}$, $\vec{\beta}_{T_k}$, $\alpha^{\sigma}_{T_k}$, $\beta^{\sigma}_{T_k}$, a list of the T^{tr} past assigned detections $\mathcal{L}_{T_k,t}$, as well as a probability measure $\pi_{T_k,t}$ that counts the number of successive frames the tracker was not assigned a detection, and which is initially 0 for new tracks.

For each frame, the following steps are executed in the given order:

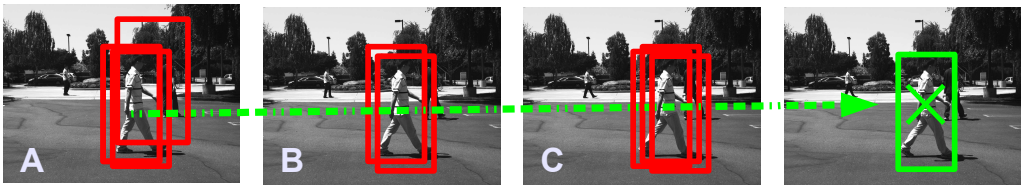

Figure 2: Schematics of linear trackers: linear regression over all past detections (red rectangles) is used to predict the coordinates and scale (green rectangles) of the next detection.

Prediction. Every track T_k, $k = 0 \dots K-1$ uses its internal variables $\vec{\alpha}, \vec{\beta}$ and $\alpha^\sigma, \beta^\sigma$ to predict the current state of the tracked pedestrian rectangle $[\vec{c}_{T_k,t'}, d_{T_k,t'}]$ for $t' = t$.

$$\begin{aligned} \vec{c}_{T_k,t'} &= \vec{\alpha}_{T_k,t-1} * (t' - t_0) + \vec{\beta}_{T_k,t-1} \\ d_{T_k,t'} &= \alpha^\sigma_{T_k,t-1}(t' - t_0)\beta^\sigma_{T_k,t-1} \end{aligned} \tag{3}$$

where $t_0 \equiv t - T$. If the track is less than two timesteps old, no linear regression has been performed yet and the last associated detection is returned as the result of prediction.

Association. Each track T_k provides a prediction of the current location of the pedestrian it is tracking, so we obtain a list of predictions $P_{T_k,t} = [\vec{c}_{T_k,t}, d_{T_k,t}]$. For each couple $D_{j,t}, P_{T_k,t}$, the detection and the prediction are compared by measuring the overlap measure introduced in Sec. 2.2 between the area of the detection box and the area of the predicted box. This overlap measure is a very good indicator for evaluating if two detection boxes are close to each other, and if they are of a similar size. If the overlap exceeds the threshold $\theta^{\text{tr}}_{\text{ass}}$, and if additionally the detection has a score that exceeds a certain threshold θ^{track}, the detection $D_{j,t}$ is assigned to track T_k:

$$\begin{aligned} \text{if } o(P_{T_k,t}, D_{j,t}) > \theta^{\text{tr}}_{\text{ass}} \wedge s_{\text{det}}(D_{j,t}) > \theta^{\text{tr}}_s \\ \rightarrow \text{ assign } D_{j,t} \text{ to track } T_k \end{aligned} \tag{4}$$

We try to assign all detections to the currently active tracks. Each track that is assigned a detection resets its counter $\pi_{T_k,t} = 0$. If a track is not assigned any detections, its counter is increased to $\pi_{T_k,t} = \pi_{T_k,t-1} + 1$. Each unassigned detection spawns a new track that is initialized with the current detection as only element of $\mathcal{L}(t)$ and having a counter of $\pi_{T_k,t} = 0$.

Observation. All tracks that have been assigned detections update their internal parameters. This consists of updating the internal state variables $\vec{\alpha}_{T_k}$, $\vec{\beta}_{T_k}$, $\alpha^\sigma_{T_k}$ and $\beta^\sigma_{T_k}$. To this end, the list $\mathcal{L}$ of past detections is extended by all detections assigned in the current frame. Conversely, if the list contains already detections from T^{tr} different timesteps, the oldest timestep t^* is identified and all detections associated to the track at t^* are deleted. Subsequently, standard linear regression techniques are used to fit a straight line through assigned detection centers and detection sizes, yielding the updated coefficients $\vec{\alpha}_{T_k,t}$, $\vec{\beta}_{T_k,t}$, $\alpha^\sigma_{T_k,t}$ and $\beta^\sigma_{T_k,t}$. From the same calculation, we obtain the error measures for position and scale, and we set $\epsilon_{T_k,t}$ to their arithmetic mean.

Deleting single-object trackers. Each track verifies its probability measure $\pi_{T_k,t}$. If no detection has been assigned to this track for a period of time $\delta^{\text{tr}}_{\text{idle}}$, we consider that the track lost the pedestrian, or that the pedestrian is occluded or out of the field of view. Subsequently, the track is deleted.

Merging. All tracks are compared in a pairwise fashion. If their estimation for the current positions are close, and if the estimation of their speeds are similar, we merge the trackers, considering that they have been tracking the same pedestrian. We keep the one with the lowest current prediction error.

2.4. Particle-based PHD tracking

The PHD filter is represented by a time-variable number of tracks T_k, $K(t) > k \geq 0$. Each track T_k contains $0 \leq n < N^{\text{tr}}$ particles. Each particle $\xi_{n,k,t}$ contains the quantities $\vec{c}_{n,k,t}$, $d_{n,k,t}$ and $\vec{v}_{n,k,t}$, $\vec{c}_{n,k,t}$ being the center coordinates, $d_{n,k,t}$ the detection scale and $\vec{v}_{n,k,t}$ the associated speed. Also, each particle has a weight $\omega_{n,k,t}$. Each track has an associated track state $Xi_{k,t}$ containing the quantities $\vec{c}_{T_k,t}$, $d_{T_k,t}$ and $\vec{v}_{T_k,t}$ which are in complete analogy to particle states. A track's parameters are: death probability P^{tr}_d, birth probability P^{tr}_b, track probability increase and decrease steps P^{tr}_+ and P^{tr}_-, false negative probability P^{tr}_{fn}, a vector of resampling and association parameters $\sigma^{\text{tr}}_i, i \in \{0, 1, 2\}$ for position,size and speed respectively, and the influence and resampling coefficients ν^{tr}, ρ^{tr}. In order not to complicate the tracking with multiple overlapping detections, we perform non-maxima suppression on detections *prior* to tracking as detailed in Sec. 2.2.

Prediction. Tracks and theirs particles propagate themselves according to a linear movement model: $\xi_{n,k,t|t-1} = f_{\xi_{n,k,t}|\xi_{n,k,t-1}}(\xi_{n,k,t-1})$. For the particular case, $d_{n,k,t|t-1} = d_{n,k,t-1}$, $\vec{v}_{n,k,t|t-1} = \vec{v}_{n,k,t-1}$, $\vec{c}_{n,k,t|t-1} = \vec{c}_{n,k,t-1} + \vec{v}_{n,k,t|t-1}$.

