Biased competition in visual processing hierarchies: a learning approach using multiple cues

Alexander R.T. Gepperth Sven Rebhan Stephan Hasler
Jannik Fritsch

November 15, 2010

Abstract

In this contribution we present a large-scale hierarchical system for object detection fusing bottom-up (signal-driven) processing results with top-down (model or task-driven) attentional modulation. Specifically, we focus on the question of how the autonomous learning of invariant models can be embedded into a performing system, and how such models can be used to define object-specific attentional modulation signals.

Our system implements bi-directional data flow in a processing hierarchy. The bottom-up data flow proceeds from a *preprocessing level* to the *hypothesis level* where object hypotheses created by exhaustive object detection algorithms are represented in a roughly retinotopic way. A competitive selection mechanism is used to determine the most confident hypotheses, which are used on the *system level* to train multimodal models that link object identity to invariant hypothesis properties.

The top-down data flow originates at the system level, where the trained multimodal models are used to obtain space- and feature-based attentional modulation signals, providing biases for the competitive selection process at the hypothesis level. This results in object-specific hypothesis facilitation/suppression in certain image regions which we show to be applicable to different object detection mechanisms.

In order to demonstrate the benefits of this approach, we apply the system to the detection of cars in a variety of challenging traffic videos. Evaluating our approach on approximately 3500 annotated video images from more than 1h of driving, we can show strong increases in performance and generalization as compared to object detection in isolation. Furthermore, we compare our results to a late hypothesis rejection approach, showing that early coupling of top-down and bottom-up information is a favorable approach especially when processing resources are constrained.

1 Introduction

Visual processing in the human neocortex is organized in a hierarchical fashion: neurons in lower levels such as LGN and V1 and A1 have small receptive fields and are sensitive to a very specific set of stimuli, whereas neurons in higher areas tend to have larger receptive fields and are increasingly broad in their selectivity[31]. As a consequence, neural activity in lower hierarchy levels is tightly coupled to sensory input whereas higher-level neurons may well respond to rather abstract categories and concepts[31]. It has long been known that information processing in such hierarchies is bi-directional, consisting of a bottom-up (away from sensory input) and a top-down (towards sensory input) component[12, 17], and this has been linked to accounts of *attentional modulation*, i.e., the selective and large-scale enhancing or suppressing of neuronal responses in accordance with task demands[14, 22, 32]. For visual processing, there seem to exist at least two concurrently active mechanisms of attentional modulation: *space-based attention* which enhances certain *locations* in the visual field and *feature-based attention* which is not localized but affects all populations of neurons representing a particular visual property[11].

Since cortical neurons, especially at high hierarchy levels, compete strongly with each other for representing the current stimulus, it has been proposed that local facilitation or inhibition of neural responses by top-down signals can explain the pronounced effects of attentional modulation simply because small local biases may result in very different stable states of the competition process[4, 18, 28]. This *biased competition* [4] account of attentional modulation has influenced many models of visual attention; we incorporated it into our research because we found that competition between object hypotheses is an unavoidable step for agents with constrained resources; the "biasing" of the existing competition mechanism is a then straightforward extension.

Since attentional modulation is observed to enhance performance w.r.t. a wide variety of tasks, the question immediately arises how models for task-specific attentional modulation are obtained. An influential concept, the so-called *reverse hierarchy theory* [12] states that such models are first acquired in high levels of the processing hierarchy and subsequently used to train task-specific responses in lower levels. We present the method of *system-level learning* which implements an important aspect of reverse hierarchy theory by introducing dependency models between highly invariant quantities available on the highest level of a processing system. This is motivated by our finding that such *system-level models* usually show high generalization ability.

1.1 Motivation for the presented work

Our experience with cluttered and uncontrolled traffic environments suggests that purely appearance-based (i.e. based on local pixel patterns) object detection suffers from significant ambiguities: the more complex a scene is, the higher is the probability that some local pixel pattern will be similar to the object class of interest. In order to overcome this difficulty, we claim that object-specific models relating appearance-based visual information to non-local and non-visual information must be taken into account to achieve the required disambiguation. For convergent, hierarchically organized systems, this implies that such models can only be formed at high hierarchy levels where the required information is available. The idea of *system-level learning* (see also [8]) is to represent all quantities available at the highest hierarchy level in a common way in order to use a single, scalable learning algorithm for detecting correlations. The focus of this article is to use system-level models for generating and using *expectations* to generate attentional modulation: given a *search cue*, e.g. a certain object identity, system-level models are queried for features correlated with this identity, and the resulting expectation is used to define attentional modulation.

1.2 Research Questions, claims and messages

Based on our experience with object detection in complex traffic scenes, we formulated a number of hypotheses. which this article will investigate based on a hierarchical car detection system system as shown in Fig. 1. We evaluate the system in challenging real-world situations using extended annotated video sequences [1].

Hypothesis 1: Detection performance The goal of this article is to demonstrate that attentional modulation signals can be derived from system-level models, and that their application to lower hierarchy levels results in strongly increased performance in object detection, as well as in significant generalization ability. The beneficial effect of suitable attentional modulation has been established in previous studies[33] in simple environments, and without using learning; our goal is to show that the benefit is even more pronounced in complex outdoor situations, and that learning attentional modulation is both feasible and efficient.

Hypothesis 2: Generality We advocate the view that biased competition[4] is a common mechanism for attentional modulation in neocortical hierarchies. In order to demonstrate this particular point, we conduct experiments with a symmetry-based object detection method and show that it can be successfully controlled by attentional modulation using a common competitive selection mechanism.

Hypothesis 3: Robustness We aim to show that the fusion of modulation signals is feasible, computationally efficient and increases robustness especially in difficult environments. Although the issue of fusing multiple modulation signals has been studied in indoor settings (see, e.g., [13, 35]), our goal is to verify the benefits in a challenging outdoor scenario. In particular, we intend

[1] We will make available the videos and annotations described in this article to researchers upon email request to the first author

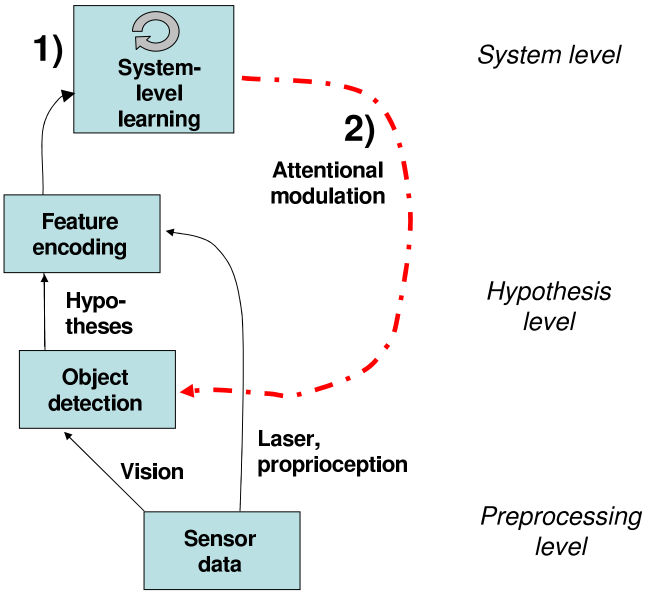


Figure 1: Illustration of the basic structure and the inherent novel points of the presented system. 1) Learning of multi-modal system-level models for generating attentional modulation *during system operation* 2) Application of system-level models for attentional modulation. What kinds of models are learned effectively depends only on the processing results that are supplied to the system-level learning mechanism.