Association. Detections are subjected to non-maxima suppression and filtered by the conditions $s(D_{j,t}) > \theta^{\text{tr}}_s$. The remaining detections $\{\tilde{D}_j\}$, are assigned to existing tracks $0 \leq k < K$, see below for more details. Those tracks which are assigned observations increase their associated probability by P^{tr}_d: $P_{k,t} = \max(P_{k,t-1} + P^{\text{tr}}_+, 1)$. Tracks that are not associated update their probability by: $P_{k,t} = \max(P_{k,t-1} - P^{\text{tr}}_-, 0)$. Detections which are not associated to any tracks create new tracks having an initial probability of $P_{k,t} = P^{\text{tr}}_b$. In case a new track is created, the probabilities of its particles are $\omega_{n,k,t} = P_{k,t}/N^{\text{tr}}, n = 0, \ldots N^{\text{tr}} - 1$ and their states are initially those of the detection creating the track. In detail, we proceed as follows:

1. For all pairs $(\tilde{D}_j, k)$, we calculate the similarity

$$G(k, \tilde{D}_j) = \mathcal{N}(\vec{c}_{\tilde{D}_j} - \vec{c}_{T_k,t|t-1}, \sigma_0^{\text{tr}})\mathcal{N}(d_{\tilde{D}_j} - d_{T_k,t|t-1}, \sigma_1^{\text{tr}}) \\ \mathcal{N}(\vec{v}_{\tilde{D}_j|T_k} - \vec{v}_{T_k,t|t-1}, \sigma_2^{\text{tr}}) \tag{5}$$

   as a product of three Gaussians with manually tuned variances $\sigma_{0,1,2}^{\text{tr}}$, where the "speed" of a detection relative to a track is defined as $\vec{v}_{\tilde{D}_j|T_k} = \vec{c}_{T_k,t|t-1} - \vec{c}_{\tilde{D}_j}$.
2. Find the closest track-detection pair. If the similarity exceeds the association threshold $\theta_{\text{ass}}^{\text{tr}}$, associate chosen detection to chosen track. Remove this track from list of pairs to associate. Repeat 2). If the similarity is smaller than the threshold, go to 3).
3. Finally, we have a list of associated pairs, a list of non-associated detections and a list of non-associated tracks.

Observation. For each new observation $\tilde{D}_j$ and for each particle $\xi_{n,k}$ of the track the observation was assigned to, the Gaussian similarity $G(\xi_{n,k}, \tilde{D}_j)$ from eqn. (5) is calculated, only this time between a particle and an observation and with all variance parameters σ_i^{tr} multiplied by an influence coefficient ν^{tr}. ν^{tr} governs how strongly particle weights are influenced by detections, thus shifting the balance between motion model and observations. Resulting similarities are normalized per observation, therefore the sum of all similarities from one observation is one. The weights of particles are calculated as a sum of similarities: $\omega_{n,k} = \sum_j G(\xi_{n,k}, \tilde{D}_j) + \omega_{n,k} \times P_{fn}^{\text{tr}}$. The last term represents the "old" particle weights in order to stabilize against missed detections.

Resampling. If track probability $P_{k,t}$ falls below the death probability P_{d}^{tr}, a track is deleted. Otherwise, its particles resample themselves using random Gaussian fluctuations, depending on $\rho^{\text{tr}}\sigma_i^{\text{tr}}$ and a multiplier $1/P_{k,t}$ which increases all σ_i^{tr} if a track's probability is lower than one, in order to disperse particles when a track is "lost", making it easier to "pick up" the track later. We term ρ^{tr} the "resampling coefficient" governing the noise added during resampling.

Merging. If some of tracks are very close and move with the same trajectory, they are supposed to be one and the newest track is deleted.

Correction. New track states are found by averaging over the internal states of all associated particles: $Xi_{k,t} = \frac{1}{N}\sum_{n=1}^{N} \xi_{n,k,t}\omega_{n,k,t}$

2.5. Application of attention priors

The way of applying a spatial attention priors depends on which of the two tracking algorithms is used, although the general principle is identical. Basically, we use the predictions of each track to increase detection scores around locations in the current image where pedestrians are predicted to be. Predictions from tracks can come in the form of a predicted pedestrian rectangle $P_{k,t}$, or in the form of predicted particles and their associated weights, each of which represents a predicted pedestrian hypothesis.

The dynamic attention prior will boost, at each scale, the dense array of detection scores at locations that are "close" to a prediction. Neighbouring scores in adjoining spatial scales are enhanced as well, albeit with a discount depending on the scale difference. The boost is always excitatory, or at worst neutral far away from any prediction, in analogy to biological modulatory feedback signals. The mechanism is visualized in Fig. 3, which also shows that only score modifications close to predicted tracks need to be computed. In regions sufficiently far from any track, virtually no modification takes place and computations can be skipped, leading to a highly efficient way of applying DAPs.

System **I***: LRT tracker+cascaded HOG detector.* Linear detection scores are multiplied by $(1 + \gamma^{\text{scale}}\gamma^{\text{xy}})$, where γ^{xy} represents a a Gaussian centered on a track's prediction $P_{k,t}$ and having a standard deviation of σ^{dap}, and γ^{scale} a discount factor depending on scale difference. Given that a tracker k predicts the pedestrian position and size $P_{k,t} = [\vec{c}_{T_k,t}, d_{T_k,t}]$, the scores $s_{lin}(\vec{x}, \sigma, t)$ as spatial scale σ will be modified as follows:

$$
\begin{aligned}
s_{\text{lin}}(\vec{x}, \sigma, t) &\rightarrow \left(s_{\text{lin}}(\vec{x}, \sigma, t) + \Delta^{\text{dap}}\right)\left(1 + A^{\text{dap}}\gamma^{\text{scale}}\gamma^{\text{xy}}\right) - \Delta^{\text{dap}} \qquad (6)\\
\gamma^{\text{scale}} &= \frac{1}{1 + \left(\log_{\sqrt{2}}\sigma_k - \log_{\sqrt{2}} d_{T_k,t}\right)^2}\\
\gamma^{\text{xy}} &= \exp - \frac{||\vec{x} - \vec{c}_{T_k,t}||}{2 * \sigma^{\text{dap, 2}}}
\end{aligned}
$$

We observe that the scores are modified by the proximity of a prediction both in 2D and in scale space, the latter of which is only relevant for identical or directly adjacent scales. The scores of the RBF classifier are modified in an identical fashion although, to save computation time, the modification is only applied where linear scores exceed the detection threshold $\theta^{\text{det}}_{lin}$. Δ^{dap} governs the minimal score that will be increased by the attention prior, whereas

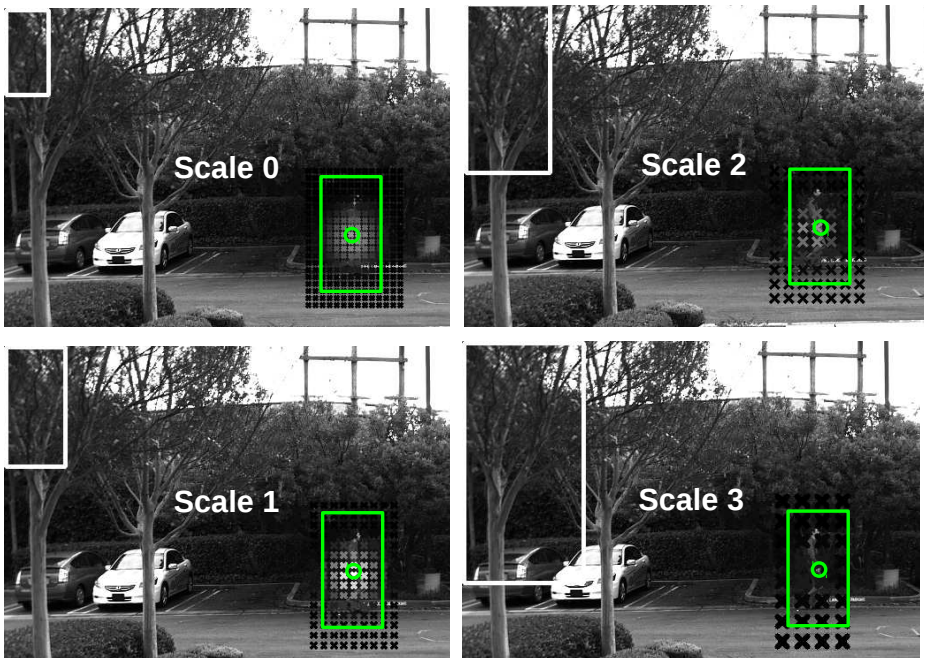

Figure 3: Multi-scale modification of detection scores around a correctly predicted pedestrian (or particle), where the green circle indicates the prediction center and the green box its associated scale). Grey crosses represent the positions of detection scores (sliding window centers) at each scale, the level of brightness indicating the strength of the boost caused by DAPs. The white box in the top-left corner of each image indicates sliding window size at a particular scale. As the size of the predicted pedestrian is roughly that of scale 1 (since white and green rectangles have similar size), the scores at scale 1 get boosted more strongly than at other scales. At scale 3 no significant boost takes places any longer. Please not that only detection scores around the pedestrian are shown, in reality the whole image is densely covered at each scale.

$A^{\text{dap}} \geq 0$ controls overall feedback strength. Indeed, if a score is below Δ^{dap}, the modification will decrease its value, and for $A^{\text{dap}} = 0$ feedback is turned off.

System **II***: PhD tracker+simple HOG detector.* As the detector in system **II** is basically identical to the linear detection stage of system **I**, the formula for

Param.	S^{det}	$\theta^{\text{det}}_{\text{lin}}$	$\theta^{\text{det}}_{\text{rbf}}$	θ^{nms}	T^{tr}	$\theta^{\text{tr}}_{\text{ass}}$	$\delta^{\text{tr}}_{\text{STM}}$	d^{tr}
Value	5	-1.5..1.5	-1.5..1.5	0.25	20	0.0	5	15
Param	$\theta^{\text{tr}}_{\text{del}}$	$\delta^{\text{tr}}_{\text{idle}}$	σ^{dap}	Δ^{dap}	A^{dap}	n^{tr}	θ^{tr}_{s}	
Value	0.25	10	15pix	1.5	0 or 0.7	5	0.5	

Table 1: Parameters for system **I** used in all experiments.

Param.	S^{det}	$\theta^{\text{det}}_{\text{lin}}$	P^{tr}_{+}	θ^{nms}	ν^{tr}	$\theta^{\text{tr}}_{\text{ass}}$	ρ^{tr}	σ^{tr}_{i}
Value	5	$-1\ldots3$	0.3	0.25	1.0	0.2	0.3	2,2,1
Param	P^{tr}_{d}	P^{tr}_{b}	σ^{dap}	Δ^{dap}	A^{dap}	N^{tr}	θ^{tr}_{s}	P^{tr}_{-}
Value	0.4	0.5	24pix	3	0 or 0.3	250	1.5	0.02

Table 2: Parameters for system **II** used in all experiments.

adapting linear detection scores is very similar to eqn.(6), except that each track's predicted particles are now the basis for score modification. Each score $s(\vec{x},\sigma,t)$ at position $\vec{x}$ and scale σ is modified as follows:

$$s(\vec{x},\sigma,t) \rightarrow \left(s(\vec{x},\sigma,t)+\Delta^{\text{dap}}\right)\left(1+A^{\text{dap}}\sum_{k}\sum_{n}\gamma^{\text{scale}}_{n,k,t}\gamma^{\text{xy}}_{n,k,t}\omega_{n,k,t}\right)-\Delta^{\text{dap}} \tag{7}$$

$$\sigma_{n,k,t} = \log_{\sqrt{2}} d_{n,k,t}$$

$$\gamma^{\text{scale}}_{n,k,t} = \exp-\left(\frac{\left(\sigma_{n,k,t}-\log_{\sqrt{2}}\sigma\right)^{2}}{2}\right)$$

$$\gamma^{\text{xy}}_{n,k,t} = \exp-\left(\frac{||\vec{c}_{n,k,t}-\vec{x}||}{2*\sigma^{\text{dap},\,2}}\right)$$

By comparing eqns.(6) and (7) while disregarding the subsequent details, one best perceives the structural identity between the two ways of applying DAPs.

2.6. Systems

System **I** and **II** are structurally very analogous. For a better comprehension, we present their working in pseudocode form. A part from the different detection and tracking methods that are used, both differ mainly in the way

NMS is applied as the PHD tracker works better when operating on detections that are pre-filtered by NMS. For obtaining experimental results, we use both systems with the parameter values given in Tabs. 1 and 2.

Algorithm *System* **I** Input: Image at time t, Output: detections $\tilde{D}_{j,t}$

1. Apply cascaded HOG detector to image, giving scores $s^{\text{lin}}(\vec{x}, \sigma, t)$ and $s^{\text{rbf}}(\vec{x}, \sigma, t)$
2. **for** $k \leftarrow 0$ **to** number of tracks $K - 1$
3. **do** Predict current pedestrian rectangle $P_{T_k,t} = [\vec{c}_{T_k,t}, d_{T_k,t}]$ for each track T_k
4. Apply DAPs to scores $s^{\text{lin}}(\vec{x}, \sigma, t)$, $s^{\text{rbf}}(\vec{x}, \sigma, t)$ based on $P_{T_k,t}$
5. Generate raw detections $D_{j,t}$ from scores by applying thresholds $\theta_{\text{lin}}, \theta_{\text{RBF}}$
6. Apply NMS to obtain filtered detection results $\tilde{D}_{j,t}$
7. Feed raw detections $D_{j,t}$ to LRT tracker if scores exceed θ_s^{tr}
8. Update tracks

Algorithm *System* **II***:* Input: Image at time t, Output: detections $\tilde{D}_{j,t}$

1. Apply linear HOG detector to image, giving scores $s^{\text{lin}}(\vec{x}, \sigma, t)$
2. **for** $k \leftarrow 0$ **to** number of tracks $K - 1$
3. **do** Predict current particles for each track T_k
4. Apply DAPs to scores based on particles of each track T_k
5. Generate raw detections $D_{j,t}$ from scores $s^{\text{lin}}(\vec{x}, \sigma, t)$ by applying threshold θ_{lin}
6. Apply NMS to obtain filtered detection results $\tilde{D}_{j,t}$
7. Feed filtered detections $\tilde{D}_{j,t}$ to PHD tracker if scores exceed θ_s^{tr}
8. Update tracks

3. Experiments

Evaluation. To easily obtain test data to estimate the effect of attention priors, we recorded a set of 11 outdoor videos recorded from a static car, during daytime, on a parking lot in California. In these monochrome videos of resolution 800x600, only a single pedestrian is ever visible in front of various and potentially complex backgrounds containing vehicles, trees and other distractor structures. We manually generated annotations for each image in these sequences in the form of bounding rectangles which completely contain any visible, non-occluded pedestrians. These rectangles are not tight around the occurring pedestrians but have a certain variability as we used a semi-automatic procedure for generating them. In addition, we manually

Figure 4: Example images from evaluation streams. Background and pedestrian identity and clothing vary strongly between video streams.

stream	0	1	2	3	4	5	6	7	8	9	10
images	814	1023	751	742	698	1479	720	558	982	1011	988
dir.changes	5	6	5	5	7	7	6	5	5	7	6

Table 3: Videos used for testing. There are 11 sequences containing a single pedestrian. The total number images is 9766, which gives a total length of 16 minutes at a frame rate of 10Hz.

annotated direction changes of pedestrians in the form of intervals that start 20 images before the onset of a direction change and that end 20 images after its completion. Fig. 4 show example images taken from these videos. Evaluations are performed on the full set of videos described in Sec. 3. We always compare the *feedback condition*, i.e., the system with dynamic attention priors, to the *bottom-up condition* where dynamic attention priors are turned off by setting $A^{\text{dap}} = 0$. By varying the detection threshold $\theta^{\text{det}}_{\text{lin}}$ (plus, at the same time, the detection threshold $\theta^{\text{det}}_{\text{rbf}}$ for system **I**), we obtain ROC-like plots for all videos; These plots represent detection performance at different trade-offs between the aims of finding all pedestrians and avoiding incorrect detections. For less-than-perfect detectors these are often conflicting aims and a ROC-like plot helps to identify acceptable compromises.