.

to demonstrate that performance is unaffected by the inclusion (or omission) of uninformative modulation signals when using our fusion approach.

Hypothesis 4: Efficiency We hypothesize that the concept of applying attentional modulation early in a processing hierarchy is a consequence of constrained resources. We verify this by comparing the object detection performance of our system under strong resource constraints when using attentional modulation versus when using a naive high-level rejection approach.

Hypothesis 5: Bootstrapping This article aims to show that successful training of system-level models can occur using a self-generated supervision signal. Bootstrapping is a well-known and nontrivial issue (see, e.g., [21, 29]); However, a system capable of bootstrapping will be truly capable of autonomous learning in an embodied agent, which will eliminate the effort of creating supervision signals completely.

1.3 Related work

Visual attention has been subject of intense research in the recent decades, resulting in a number of theoretical models such as Guided Search 2.0[40], Selective Tuning[34] or Biased Competition[4].

A large number of computational models were proposed subsequently, which we will review in this section, focusing mainly on approaches that address learning of attentional modulation.

A strictly feature-based attention model was proposed by [16]. It focuses on feed-forward processing and lateral competition, either in the form of center-surround filtering or explicit competition mechanisms. This model was applied in numerous real-world scenarios, e.g., [15], for goal-driven scene analysis[25] or fast object detection and recognition[37]. While the work described in [25] employs high-level semantic models of object-to-object or object-to-goal relations to guide visual attention to behaviorally important locations, these models are specified by a designer and not acquired through learning. The work of [37] couples an exhaustive object detection mechanism to signal-driven saliency with beneficial results. In this approach, object-specific models enter only through training the object detection mechanism.

The coupling of object detection and contextual information mediated by low-level modulation is demonstrated in [24] where context information about the "gist", i.e., a low-dimensional description of a scene, is used to infer the locations of relevant objects in images by statistical models constructed from training examples. In this work, learning is achieved by computing statistical models about the location and size of objects depending on scene gist in an offline fashion. The concept of gist is taken further in [13] where a generic probabilistic model of 3D scene layout is proposed that can be queried for likely image locations of, e.g., cars or pedestrians in order to inform an exhaustive local object detector. This work is interesting because the images used to reason about 3D scene layout were actually monocular. Furthermore, object detection may not only be guided by global scene properties but also by other objects in the scene: in [3], a discriminative model of local object-to-object interaction is proposed that formalizes cooperation and competition between local detections of multiple object classes and gives a probabilistic interpretation of this process. Lastly, object detection may also be regarded as an active process in which the performed gaze actions (i.e., object detections) should maximize information acquisition. Based on the saliency map approach of [16], a POMDP formalism is used in [35, 36] to optimize gaze target selection based on the detections arising from previous gaze targets, visual saliency and global scene priors.

The Selective Tuning Model, originally proposed in [34], was integrated into a number of computational attention models. The focus of these models is, on the one hand, on explaining cognitive phenomena such as feature binding in cortical hierarchies[30] and, on the other hand, showing real-world capability using, e.g., visual motion as attentional cue as demonstrated in[33]. Methodically, the Selective Tuning model is a feature-based model that emphasizes the importance of lateral competition (modeled by winner-takes-all mechanisms) and top-down feedback signals. The models used to generate attentional modulation signals are not obtained by learning but chosen "by hand". Qualitative evaluation is performed on indoor scenes to validate and demonstrate the used models.

Attentional models more strongly motivated by neural processing can be found in [2, 9, 10]. All employ neural dynamics as a key ingredient with emphasis on bottom-up and top-down data flow in recurrent architectures. A key issue in [9, 10] is the interplay and fusion of bottom-up and top-down information, where the realization of biased competition by the modulating competitive neural dynamics is central to the work of [2]. Whereas the attentional effects obtained in [2] are purely feature-based, the models of [9, 10] include aspects of space-based attentional modulation as well. Evaluation is performed on still-images of indoor scenes in [9, 10] and by an analysis of single-neuron responses in [2]. Both models do not emphasize learning but employ fixed models for generating attentional modulation.

Another group of attention models focuses on feature-based, object-specific selectivity through learned search models, as well as applicability in real-world scenarios. Whereas the work of [23] focuses on car detection in road traffic scenarios, the VOCUS model[6] targets mobile robotics applications. Both approaches use an offline optimization procedure to generate feature-based object search templates based on small numbers of image patches. These templates are fused with a bottom-up attention signal similar to [16] such that both visual saliency as well as proximity to the search template may trigger object detection.

2 Methods

We present a system (see Figs. 1,2 and [7]) of significant complexity which receives inputs from a stereo camera, the vehicle-internal CAN bus and two laser range-finding sensors. It computes a list of entities that are judged to be relevant, i.e., cars and vehicles. The system is not yet running in a vehicle but receives its inputs by a timestep-based replay of recorded data, which is exactly equivalent to the way data would be received in our prototype vehicle. Since the system is not operating on "live" data, it is possible to replay annotations (e.g., positions and identities of other traffic participants) as "virtual sensors", that is, as if they were obtained from measurements.

2.1 Interfacing of system components by population coding

Population coding is a biologically inspired way of encoding information. Basic properties of population coding models[26, 41] are the representation of information on two-dimensional surfaces in analogy to cortical surfaces, and, on the other hand, the concept of storing confidence *distributions* for all represented quantities.