3.1. Preliminary experiment: feasibility

The first experiment, conducted in the bottom-up condition (see above) investigates whether the use of dynamic attention priors as described in Sec. 2.5 is feasible, and what parameters might be appropriate. We use system **I** for this purpose as described in Sec. 2.6. To this effect, we com-

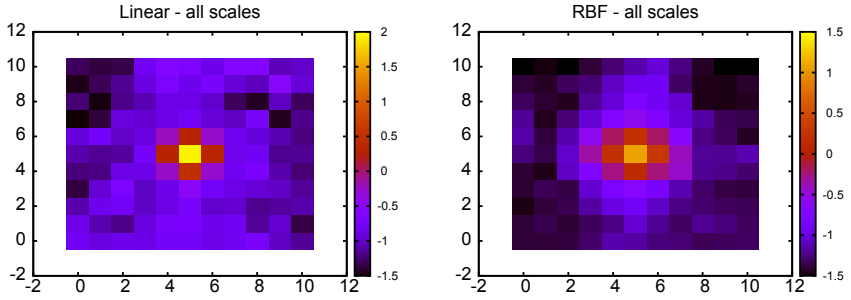

Figure 5: Dense 11x11 matrices of (unmodified) detection likelihoods around local maxima caused by real pedestrians, averaged over all videos and scales. Shown are likelihoods derived from linear (left) and RBF (right) detector of system **I**. In both cases, a sharp drop away from the local maximum at 5/5 is evident, especially for the RBF case. This is very convenient as a moderate misplacement of attentional modulation will not result in additional detections.

pute the local score profile of pedestrian detections, both for linear and RBF scores. For each time step, we first determine the position and scale of the strongest linear or RBF score:

$$\vec{x}^{(0)}, \sigma^{(0)} = \arg\max_{\vec{x},\sigma} s_{\text{lin}}(\vec{x}, \sigma, t) \tag{8}$$

Centered around this score, which we assume to indicate a pedestrian detection, we determine the $11x11$ grid of linear/RBF scores on the same spatial scale. We then calculate the average values of this grid separately for each spatial scale σ, and over all time steps (=images) in the evaluation sequences described in 3.

This experiment addresses the first message of Sec. 1.3: the basic feasibility and efficiency of DAPs. As we wish that modulation by dynamic attention priors should be as strong as possible in order to maximally enhance detections, but at the same time that it should not introduce spurious detections, it is necessary for the scores to fall off sharply around a detected pedestrian. The more pronounced this decay is, the stronger and broader the modulating signal can become without causing spurious detections. The results are shown in Fig. 5. From Fig. 5, it is evident that the structure of our detection system is well-suited for attentional modulation, as the distribution of scores around their maxima is indeed strongly peaked. we therefore conclude that a broad and strong modulatory signal can be applied. The

strongly peaked distribution of scores prevents the creation of spurious detections in this case, as score values drop quickly to a point where they will not be sufficiently enhanced even by strong modulation. The use of broad and strong modulation is favorable since strong modulation can maximally enhance (correct) sub-threshold detections, while broad modulation allows considerable deviations from the center of the modulation.

Furthermore, we measure execution time of the sliding-window stage of system **I**, see Sec. 2.1, to the baseline condition. Time measurements are in both cases averaged over a whole video stream of 500 images. We find that execution time in the baseline condition is 100 μs whereas it is only 3% higher in the top-down condition. Experiments were conducted on a 2.5GHz desktop computer with four cores and a CUDA GPU. For system **II** a similar 2% increase in computation time is observed on the same computer although of cause absolute frame rates are much lower as we use no GPU acceleration for this system. This computational efficiency is a consequence of the locality of DAPs around each track, which allows to restrict the computation of eqns.(6) and (7) to a local neighbourhood of tracks.

Summing up, we find that the modification of detection scores by DAPs does not at all impair computational performance. Together with benign behavior of detection likelihoods which favors broad and strong modulation, this allows the conclusion that the application of DAPs is feasible both in principle and w.r.t. execution time.

3.2. Improvement of detection performance

This experiment addresses the second message of Sec. 1.3, quantifying the benefit of dynamic attention priors in terms of a ROC analysis. We perform this analysis for both presented systems, see Sec. 2.6. As suggested by a comprehensive study on pedestrian detection[33], evaluation is performed by means of ROC-like diagrams giving the percentage of missed objects as a function of the number of incorrect detections per image. This is the only meaningful way to evaluate detection experiments; more well-known measures like precision or F-scores require knowledge of the number of negative "objects" in an image, which is not well defined. We present a stream-by-stream analysis for all evaluation streams described earlier in this section, using the parameters given in Tabs. 1 and 2. A more in-depth discussion of choosing these parameters is conducted in Sec. 4.1.

Evaluation is performed on detection results which are subjected to non-maxima suppression as described in Sec. 2.2. The linear and non-linear