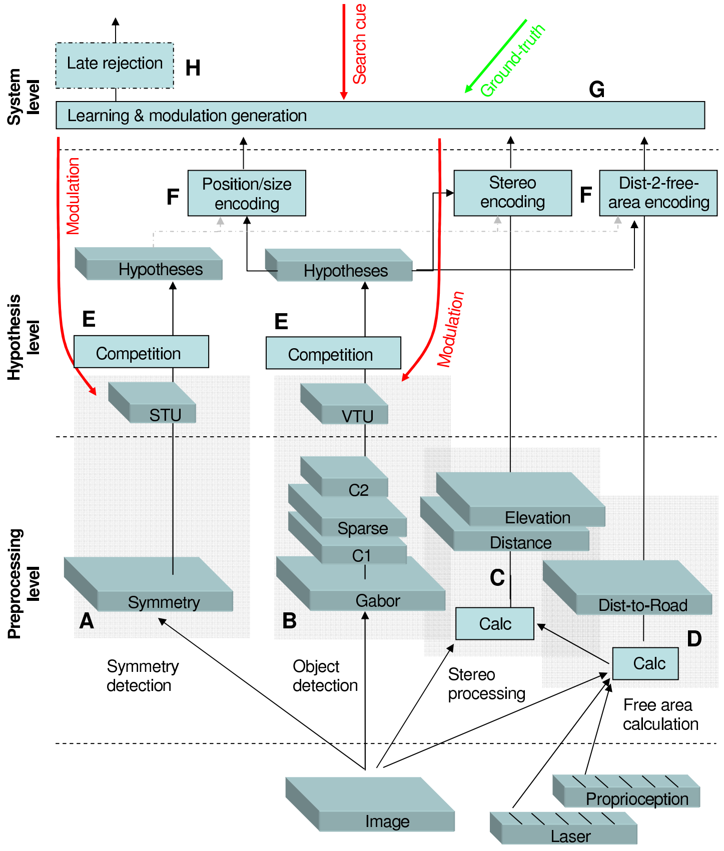


Figure 2: Global structure of the described hierarchical object detection system. Functional modules are object detection (**A,B**), stereo processing (**C**), free-area computation (**D**), competitive hypothesis selection (**E**) and population encoding (**F**). Attentional modulation is trained at the system-level **(G)**, linking hypothesis identity to elevation, distance, distance-to-free-area and 2D image position (**F**). System-level training happens in a supervised way using ”true” object identities supplied by *ground-truth data*. Given an arbitrary desired object identity (the *search cue*), attentional modulation is applied to the hypothesis level of object detection, thus favoring the detection of objects of the desired identity. Data flows from symmetry detection (**A**) to other modules are identical to data flow from the appearance-based classifier (**B**) but are not shown for clarity. For comparison, we also implemented a ”late rejection” module at system level (**H**) which uses a multilayer perceptron for directly (without influencing lower system levels) mapping population-coded quantities produced at the hypothesis level to an object identity decision.

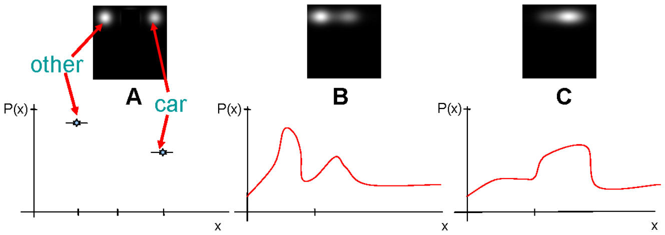


Figure 3: Transfer of different types of measurements to population codes. The particular type of measurement determines how it is translated into a population code. **A)** a discrete distribution from, e.g., an object classifier, is translated into a population code where only certain locations carry information. **B),C)** Quasi-continuous one-dimensional measurements (e.g., object elevation and distance) are encoded into population codes that are extended along one axis. Note that the uncertainty (multimodality) of measured distributions is transferred to the resulting population code. The precise way of encoding is determined on a case-by-case basis.

Mathematically, a population code is therefore a collection of one or several two-dimensional lattices, where each lattice point (”neuron”) stores a normalized confidence value corresponding to the belief that a certain property (”preferred stimulus”) associated with this lattice point is present in the encoded information. These properties link population coding closely to the Bayesian approach to probability[1]. In particular, population coding represents encoded quantities as distributions over possible values, thus implicitly storing the associated uncertainty in accordance with the ”Bayesian brain” hypothesis[19].

In order to be able to link system-level information by learning methods as described in Sec. 2.8, we convert such quantities into population codes. The system-level quantities we want to encode are confidence distributions which may be either one- or two-dimensional which we denote *source distributions*. The nature of source distributions may be spatially discrete (i.e., having nonzero confidence values only at certain positions) or continuous as well as graded (with confidences assuming values in a range between 0.0 and 1.0) or binary. Examples of different kinds of source distributions and their population encoding are shown in Fig. 3. For the actual encoding, we employ the convolution coding technique [26] using a Gaussian kernel of fixed size. In case a source distribution is one-dimensional, we embed it into a two-dimensional distribution along a specified axis before performing convolution coding.

2.2 The appearance-based classifier

The appearance-based classifier[39] generates object hypotheses in two successive steps. As a first step, it generates retinotopic confidence maps as described

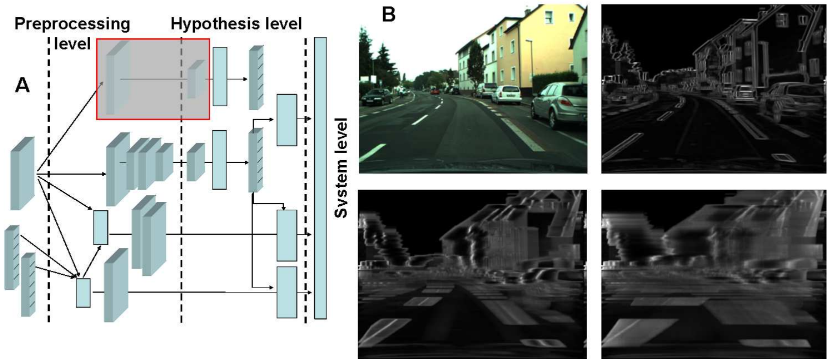


Figure 4: Performance example of symmetry-based object detection. **A)** Embedding into processing system. **B)** Input image and generated multi-scale confidence map. A total of $K = 8$ scales is used for symmetry, corresponding to filters of pixel height $h = 3$ and half-width $\frac{w}{2} = 5, 9, 13, 18, 25, 35, 49, 69$. Shown are confidence maps corresponding to filter widths 5,25 and 69.

in [38]. Each pixel of a confidence map represents the detection of a specific view of an object (in our case: back-views of cars) at a specific scale $k = 0, \ldots, K-1$. In a second step, object hypotheses are generated from the confidence maps by the competitive selection process described in Sec. 2.7. Details about processing and classifier training are given in [7].

2.3 Symmetry-based object detection

Just as the appearance-based classifier, symmetry-based object detection generates object hypotheses in two steps: first, generation of a multiscale, retinotopic confidence map and second, competitive hypothesis selection (see Sec. 2.7) based on the produced maps. Fig. 4 shows an example of a confidence map for a given input image. Details of the symmetry calculation can be found in [7].

2.4 Free-area computation

The *free area* is defined as the obstacle-free area in front of the car that is visually similar to a road. This quantity carries significant semantic information. Since it is, by construction, bounded by all obstacles that the car might collide with, many relevant obstacles are close to the boundaries of the free area. For the purposes of the presented system, the quantity of interest is therefore the distance of an object hypothesis to the free area. Details of free-area calculation and the transfer to population codes are given in [7]. Please see Fig. 5 for examples of free-area computation and the transfer of the corresponding distance-to-free-area measurement to population codes $z^1(\vec{p})$.

2.5 Distance and elevation computations

We employ dense stereo processing for measuring the distance and height of image pixels in car-centered coordinates. For obtaining hints about the identity of objects, such measurements are helpful but not optimal: It is not really the height relative to a car-centered coordinate system that carries semantic information, but rather the height over the road surface. Details about the computation of this quantity as well as stereo distance computation are given in [7]. Please see Fig. 7 for an example of the transfer of elevation (and distance) measurements to population codes $z^2(\vec{p}), z^3(\vec{p})$.

2.6 Position and size related analysis

Lastly, two important system-level quantities are ”retinal” hypothesis position and size. Even though the retinal position of objects changes, for example, during turning maneuvers (similar examples can be mentioned for retinal size), we found that these quantities can nevertheless provide useful hints about object identity. Therefore, they are encoded into population codes $z_i^0(\vec{p})$ at the hypothesis level of our system as shown in Fig. 6. Details about computation and population encoding are given in [7].