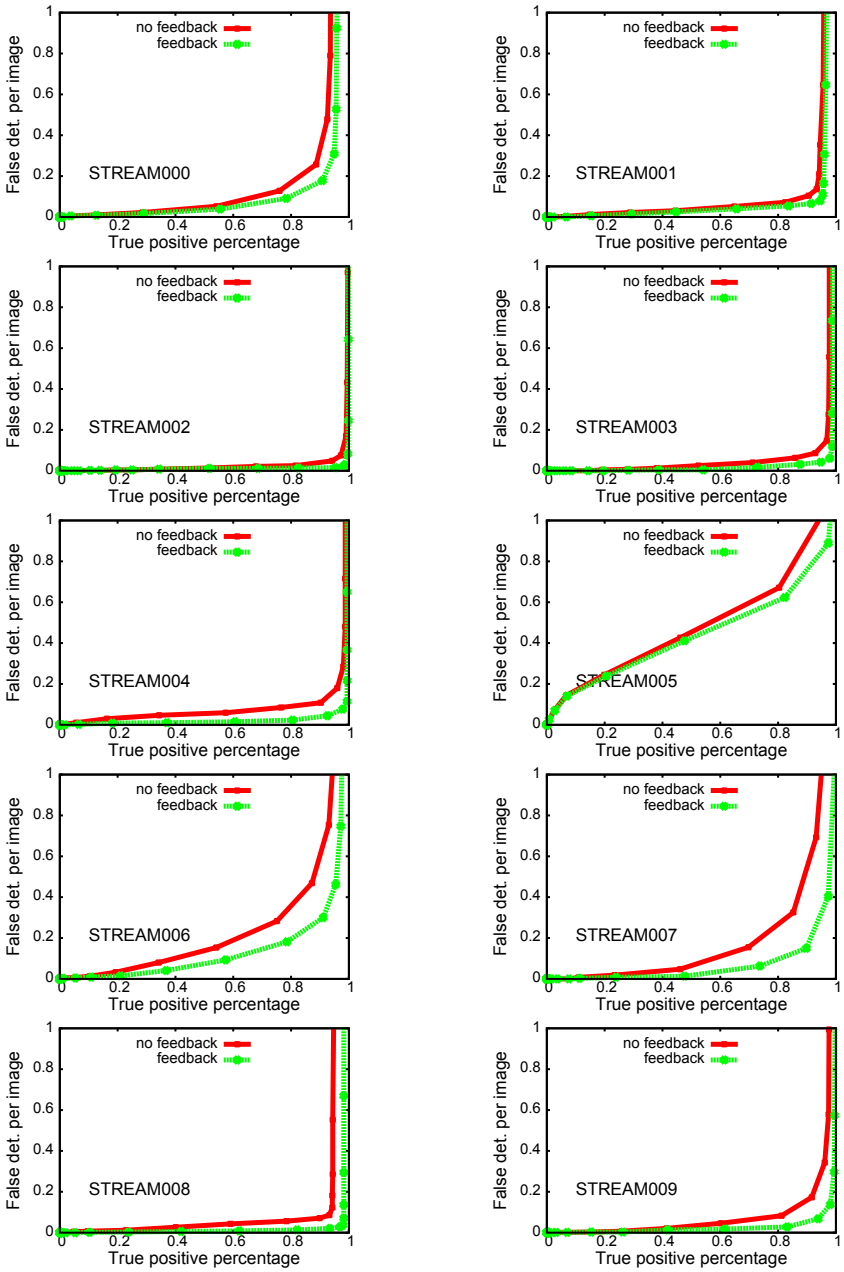

Figure 6: Comparing baseline and top-down condition for system **I** by ROC-like plots. Red solid curves show the baseline detection performance without dynamic attention priors, green dashed curves show the top-down performance. Hint: A ROC-like plot is "better" than another one if it is consistently below the other.

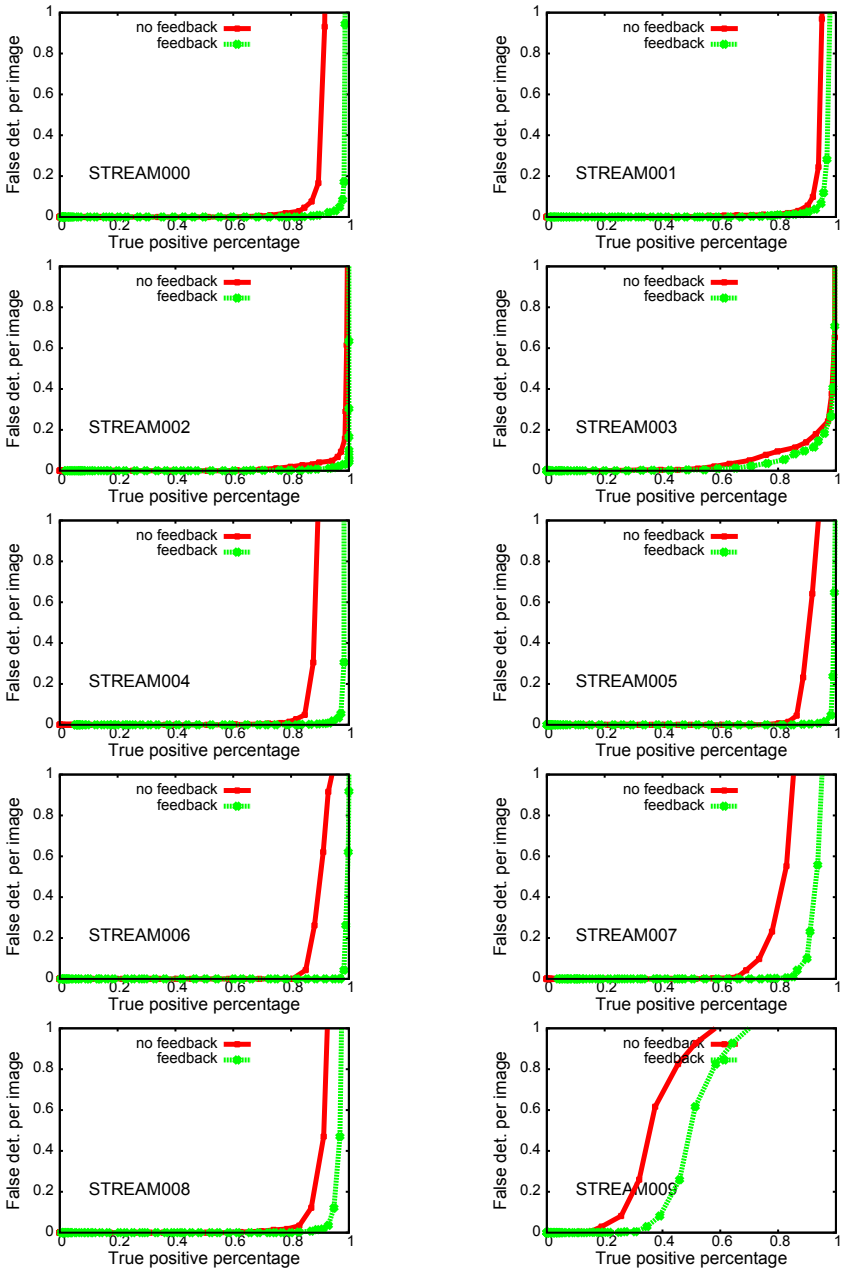

Figure 7: Comparing baseline and top-down condition for system **II** by ROC-like plots. Red solid curves show the baseline detection performance without dynamic attention priors, green dashed curves show the top-down performance. Hint: A ROC-like plot is "better" than another one if it is consistently below the other.

detection thresholds $\theta_{\text{lin}}^{\text{det}}$, $\theta_{\text{rbf}}^{\text{det}}$ are synchronously varied in a range of [-5,5]; incorrect detections/missed pedestrians are counted for each threshold setting by a comparison to the annotations described in Sec. 3. Following the evaluation procedure of [20], we use the overlap measure $o(P, Q)$ introduced in Sec. 2.2 for comparing detected and annotated objects. The results for individual videos are given in Fig. 6 for system **I**, and in Fig. 7 for system **II**. They show, for both systems, that rather broad DAPs, as suggested by the preliminary experiments, indeed improve overall detection performance considerably. This result is a relevant one as it is obtained from a considerable total video length comprising several pedestrian backgrounds and illumination conditions. Whats is more, the pedestrians in the test videos change direction quite often, so DAPs are often applied at incorrect positions. If overall top-down performance is still superior compared to the baseline condition, then either incorrect DAPs do not normally cause incorrect detections, or this performance loss is offset by significant performance gains elsewhere.

As a last point, we observe that the performance of the two systems in the baseline condition is not identical but comparable. The improvement obtained by DAPs it significant in both cases, and even slightly more pronounced for system **II**.

3.3. Behavior under violation of motion model

Relying on system **I**, see Sec. 2.6, this experiment addresses the third message of Sec. 1.3. We want to show that detection performance is not impaired by a violation of the simple linear motion models used to predict pedestrian positions, and thus to apply DAPs. To this end, we quantify the detection performance of our system, in the top-down condition, in and around abrupt direction changes of pedestrians, basically repeating the steps of the previous section, see Sec. 3.2. The only difference is that evaluation is restricted to intervals around a direction change which were manually annotated as detailed in Sec. 3. We intend to show that the performance in the top-down condition does not drop below baseline condition even though DAPs are predominantly in the wrong place. The results of this experiment are visualized in Fig. 8 to allow an easy comparison to baseline and top-down conditions.

It is clearly shown that performance does not degrade on average w.r.t. the baseline condition when DAPs are applied predominantly in the wrong place. We may speculate that this is partly due to the local center-surround structure of detection scores that was discussed before, and due to the broad

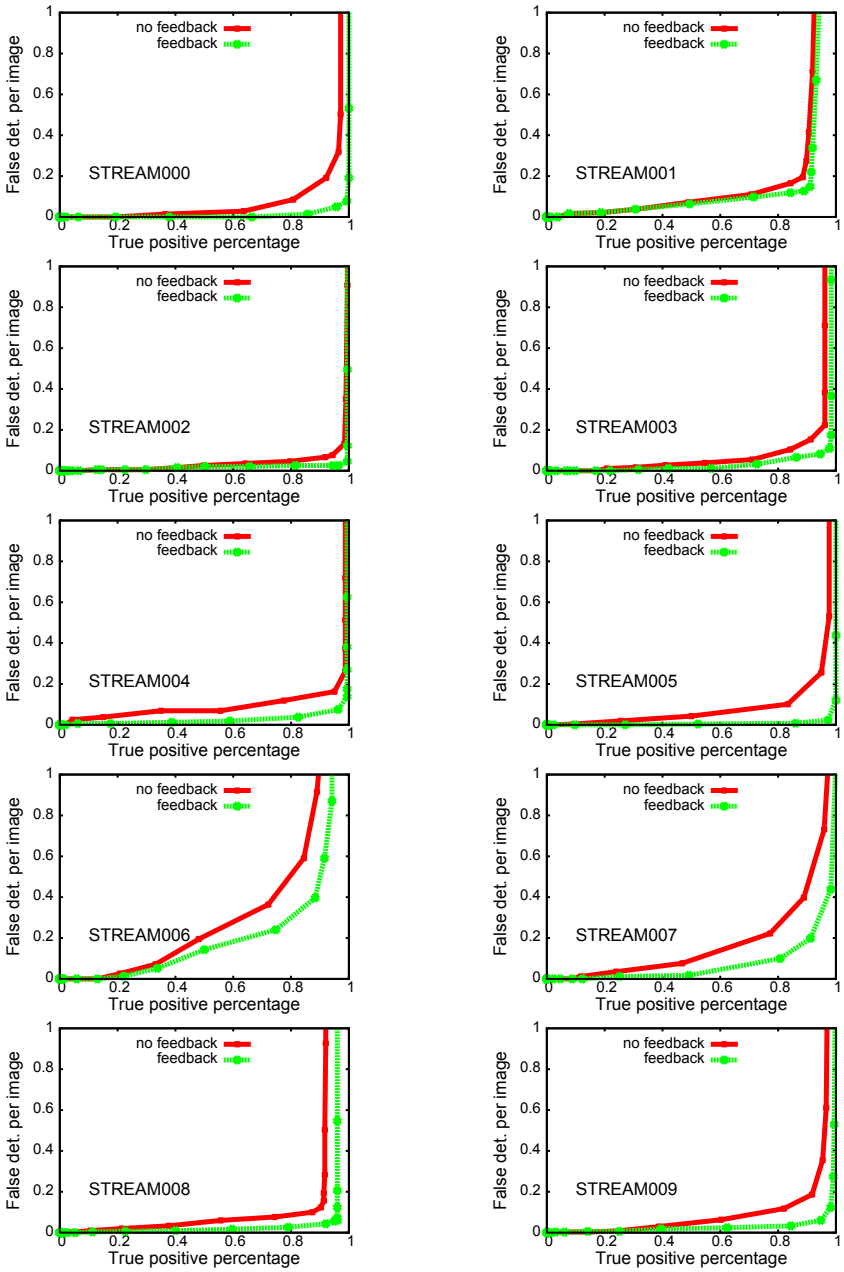

Figure 8: Comparing baseline and top-down condition for system **I** by ROC-like plots in situations where the motion model is violated, i.e., when pedestrians change direction. Red solid curves show the baseline detection performance without dynamic attention priors, green dashed curves show the top-down performance. Hint: A ROC-like plot is "better" than another one if it is consistently below the other.

DAPs used: if the DAP is only slightly off the correct pedestrian position, the Gaussian will still enhance the correct detection score. This fact justifies, in hindsight, the choice of simple linear trajectory models for prediction purposes. As the basic motion model for the particle-based PHD filter is linear as well, we expect these results to generalize to system **II** as well.

4. Discussions

In this section, we will first discuss the experimental results and judge the validity and significance of the presented work. Subsequently, we will critically examine, and ultimately justify our experimental procedure in order to forestall potential criticisms. Lastly, we will explain, in qualitative terms, the reasons for the validity of the approach based on illustrations taken from the experiments.

4.1. Choice and influence of parameters

In this section, we wish to describe the most important parameter variations and their effects. In any complex processing system, there are many parameters that can be tuned, and it is not at all clear in the beginning which are the ones that are responsible for good (or bad) performance. As we conduct an investigation that evaluates two different systems, composed of subsystems which differ as well, we restrict this discussion to the parameters relevant to the DAP mechanism and assume that individual component methods (detection, tracking) have been optimally parametrized "by hand" for the given detection task. The most relevant parameters for DAP operation are the threshold θ_s^{track} which controls which detections are allowed to contribute to tracking, the coefficient A^{dap} which governs the overall strength of DAPs, the standard deviation σ^{dap} which determines the "broadness" of DAPs, and lastly the offset Δ^{dap} which determines the lower boundary of scores than can be excited by DAPs. Other parameters influence the performance of DAPs as well, namely the tracking timescale T^{tr} for system **I** or its equivalent, the resampling coefficient ρ^{tr} for system **II**.

We set T^{tr}, or alternatively ρ^{tr} such that tracks adapt rather slowly to changed motion models. This is in any case a necessity as detections exhibit considerable fluctuation, and thus a single detection inconsistent with the current track should not have a significant effect. On the other hand, it is a goal of this study to show that DAPs work well even when the model is currently inconsistent with object motion.

In both systems **I** and **II**, a good value of θ_s^{track} is crucial to DAP function. If it is set too low, tracks can be initiated by spurious detections, which will then be reinforced by DAPs causing self-stabilized, "immortal" false detections. Therefore, this threshold needs to have a sufficiently high value in order to eliminate spurious detections, or at least to limit their frequency such that created tracks die immediately after their creation. Ultimately, a good value for θ_s^{track} therefore depends on the statistics of scores generated by the underlying visual detector.

Similarly, the offset Δ^{dap} depends on the statistics of detection scores. It should be set such that the lowest observed score caused by a real pedestrian pattern can be boosted beyond the actually detection threshold by DAPs. This implies a dependency of this parameter on the overall strength of DAPs, A^{dap}. In practice, we fix Δ^{dap} to be roughly 1.0 below the smallest observed pedestrian score, and then calculate A^{dap} such that this score just reaches the detection threshold. For excessively strong values of A^{dap}, we often observed a *locking* behavior where an object, once detected, would remain detected irrespective of image content due to strong attentional modulation. Such an object would be counted as a detection, be passed to the tracker and thus reinforce its own position and existence, regardless of detection likelihoods.

Lastly, the spatial scale of DAPs is a crucial parameter that can totally change the system's behavior. We set it to be broad, that is half a pedestrian size (multiplied by the spatial scale DAPs are applied) as the preliminary experiments (sec. 3.1) indicate that broad DAPs are very unlikely to create spurious detections away from pedestrians.