2.7 Competitive hypothesis selection

Situated on the hypothesis level of our system (see Fig. 2), competitive hypothesis selection is roughly modeled based on the way lateral inhibition operates in

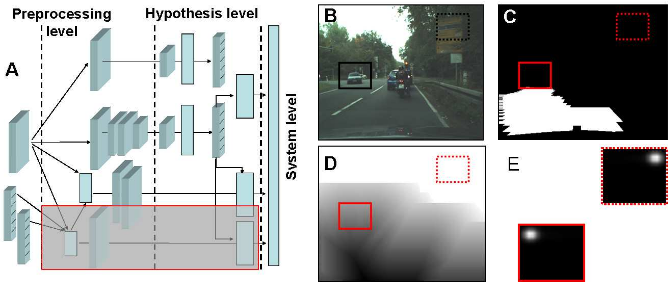


Figure 5: Performance example of free-area and distance-to-free-area computation for two object hypotheses. **A)** Embedding into processing system **B)** Video image **C)** computed free area $\tilde{F}^1(\vec{x})$ **D)** pixelwise distance-to-free area map $F^1(\vec{x})$. Each pixel value in the map is determined by that pixel's minimal distance to a computed free-area pixel. Due to computational reasons, an upper limit $d_{\max}$ is imposed. Note that distances are negative for pixels on the free area. **E)** Population codes $z^1(\vec{p})$ obtained for two different object hypotheses.

cortical surfaces. It requires a resolution pyramid of K scales containing retinotopic *confidence maps* $c_i(\vec{x}), i = 0, \ldots, K-1$ produced either by the appearance-based classifier or the symmetry detection (see Secs. 2.2, 2.3), and generates up to a desired number H of object hypotheses $h_j, j = 1, \ldots, H$. Examples of retinotopic confidence maps at several pyramid levels are shown in Figs. 8 and 4. Selection processes described in this article typically use a value of $H = 40$.

Bottom-up operation Based on the pyramid of confidence maps, $c_i(\vec{x})$, local activity maxima are detected across all scales $i \in [0, K]$. Each local maximum with index j at a position $\vec{x}^*_j$ and scale s^*_j is interpreted as a rectangular object hypothesis centered at $\vec{x}^*_j$, having a width/height determined by s^*_j. Based on the peak values $c_{s^*_j}(\vec{x}^*_j)$, the list of maxima is subjected to a thresholding operation to suppress weak hypotheses. This threshold θ, $\theta > 0$, strongly influences the number of generated hypotheses and the types of possible errors. With increasing threshold usually more objects are missed, while low thresholds lead to increased false detections. Competitive hypothesis selection works in a greedy fashion, i.e., the maximum with the highest peak value is chosen first and its position and scale $\vec{x}^*_0$, s^*_0 are used to define a surrounding region of inhibition (see Fig. 8) in all confidence maps $c_i(\vec{x})$. Maxima in inhibited regions cannot be selected any more. The remaining maxima are processed in descending order, where all hypotheses are rejected whose area intersects with an already inhibited area by more than 75%. The process stops when the desired number of hypotheses, H, is reached or no further local maxima remain. We discovered that the detection performance for cars increases when using a specific region of inhibition which is higher and less wide than the object hypothesis itself, probably because this accounts better for the typical occlusions between cars.

Integrating attentional modulation Assuming a pyramid of attentional modulation maps $m_i(\vec{x})$, $m_i(\vec{x}) \in [0, 1]\ \forall \vec{x},\ i$, attentional modulation can be applied before hypothesis selection:

$$c_i^{\text{mod}}(\vec{x}) = c_i(\vec{x}) m_i(\vec{x}) \tag{1}$$

This process provides a systematic bias to the competitive hypothesis selection by changing the relative strengths of local maxima in the confidence maps, thus realizing biased competition[4]. In effect, attentional modulation enhances or attenuates local maxima depending on their agreement with the system-level models encoded in the modulation maps. Sufficiently strong local maxima can survive even though they are attenuated by attentional modulation if they continue to exceed the selection threshold, and if there are no competing local maxima within the radius of inhibition. Examples of the ”survival” of strong local maxima can be observed in Fig. 9.

2.8 Data transmission and associative learning

We assume that positions $\vec{x}, \vec{y}$ in arbitrary population-coded neural representations A,B with activities $z^A(\vec{x}, t), z^B(\vec{y}, t)$ (see, e.g., Fig. 3) are connected by synaptic weights $w^{AB}_{\vec{x}\vec{y}}$. The transmission of information from A to B by means of learned synaptic connections $w^{AB}_{\vec{x}\vec{y}}$ is governed by a simple linear transformation rule:

$$z^B(\vec{y}, t) = \sum_{\vec{x}} w^{AB}_{\vec{x}\vec{y}}(t) z^A(\vec{x}, t). \tag{2}$$

We employ a supervised learning strategy where the supervision signal can come from annotated data or can be generated within the system (bootstrapping).

In line with our focus on simple but generic learning methods, we perform an online gradient-based optimization of (2) based on the mean squared error w.r.t the teaching signal for neurons in representation B. Given two neurons at positions $\vec{x}, \vec{y}$ with activities $z^A(\vec{x}), z^B(\vec{y})$ in two population-coded representations A,B, plus a teaching signal for representation B, $t^B(\vec{y})$, the learning rule reads

$$\begin{aligned} w^{AB}_{\vec{x}\vec{y}}(t+1) &= \epsilon z^A(\vec{x})[z^B(\vec{y}) - t^B(\vec{y})] \\ &\equiv \epsilon x(y - y^*). \end{aligned} \tag{3}$$

where $\epsilon << 1$ is a small *learning rate* constant. We used the abbreviations $x \equiv z^A(\vec{x}, t), y \equiv z^B(\vec{y}, t), y^* \equiv t^B(\vec{y}, t)$ for presynaptic and postsynaptic neurons as well as target values to obtain a more usual way of writing this learning rule.

For obtaining the *expected* activity in a population-coded representation B, we simply train weights $w^{BA}_{\vec{x}\vec{y}}$ performing the reverse mapping $B \rightarrow A$ and thus can obtain

$$e^A(\vec{x}, t) = \sum_{\vec{y}} w^{BA}_{\vec{x}\vec{y}}(t) z^B(\vec{y}, t) \tag{4}$$

2.9 System-level learning of object models

Input to the system-level, the highest hierarchy level of our system, is the set of population codes for space/feature-based hypothesis poperties $z^0_i(\vec{x}), z^{1,2,3}(\vec{x})$ as well as *ground-truth data*, i.e, information about "true" positions and identities of relevant objects obtained from annotations (see Sec. 2.13).

As shown in Fig. 10, the following steps are performed for each hypothesis: the hypothesis and the feature maps are jointly used to generate population-coded representations of hypothesis features (see Fig. 3), in this case distance, elevation, image position and distance-to-free-area. Using ground-truth data (see Fig. 2), a population-coded representation of the teaching signal for object identity is generated (see Fig. 3) depending on whether there is an annotated object containing the center pixel of the hypothesis (see [7] for details). Alternatively, the population-coded teaching signals may also be obtained from the identity estimate provided by the appearance-based classifier. The teaching signal is then used to update the mapping from population-coded hypothesis features to object identity.