An overlap threshold of $\theta^{\text{nms}} = 0.25$ is chosen due to the less-than-optimal quality of the annotations which do not tightly encompass pedestrians.

4.2. Novelty and significance of results

This article proposes the concept of dynamic attention priors, that is to say, attentional modulation derived from predictions based on dynamic quantities such as moving objects. DAPs are a new aspect of visual attention that is inspired by very recent psychophysical findings[23]. We consider it significant that it is possible to directly transfer such insights into a technical implementation, leading to a marked performance improvement in a difficult visual detection task.

What is more, the technical realization of DAPs is very light-weight in terms of computation time and can thus be applied even in systems working

under real-time constraints. DAPs can even be put "on top" of an existing detection system (in fact this is what we did) without changing system structure at all, as long as there is a topographic representation of continuous detection scores. After modifying those scores using DAPs, the normal detection procedure, i.e. the search for local score maxima, can go its course without modification, as well as any subsequent operations such as non-maxima suppression.

Another very significant aspect of DAPs is that they do not impair performance when they are temporarily incorrect, which suggests that simple prediction models for choosing DAPs are sufficient. As the movement of, e.g., humans is a very complex thing to predict, some robustness against prediction errors is imperative, since even complex prediction models will be incorrect from time to time. Indeed, in this article we chose the simplest models possible which just perform a linear prediction of trajectories, with excellent results.

As a last point, we wish to emphasize that the vision systems we presented are is in good accordance with the notion of biological visual perception approximating a probabilistic inference process. To be sure, our detection system does not produce probabilities but detection scores which are neither normalized nor calibrated. However, empirical work on the probabilistic interpretation of SVM outputs [35] suggests that these scores are approximately related to probabilities by a simple monotonous transformation. This technique was not used in the described system for performance reasons, leading to the rather complicated expressions (6) and (7) for DAPs, instead of simply using a normalized sum-of-Gaussians probability distribution. However, the essential operation performed by DAPs is still probabilistic inference: the combination of data-driven likelihoods (the detection scores) with a priori knowledge in the form of DAPs, forming an a posteriori distribution (the modulated scores) that allows a better estimation of object positions. In this respect, our system resembles biological systems which also do not represent probabilities in a direct form, but whose mechanism of attentional feedback to lower hierarchy layers nevertheless seem to approximate probabilistic inference.

4.3. Critical points

The first and most obvious criticism is that the considered evaluation sequences are "too easy", given that even the results in the baseline condition,

see Fig. 6, are way better than results reported on standard public benchmark databases[33]. It is definitively correct that the chosen sequences are rather simple as there are no occluded pedestrians nor groups of pedestrians that could cause detection issues. The backgrounds are rather simple as well although there is a large amount of potential distractor objects such as poles, trees, etc, and the pedestrians often move before this structured background. The evaluation sequences were chosen precisely for their simplicity, as with more complex sequences it would have been difficult to attribute any performance improvements to the effect of DAPs. In addition, poses and direction changes of pedestrians are very clearly defined in the chosen evaluation sequences which is what we needed for an unambiguous analysis. We do not believe that the simplicity of the evaluations is responsible for the strong positive effect of DAPs. Rather the reverse is the case: it is much easier to improve mediocre detection results than those which are already quite good, as in the present case. On the whole, we therefore believe that on difficult sequences, the performance differences due to DAPs will be even more pronounced. This is part of ongoing work.

Another quite obvious criticism is that we did not prove the effects of DAPs using another detection system than HOG+SVM. We chose HOG+SVM as it offers the best compromise between speed and performance for real-time pedestrian detection[33]. However, we have published several results on the use of static attentional modulation for vehicle detection[20, 18, 19] using the detection system of [40]. In all of these works, we could report strong performance gains due to attentional modulation, which rather underlines the fact that many detection architectures may be coupled to attentional modulation as long as continuous detection scores are topologically organized.

4.4. Mechanisms and limitations of DAPs

In addition to the quantitative experiments of the previous section, here we want to elucidate under which conditions DAPs can (or cannot) improve detection performance, and why they do not normally cause problems under violations of the linear prediction model. For the question why DAPs work, we refer to Fig. 9. Here one can see that DAPs have a beneficial effect in two cases. First of all, this is the case when the pedestrian is occluded or before strongly structured background which can confuse the gradient-based detection/classification system. Secondly, DAPs play a role when the pedestrian exhibits a pose not well recognized by the detection SVM, normally due to insufficient training data for this pose. This will cause intermittent

Figure 9: Visualizing the reduction of false detections by dynamic attention priors on three key scenes. Each panel shows the result of detection in the feedback (left) and bottom-up(right) conditions. Red boxes indicate detections; in the absence of detections, white boxes visualize the prediction for the current moment. White or blue crosses (color chosen for best visibility depending on image background) indicate the center point of the prediction. 1) Enhancement of partially occluded pedestrian 2) Enhancement of pedestrian "in-between" pose which does not often occur in training data 3) enhancement of pedestrian before strongly structures background.

failures to detect a pedestrian, leading to a "flicker" type of detection performance. Although detection scores will be significantly higher than average when pedestrians are missed in this way, they will not be high enough to "survive" the detection thresholds. Here, DAPs ensure that such scores are enhanced just sufficiently to exceed the respective thresholds.

The downside here is that DAPs will also enhance detections that are consistently wrong, as shown in Fig. 10. In such cases, DAPs will actually increase the number of false detections which is clearly undesirable. However it should be stressed that this happens only for false detections that are consistently in the same place for a sufficient time to be tracked, which excludes the usual spurious detections exhibited by any detector.

As for the question why a violation of the linear motion model does not usually increase error rates: the basic reason is, of course, that attentional modulation will have no effect if detection scores do not at least attain a value of Δ^{dap}. Of course confusions could arise in scenes with many pedestrian-like distractor objects. However, in order for this to happen, a distractor object would have to be in close proximity to the pedestrian and additionally in the place where a wrong prediction is applied, which is not extremely probable

Figure 10: A particular situation where DAPs can actually increase the number of incorrect detections. Shown are two consecutive images from stream 5, see Tab. 3, where an incorrect detection occurs in a spatiotemporally consistent way (non-linear SVM score is shown along with the detection). This will lead to self-enhancement, and therefore to further detections and further tracking. Such undesirable behavior will continue as long as there is a sufficient frequency of incorrect but consistent detections to allow tracking to continue.

although it clearly could occur.

5. Conclusion and future works

In this article, we have demonstrated the "translation" of dynamic and predictive visual attention, as observed in humans, into two variants of an object detection system, and shown that this improves the very complex task of visual pedestrian detection in a simple benchmark. We chose a level of abstraction that permits an efficient implementation while taking care to preserve selected aspects of biological information processing. In particular, we modeled how highly dynamic quantities such as moving pedestrians can give rise to predictive attention priors, and we investigated the effect of incorrect predictions. Due to the combination of signal-driven pedestrian likelihoods and prediction-based attention priors, our model approximates a Bayesian inference process which is considered to be a key ingredient in human environment perception[44]. It is this property that effectively ensures robustness to incorrect prediction models, because the final percepts are always composed from likelihood and attention prior taken together.

Future work will concern the learning of more advanced prediction models that incorporate the pose, i.e., the orientation in space, of detected pedestrians. It is immediately obvious that this information can give valuable hints about a pedestrian's imminent actions, and vice versa a pedestrian's actions can give hints of what a good definition of appearance-based pose categories might be. Furthermore, we will integrate the ego-motion of the observer into prediction models, making them effectively non-linear. This is in accordance with [23] where the use of very advanced prediction models for object trajectories was demonstrated in humans. Evidently, predicting a pedestrian's position in the face of strong ego-motion is a challenge, but as the results of this article clearly show, prediction models do not need to be perfectly accurate to be useful. Indeed, approximate models might be quite easy to formulate, maybe with the help of implicit perspective models or 3D information from stereo vision/laser sensors.