We perform training using a procedure called *blocking*: we group the stream of hypotheses into intervals corresponding to 30s of real time and apply system-level model training only for odd-numbered groups. The even-numbered groups are later used for evaluation, therefore they are processed using a learning constant of $\epsilon = 0.0$. Blocking is a widely accepted procedure (see [20]) that allows us to train *and* evaluate system-level models on all streams while maintaining a high dissimilarity between training and evaluation sets (traffic scenes usually change strongly in 30s).

2.10 Generation of object-specific attentional modulation

This function is in many respects the reverse of the learning procedure shown in Fig. 10: a population-coded object identity (see Fig. 3), the *search cue*, is specified and activation is propagated *backwards* through the system-level network using Eqn.(4), the search cue and the learned reverse weight matrices. In this way, object-specific *expected feature distributions* $e^k(\vec{p})$ are obtained, again in the form of population codes.

For space-based attention, the reverse propagation produces a pyramid of K expected image position distributions $e^0_i(\vec{p}), i = 0, \ldots, K-1$ which can be upscaled using bicubic interpolation to obtain space-based modulation signals:

$$m^0_i(\vec{x}) = \text{scale}_{N,M} e^0_i(\vec{p}) \tag{5}$$

For feature-based attentional modulation, the expected feature distributions $e^k(\vec{p})$, $k = 1, 2, 3$ must first be *decoded*. Since each position $\vec{p}$ in a population-coded representation is associated with a certain feature value, the expected feature distributions can be transformed from distributions over positions, $e^k(\vec{p})$, into distributions over feature values, $\tilde{e}^k(\tilde{p}_n), n = 1, \ldots, \nu$. For feature-based attention, we set $\nu = 1$ whereas space-based attention requires $\nu = 2$ since we encode a two-dimensional image position.

Individual feature-based attentional modulation signals $m^k(\vec{x})$ can be generated by a lookup operation in retinotopic feature maps $F^k(\vec{x})$ produced by the algorithms of the preprocessing level, see Secs. 2.5, 2.4:

$$m^k(\vec{x}) = \tilde{e}^k(F^k(\vec{x})). \tag{6}$$

Up to this point, feature-based modulation maps are only selective for feature values and not to position and size of object hypotheses. In order to fuse feature- and space-based modulation signals, we first perform a separate normalization step for space- and feature-based contributions. Subsequently, we sum feature-based modulation signals and duplicate them over all pyramid scales. The final normalized multiscale modulation map $m_i(\vec{x}), i = 0, \ldots, K-1$ is obtained by

$$\begin{aligned}
\hat{m}^k(\vec{x}) &= \frac{m^k(\vec{x})}{\max_{\kappa\in\{1,2,3\},\vec{x}} m^\kappa(\vec{x})}, k \in \{1,2,3\} \\
\hat{m}_i^0(\vec{x}) &= \frac{m_i^0(\vec{x})}{\max_{i,\vec{x}} m_i^0(\vec{x})} \\
\tilde{m}_i(\vec{x}) &= \left(\sum_{\kappa=1,2,3} \hat{m}^\kappa(\vec{x})\right) + \hat{m}_i^0(\vec{x}) \\
m_i(\vec{x}) &= \frac{\tilde{m}_i(\vec{x})}{\max_{j,\vec{x}} \tilde{m}_j(\vec{x})} \qquad (7)
\end{aligned}$$

The multiscale modulation map $m_i(\vec{x})$ is used to influence competitive hypothesis selection as described in Sec. 2.7. The process of generating attentional modulation using learned system-level models is schematically shown in Fig. 11; the fusion of space- and feature-based modulation signals is separately visualized in Fig. 12.

2.11 Model training for "late rejection"

In order to perform "late rejection" of object hypotheses as envisioned in Fig. 2, a mapping from population-coded system-level quantities to object identity (likewise a system-level quantity) must be determined. In addition to the available linear system-level models, we use a multilayer perceptron (MLP) for this task which may achieve better performance due to the employed nonlinearities.

MLP training is performed in an offline fashion; we run the system without attentional modulation on stream I (see Sec. 2.13 and Fig. 13), recording the population codes generated for the first 10000 object hypotheses. Inputs to the MLP are data vectors consisting of the concatenation of all population-codes obtained from a single hypothesis, where population codes are downsampled to a size of 16x16 pixels. The dimensionality of the input space is therefore 256 x 19 = 4864, thus encompassing distance, distance-to-free-area, elevation and the 16 position features). We train the MLP using Rprop, early stopping regularization, weight decay and manual equalization of the imbalance between car and non-car examples[27]. The MLP uses a sigmoid nonlinearity and has three layers: one input, a hidden layer of size 50 and an output layer of size 1. The size of the input layer is given by the summed size of the system-level features described in Sec. 2.8. The teaching signal is applied such that an activity of 1.0 at the output neuron indicates car detection whereas a value of 0.0 corresponds to a non-car object. The training of system-level models is performed by running the system without modulation using a learning constant of $\epsilon = 0.0001$; Both methods are trained respecting the blocking procedure of Sec. 2.9, the blocking interval being 30s.

2.12 System configurations

The described system can be run in two ways: in *training mode* and *processing mode*.

In *training mode*, the appearance-based classifier is used for hypothesis generation using a competitive selection threshold (see Sec. 2.7) of $\theta_{\text{class}} = 0.0$. Furthermore, the neural learning constant ϵ (see Sec. 2.8) is set to 0.00002 which amounts to assuming $\frac{1}{\epsilon} = 50000$ training examples (more examples do not cause problems, however the relative influence of "old" examples deteriorates in this case). Learning is disabled for examples from the evaluation set (see Sec. 2.9). The system is presented with a concatenation of video streams I,II,III. Once $\frac{1}{\epsilon}$ training examples are processed, training is stopped and the neural weights are stored. During training mode, attentional modulation is disabled since models are only meaningful after training. This is again for convenience only since untrained attentional modulation essentially produces a constant distribution over the image and is thus not causing any effects.

In *processing mode*, learning is switched off by setting $\epsilon \equiv 0$. Instead, previously trained weights are used to generate attentional modulation. In processing mode, *either* symmetry (see Sec. 2.3) *or* the appearance-based classifier (see Sec. 2.2) are used for generating object hypotheses but never both at the same time.

Since object hypotheses have to be of sufficient quality for the online training of accurate system-level models, we do not use symmetry in training mode since its overall car detection performance (when not supported by attentional modulation) is poor, see also Sec. 3. In contrast, we are able to evaluate both methods separately in processing mode with no detrimental effects. Generally, classifier and symmetry have to be used in a mutually exclusive way since the system does not "fuse" results from different object detection mechanisms in the presented form. Apart from this technical point, the distinction between training and processing mode is for convenience only: in this way, the system needs to be trained only once instead of being trained separately for every performance evaluation. A detailed list of parameter settings in training and processing mode can be found in Tab. 2.

2.13 Experimental setup

Video streams and annotations We recorded five distinct video streams covering a significant range of traffic, environment and weather conditions. All videos are around 15 minutes in length and were taken during test drives along a fixed route covering mainly inner-city areas, along with short times of highway driving. Please see Tab. 1 for details and Fig. 13 for a visual impression. For the quantitative evaluation of object detection performance, we manually annotated relevant objects in the recorded video streams, please see Fig. 14 for details.

Evaluation measures For each image, we compute the number of *false positive* hypotheses and *false negative* annotations (see Fig. 15). From these,

ID	weather	daytime	single images	annotated images
I	overcast,dry	afternoon	9843	957
II	low sun, dry	late afternoon	22600	949
III	heavy rain	afternoon	6725	643
IV	dry	midnight	6826	464
V	after heavy snow	afternoon	16551	867

Table 1: Details about the used video streams. Please note that streams II and V were recorded at a frame rate of 20Hz.

Table 2: Global parameters used for experiment

Parameter	explanation	value
H	max. Nr of Hypotheses	40 or 10 (Sec. 3.6)
N,M	image/feature map size	400,300
n,m	population code size	64,64
θ_{class}	classifier selection threshold	task-dependent, 0.0 for eval.
θ_{symm}	symmetry selection threshold	task-dependent, 0.0 for eval.
θ_{MLP}	symmetry selection threshold	task-dependent, 0.0 for eval.
ϵ	system-level learning rate	0.00002 or 0.0001 (Sec. 3.2)
K	nr of pyramid scales	16 (classifier) or 8 (symmetry)
n_{blocking}	blocking interval	30s

we obtain two standard quality measures (see,e.g., [5]) denoted *false positives per image* (fppi) and *recall*. For a fixed parameterization of the system, the performance is given by a point in a recall/fppi-diagram. By plotting these two quantities against each other for variations of the detection thresholds θ_{class} or θ_{symm}, we obtain plots similar in interpretation to receiver-operator-characteristics (ROCs). Such ROC-like plots will be used for visualizing object detection performance in Sec. 3. We only consider annotations whose associated occlusion value (see Fig. 14)is less than 80%.

3 Experiments and Results

For all experiments, the training and evaluation of system-level models is performed using the blocking procedure described in Sec. 2.9. To reduce computational effort, the variation of the object detection thresholds, be it θ_{class}, θ_{symm} or θ_{MLP}, is not conducted by running the system over a whole video stream for each possible threshold value. Rather, the system is run once using object detection thresholds of 0.0 and simultaneous storing of object detection confidences. Subsequently, detection performance for *all* threshold values can be computed offline using the recorded confidences. As a consequence, all object detection thresholds have zero values in Tab. 2. For actually running the system for performing car detection, a suitable threshold would have to be selected for either θ_{class}, θ_{symm} or θ_{MLP}.

3.1 Effect of learned attentional modulation on object detection performance

For determining the performance gain due to attentional modulation, system-level models are trained using streams I,II,III (since a comprehensive training set can be expected to result in good generalization ability of the trained models). Performance is evaluated for the appearance-based classifier on streams I-V using both space- and feature-based attentional modulation. In order to establish a baseline for comparison, we additionally evaluate the system's performance when attentional modulation is disabled (i.e., the pyramid of modulation maps from Sec. 2.7 is set to $m_i(\vec{x}) \equiv 1$) which amounts to evaluating the appearance-based classifier alone.

As described in Sec. 2.13, we create ROC-like plots by varying the classifier threshold θ_{class} (see Sec. 2.2) for comparing the system performance to baseline performance. The resulting plots are given in Fig. 16.

3.2 Generalization to different environment conditions

In analogy to cross-validation methods, this experiment is intended to show that training system-level models on data from *any* video stream and testing on the remaining ones gives comparable performance in each case. We therefore trained system-level models on each stream separately using parameters given in Tab. 2 and evaluated on streams I-V as in Sec. 3.1. Results did not show notable differences to the performance observed in Sec. 3.1, therefore Fig. 17 shows results only for the case of training using stream III, one of the most challenging video streams.

3.3 Bootstrapping using the appearance-based classifier

The third experiment is intended to show that the successful training of system-level models does not require annotations. In the case of the presented system, the appearance-based classifier (see Sec. 2.2) can, due to its already strong performance, replace annotated data by its object class estimate for each training sample. Each object class estimate provided by the classifier is converted to a population code as described in Sec. 2 and Fig. 3 and provided as supervision signal to the training of models (see Sec. 2.9). Results are shown in Fig. 18.

3.4 Fusion of multiple modulation signals

This set of experiments provides insights into the effects of fusing attentional modulation signals. Using system-level models trained as described in Sec. 3.1, we repeatedly evaluate the system's performance for streams I-V, applying various subsets out of the set of available modulation signals. In this way, we can quantify the individual contributions of each modulation signal when using the fusion mechanism described in Sec. 2.10. Furthermore, we conduct experiments about the effects of the uninformative distance-based modulation signal (see

Sec. 4.1) on the fusion process as stated in Sec. 1.2. By including and omitting this modulation signal, a quantitative statement w.r.t. the robustness of the fusion process can be obtained. The results can be viewed in Fig. 19.

3.5 Generalization to different object detection methods

In order to show that attentional modulation can be applied with benefits to different object detection algorithms, we evaluate the effects of attentional modulation using symmetry (see Sec. 2.3) for generating object hypotheses. Symmetry requires no training but only produces meaningful object hypotheses at night. Therefore, evaluation was conducted using stream IV only. The results are given in Fig. 20.

3.6 Assessment of early attentional modulation versus late rejection

This experiment is intended to assess performance differences between our method of attentional modulation where models are coupled in *early*, i.e., before competitive hypothesis selection (see Sec. 2.7), and the alternative where a *late* coupling of models is performed, i.e., after hypothesis selection. For this purpose, we implement and train such a "late" system as described in Sec. 2.11 and compare its performance to that achieved using "early" attentional modulation.

For this purpose, two experiments are conducted for each stream, differing only in the value of H, the upper limit on the number of hypotheses imposed by competitive hypothesis selection (see Sec. 2.7). The different values of H reflect different degrees of resource constraints: $H = 40$ represents the default case of abundant resources, whereas $H = 10$ is intended to simulate strong constraints on, e.g., processing time.

For both experiments, the performance of late rejection and early modulation is evaluated. This is achieved by varying one of the thresholds θ_{class}, θ_{MLP} while leaving the other at 0.0. We therefore obtain two ROC-like curves per experiment and stream. For each stream, we now compare performances of early and late approaches for different values of H. Evaluation is performed on streams I-V but did not differ significantly, therefore Fig. 20 shows only results for stream I and III.

4 Discussion

In this section, we will discuss the evaluation of the presented system w.r.t. these requirements, based on the research hypotheses put forward in Sec. 1. In Sec. 4.3, we will present a critical comparison to existing work and suggest possible improvements.

4.1 Assessment of results w.r.t. research hypotheses

Performance increase by attentional modulation The experiments of Sec. 3.1 showed that the "translation" of multimodal system-level models into attentional modulation signals is feasible and results in significantly increased object detection performance. The performance increase is more marked for the "difficult" streams IV and V; we hypothesize that this is due to increased visual ambiguity (caused by imprecise classifier models, low light or low contrast conditions) whose resolution by attentional modulation then has a potentially larger effect. It can also be observed that attentional modulation improves performance on both the fppi and the recall axis in Figs. 16,17, reflecting the fact that modulation can enhance as well as suppress. To be certain of our results, we checked whether the performance increase occurs for stricter match measures (see Sec. 2.13) as well and found that, although absolute performance drops, the relative improvement by attentional modulation persists.