6. Acknowledgements

Michael Garcia Ortiz gratefully acknowledges support from Honda Research Institute USA, Inc. Egor Sattarov gratefully acknowledges support from Digiteo Foundation.

7. Bibliography

[1] R. Benenson, M. Mathias, R. Timofte, and L. Van Gool. Pedestrian detection at 100 frames per second. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 2903–2910. IEEE, 2012.

[2] G. Bradski and A. Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Incorporated, 2008.

[3] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. Van Gool. Online multiperson tracking-by-detection from a single, uncalibrated camera. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(9):1820–1833, 2011.

[4] C. Chang and C. Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[5] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 886–893. IEEE, 2005.

[6] G. Deco and E. T. Rolls. A neurodynamical cortical model of visual attention and invariant object recognition. *Vision Res*, 44(6):621–642, Mar 2004.

[7] B. Dittes, M. Heracles, T. Michalke, R. Kastner, A. Gepperth, J. Fritsch, and C. Goerick. A hierarchical system integration approach with application to visual scene exploration for driver assistance. In *ICVS*, 2009.

[8] P. Dollár, S. Belongie, and P. Perona. The fastest pedestrian detector in the west. In *British Machine Vision Conference (BMVC)*, 2010.

[9] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. Gavrila. Multi-cue pedestrian classification with partial occlusion handling. pages 990–997. IEEE, 2010.

[10] M. Enzweiler and D. Gavrila. Monocular pedestrian detection: Survey and experiments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2179–2195, 2009.

[11] M. Enzweiler and D. Gavrila. Integrated pedestrian classification and orientation estimation. In *Computer Vision and Pattern Recognition (CVPR)*, pages 982–989. IEEE, 2010.

[12] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.

[13] S. Frintrop, G. Backer, and E. Rome. Goal-directed search with a top-down modulated computational attention system. In *Pattern Recognition*, Lecture Notes in Computer Science. Springer, 2005.

[14] C. G, J. Meguro, Y. Kojima, and T. Naito. Detection of pedestrians in road context for intelligent vehicles and advanced driver assistance systems. In *IEEE International Symposium on Intelligent Vehicles (IV)*, 2013.

[15] T. Gandhi and M. M. Trivedi. Pedestrian protection systems: Issues, survey, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 8(3):413–430, 2007.

[16] Á. García-Martín and J. M. Martínez. On collaborative people detection and tracking in complex scenarios. *Image and Vision Computing*, 30(4):345–354, 2012.

[17] M. Garcia Ortiz, A. Gepperth, and B. Heisele. Real-time pedestrian detection and pose classification on a GPU. In *16th International IEEE Conference on Intelligent Transportation Systems(ITSC)*, 2013.

[18] A. Gepperth. Efficient online bootstrapping of representations. *Neural Networks*, 2012.

[19] A. Gepperth, B. Dittes, and M. Garcia Ortiz. The contribution of context information: a case study of object recognition in an intelligent car. *Neurocomputing*, 2012.

[20] A. Gepperth, S. Rebhan, S. Hasler, and J. Fritsch. Biased competition in visual processing hierarchies: A learning approach using multiple cues. *Cognitive Computation*, 3(1):146–166, 2011.

[21] F. H. Hamker. A dynamic model of how feature cues guide spatial attention. *Vision Res*, 44(5):501–521, Mar 2004.

[22] F. H. Hamker. Modeling feature-based attention as an active top-down inference process. *Biosystems*, 86(1-3):91–99, 2006.

[23] M. M. Hayhoe, T. McKinney, K. Chajka, and J. B. Pelz. Predictive eye movements in natural vision. *Experimental brain research*, pages 1–12, 2012.

[24] L. Itti and C. Koch. Computational modeling of visual attention. *Nature Reviews Neuroscience*, 2(3):194–203, Mar 2001.

[25] B. Leibe, N. Cornelis, K. Cornelis, and L. Van Gool. Dynamic 3d scene analysis from a moving vehicle. 2007.

[26] E. Maggio, E. Piccardo, C. Regazzoni, and A. Cavallaro. Particle phd filtering for multi-target visual tracking. In *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, volume 1, pages I–1101–I–1104, April 2007.

[27] T. Michalke, A. Gepperth, M. Schneider, J. Fritsch, and C. Goerick. Towards a human-like vision system for resource-constrained intelligent cars. In *The 5th Int. Conf. on Computer Vision Systems Conference*. Universit.tsbibliothek Bielefeld, 2007.

[28] T. Michalke, A. Gepperth, M. Schneider, J. Fritsch, and C. Goerick. Towards a human-like vision system for resource-constrained intelligent cars. In *The 5th International Conference on Computer Vision Systems*, 2007.

[29] D. Musicki and B. La Scala. Multi-target tracking in clutter without measurement assignment. *Aerospace and Electronic Systems, IEEE Transactions on*, 44(3):877–896, 2008.

[30] M.W. and Spratling. Predictive coding as a model of biased competition in visual attention. *Vision Research*, 48(12):1391 – 1408, 2008.

[31] V. Navalpakkam and L. Itti. An integrated model of top-down and bottom-up attention for optimal object detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2049–2056, New York, NY, Jun 2006.

[32] B. S. P. Dollár, C. Wojek and P. Perona. Pedestrian detection: A benchmark. In *Computer Vision and Pattern Recognition (CVPR)*, June 2009.

[33] B. S. P. Dollar, C. Wojek and P. Perona. Pedestrian detection: An evaluation of the state of the art. 2011.

[34] R. Perko and A. Leonardis. A framework for visual-context-aware object detection in still images. *Computer Vision and Image Understanding*, 114(6):700–711, 2010.

[35] J. Platt et al. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74, 1999.

[36] M. Ristin, J. Gall, and L. Van Gool. Local context priors for object proposal generation. In *Computer Vision–ACCV 2012*, pages 57–70. Springer, 2013.

[37] J. Schmuedderich, N. Einecke, S. Hasler, A. Gepperth, B. Bolder, R. Kastner, M. Franzius, S. Rebhan, B. Dittes, H. Wersing, J. Eggert, J. Fritsch, and C. Goerick. System approach for multi-purpose representations of traffic scene elements. In *International IEEE Annual Conference on Intelligent Transportation Systems*, 2010.

[38] D. Simon. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*. Wiley-Interscience, 2006.

[39] A. Torralba. Contextual priming for object detection. *IJCV*, 53:2003, 2003.

[40] H. Wersing and E. Körner. Learning optimized features for hierarchical models of invariant object recognition. *Neural Computation*, 15(7), 2003.

[41] C. Wojek, S. Walk, S. Roth, K. Schindler, and B. Schiele. Monocular visual scene understanding: Understanding multi-object traffic scenes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(4):882–897, 2013.

[42] J. Wolfe. Guided search 2.0: a revised model of visual search. *Psychonom. Bull. Rev.*, 1:202–238, 1994.

[43] J. Yu, D. Farin, and B. Schiele. Multi-target tracking in crowded scenes. In *Pattern Recognition*, pages 406–415. Springer, 2011.

[44] A. Yuille and D. Kersten. Vision as bayesian inference: analysis by synthesis? *Trends in cognitive sciences*, 10(7):301–308, 2006.