Generalization The results presented in Sec. 3.2 suggest that trained attentional modulation, in contrast to the appearance-based classifier, exhibits significant generalization to environment and weather conditions encountered in the video streams. The system-level models of Sec. 3.2 are trained using examples from stream III only: nevertheless performance of the attentional modulation on the remaining streams, e.g., I,II,IV,V is strong. When using the blocking procedure on the same video stream, it might be argued that general environment conditions are still shared because they are taken from the same video stream. However, considering the extreme differences in lighting, visibility and contrast *between* video streams, this experiment demonstrates that strong generalization can indeed be achieved.

It should be ensured that this generalization is due to the system-level models and not just coincidence. Although it is unlikely that overfitting occurs given the good generalization demonstrated for attentional modulation signals, we want to explicitly compare performance of system-level models on training and evaluation sets. For this purpose, we employed the training and evaluation procedures described in Sec. 2.11 using stream I. Fig. 21 shows that performance on the training sets is superior, but only slightly. These results persist when using video streams II-V.

Given the strong within-stream differences that are reflected by the large blocking interval, we can state that overfitting does not occur to a significant extent. It is intuitively clear that small blocking intervals lead to similar training and test sets in continuous video streams. Since there is one annotated image per second, and as there are 30-40 training examples (object hypotheses) per image, the blocking interval of 30s amounts to approximately 1000 examples. For this reason, we argue that training and test sets are sufficiently dissimilar to assess generalization behavior. The blocking procedure described in Sec. 2.11 is an accepted way of evaluating the real-world performance of detection systems, see, e.g.,[20].

Fusion of attentional modulation models The experiments of Sec. 3.4 show that the fusion of informative attentional modulation signals improves performance. Conversely, performance is unaffected when an *uninformative* signal is added to the fusion process due to the intrinsic properties of uninformative signals. This robustness property is crucial for real-world applicability since the uncontrollability of real environments can easily give rise to situations where individual system-levels become uninformative. In such cases, attentional modulation must continue to be meaningful, otherwise misjudgments can occur with potentially grave consequences.

We determine whether a modulation signal is informative by analyzing the performance of its underlying system-level model. As can be seen from Fig. 21, the system-level models for distance-to-free-area and elevation are much more informative than the distance-based system-level model which is essentially at chance level. System-levels for position/size are informative as well (not shown) but show inferior performance. The experiments of Sec. 3.4 suggest that combining informative modulation signals increases performance beyond the level achieved by individual attentional modulation signals: this is especially the case for stream V where one can observe an improvement due to the fusion process even though the individual modulation signals (especially the distance-to-free-area) achieve unsatisfying results by themselves.

Application to different object detection mechanisms By applying attentional modulation to a simple symmetry-based detection mechanism, we could show that the proposed mechanism of learned attentional modulation is applicable to very different object detection methods with beneficial results. The detection method need not even be specific to the object class of interest (just as symmetry detection is not a really good car detector, see Sec. 3.5); in such cases, the object specificity is almost exclusively due to the influence of attentional modulation. The only requirements are the existence of a (possibly multiscale) confidence map with retinotopic organization and a competitive hypothesis selection process, e.g., as described in Sec. 2.7. As a consequence, the described learned attentional modulation can be expected to work well with saliency maps[16] or other low-level point detectors.

Benefit of early modulation As can be clearly seen from Sec. 3.6, the late rejection approach is moderately inferior for $H = 40$. For $H = 10$, however, the difference is very pronounced, especially considering the achieved values on the recall axis. This is a very important result when considering object detection in autonomous agents usually facing severe computational constraints. In order to ensure that this effect is really due to the beneficial influence of attentional modulation, it must be established that the reported performance gain is not simply due to superior performance of the system-level models as compared to the MLP. In order to clarify this, we compare the classification performance of the individual system-level models to the performance of the trained MLP. System-level models perform two-class discrimination and can therefore be evaluated by ROC analysis. As can be seen from Fig. 22, even the best system-level model does not approach the classification performance of the MLP. We conclude that superior object detection performance occurs because the modulated classifier has access to more information: it can use both system-level and detailed retinotopic information, whereas the MLP can just use system-level information.

Bootstrapping The results of Sec. 3.3 show that attentional modulation derived from bootstrapped system-level models yields results that are significantly superior to those obtained without modulation. At the same time, performance is only slighty inferior to the performance achieved by using system-level models trained on ground-truth information. For the purposes of this article, successful bootstrapping implies that our system is capable of fully online operation without requiring ground-truth data for training at run-time. Obviously, ground-truth data is still required for training the classifier, but but the *additional* ground-truth data required for training successful system-level models is avoided. A systematic comparison and an in-depth analysis of the benefit of bootstrapping will be given in a subsequent publication (but see [21, 29]).

4.2 Online learning capability

As the term ”online learning” is used in various ways in the literature, we wish to give a precise definition here before we discuss the presence of this property in our system. We assume the following properties:

1. The total number of training examples does not have to be known at any point during the system's run-time.

2. Each training example is seen only once

3. Learning is performed using only information that is (or would be) available to a performing system. This specifically excludes the use of annotated data at run-time, whereas the use of annotated data prior to run-time is of course acceptable.

Without considering bootstrapping, our system fulfills only the first two conditions. Although, by the choice of the learning rate constant ϵ, a time scale is defined after which previously presented examples are slowly forgotten, this does not contradict the stated requirements. Moreover, forgetting only occurs if a training example is not reinforced by similar ones. When taking bootstrapping(see Sec. 3.3) into account, also the third requirement is fulfilled. In this configuration, annotated data is only used for training the appearance-based classifier which occurs prior to the run-time of the system. We therefore conclude that the presented system is indeed capable of performing online learning, enabling it to run and learn in a ”live” system once processing speed has been optimized sufficiently.

4.3 Comparison to related work

There are several differences of our work to the literature discussed in Sec. 1.3. First of all, most investigations do not share the symmetry between training and evaluation our system exhibits. Mostly, system components or prior distributions are trained offline and separately, and are later connected by either probabilistic inference or heuristic coupling. In contrast, we present a system which obtains all required training information while running. As a consequence, training and evaluation of system-level models can be assumed to operate on similar underlying probability distributions: in this way, the common effect that heuristically chosen training data (e.g., negative examples) are actually different from evaluation scenarios cannot occur. Furthermore, the learned system-level models are not purely visual but multimodal in nature and are derived from object properties with powerful semantic meaning, such as an object's distance to the obstacle-free area ahead of the agent, or an object's height above the computed ground plane. We thereby go beyond many approaches which use straightforward visual object properties like pixel size, pixel position, color a.s.o. We also present an example of successful bootstrapping of models for attentional modulation, showing that a system can perform successful online learning if a self-supervision signal of sufficient quality is available. Additionally, we present an investigation which shed light on a previously disregarded aspect of attentional modulation, namely the benefit compared to "late coupling" approaches that eliminate inconsistent detections only at the end of a processing chain. Lastly, the presented system differs from related work by its large-scale evaluation using continuous and variable traffic video sequences. Some authors[3] use extensive evaluation datasets like the PASCAL data but the focus is not on recognizing relevant classes in road traffic scenes, but to recognize and discriminate a large number of object classes in arbitrary scenes. Other authors[13] evaluate performance in traffic scenes but with evaluation sets that are much smaller than ours.

When comparing our system to [2, 9, 10], it is obvious that the modeling of cortical interactions is much more restricted since we focus strongly on the modeling of abstraction hierarchies. Functionally, we use a simplified competition mechanism at hypothesis level (see Sec. 2.7) which clearly does not capture the details of a fully neuro-dynamic approach (hysteresis, latency behavior, ..). However, this simplified mechanism still converges to attractor states which are non-trivially influenced by attentional modulation signals. Thus, while gaining computational efficiency and simplicity, our system is able to make use of the computational power of the biased competition mechanism. Furthermore, the model of [10] considers only the learning of a single object search template; this is in contrast to our approach where a large number of examples are processed to generate a detailed but general system-level model that can be inverted for detecting the target object class. This point applies equally to [2] where learning is not considered at all, and an even more strong focus is given to the network dynamics and biological plausibility.

Approaches based on high-level semantic models such as [25] use models of higher abstraction and complexity than our system with impressive results. The key difference is that such models are designed not learned, and a rigorous evaluation in real-world environments is not targeted.

In contrast to Selective Tuning models [30, 33] where attentional feedback is propagated through multiple hierarchy levels, attentional modulation is only propagated to the intermediate level of the presented system. We did not implement further feedback propagation since the algorithms at the preprocessing level are, at present, not suited to deal with this information. Similarly to Selective Tuning, we use winner-takes-all interactions and, effectively, an inhibition-of-return mechanism at the hypothesis level.

Top-down attention approaches such as [6, 23] differ from our work by the way of acquiring models. Although these authors present integrated systems using object specific attentional modulation, such modulation is obtained by performing an offline optimization based on heuristically defined positive and negative examples. The authors describe evaluations in indoor scenes[6] and traffic environments[23] although the number and diversity of annotated images is much lower than in the presented work, especially for indoor evaluations.

Very closely related to our work is the work by [13] which aims at reconstructing 3D scene geometry from monocular images; such geometric information is then used to guide exhaustive object detection mechanisms. In contrast, we use information about 3D scene layout directly obtained from advanced stereo processing; on the other hand, our system does not perform Bayesian inference to determine the most likely 3D scene layout since we rely on the quality of our stereo information. Additionally, our system is able to process and train models using various quantities not related to 3D scene layout, such as distance-to-free-area, image position/size and many more (color, texture, aspect ratio which were not shown because their influence on performance was not significant). An evaluation of car and pedestrian detection performance in images of outdoor traffic scenes is given in [13], although the number of annotated images and objects is much smaller than in our evaluation.

Similarly to [13], the work of [24] uses global scene properties to infer positions and sizes of objects; however in contrast to [13], this is an unidirectional process where the position of objects cannot influence scene property estimation. In [24], a low-dimensional scene descriptor (”gist”) is computed and used for object training models that relate the positions of certain object classes to the current gist value. Using a small training and evaluation set of indoor/outdoor scenes, performance improvement is demonstrated w.r.t. exhaustive object detection. This is similar to our approach, although our evaluation datasets are much larger, and we employ a larger number of models that inform object detection about the likely positions and sizes of objects.

Another interesting approach to attentional modulation is presented in [36]; as in our work, the influences of multiple models are fused using probabilistic inference to obtain attentional modulation. Used models are: prior distribution over object positions, visual conspicuity computed by a saliency map and a model computing the location where the greatest information gain given previous detections may be obtained. In addition, another topic also discussed here is raised: accuracy of object detection using only a limited number of object hypotheses (there called ”gaze targets”). The authors show that, using their method with a limited number of gaze targets, the performance of exhaustive object search can be approached in indoor scenes. We obtain exactly this result, although our evaluation is considerably more extended and the detection task of finding cars in cluttered outdoor scenes is, to our mind, a more challenging one.

The method put forward in [3] proposes a generic framework for spatial inter-object influences in object detection. In contrast to our system which uses heuristic non-maxima suppression (NMS) to reduce the number of object hypotheses, the authors of [3] train discriminative models for performing this task in a way that is learned from data. It is notable that this framework is also capable of enhancing object hypotheses; this is in contrast to our NMS method which can just suppress. In our investigations, we heuristically determined certain parameters in the NMS of Sec. 2.2 that are beneficial for detecting cars, so we can confirm that the optimization of NMS can indeed improve detection performance.

5 Summary and future work

We presented a large-scale integrated processing system performing object detection in challenging and diverse visual environments. It is our conviction that the presented system is unique in enhancing object detection by space- and feature-based attentional modulation that is autonomously trained within the system, as well as a rigorous evaluation of real-world performance.

Future work will include the investigation of attentional modulation signals with higher object specificity, as well as space-based attentional modulation based on more behavior-centered spatial representations. We will continue evaluating our research based on real-world data while considering more task-specific ways of evaluating detection performance. As a last point, we will conduct further investigations regarding the possibilities of *bootstrapping*, especially w.r.t. the minimal quality an object detector must achieve for successful bootstrapping.

References

[1] CM Bishop. *Pattern recognition and machine learning.* Springer-Verlag, New York, 2006.

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

[3] C Desai, D Ramanan, and C Fowlkes. Discriminative models for multi-class

object layout. In *International Conference on Computer Vision (ICCV)*, 2009.

[4] R. Desimone and J. Duncan. Neural mechanisms of selective visual attention. *Annu Rev Neurosci*, 18:193–222, 1995.

[5] P Dollar, C Wojek, B Schiele, and P Perona. Pedestrian detection: A benchmark. In *CVPR*, June 2009.

[6] 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.

[7] A Gepperth. Implementation and evaluation details of a large-scale object detection system. Technical Report TR 10-11, Honda Research Institute Europe GmbH, 2010.

[8] A Gepperth, J Fritsch, and C Goerick. Cross-module learning as the first step towards a cognitive system concept. In *First International Conference on Cognitive Systems*, 2008.

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

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

[11] BY Hayden and JL Gallant. Time course of attention reveals different mechanisms for spatial and feature-based attention in area V4. *Neuron*, 47, 2005.

[12] S Hochstein and M Ahissar. View from the top: hierarchies and reverse hierarchies in the visual system. *Neuron*, 36(5):791–804, Dec 2002.

[13] D Hoiem, AA Efros, and M Hebert. Putting objects into perspective. *International Journal of Computer Vision*, 80(1), 2008.

[14] JB Hopfinger, MH Buonocore, and GR Mangun. The neural mechanisms of top-down attentional control. *Nat Neurosci.*, 3(3), 2000.

[15] L Itti, C Gold, and K Koch. Visual attention and target detection in cluttered natural scenes. *Optical Engineering*, 40(9):1784–1793, Sep 2001.

[16] L Itti and C Koch. Computational modelling of visual attention. *Nat Rev Neurosci*, 2(3):194–203, Mar 2001.

[17] C-H Juan and V Walsh. Feedback to V1: a reverse hierarchy in vision. *Exp Brain Res*, 150, 2003.

[18] S Kastner and LG Ungerleider. Mechanisms of visual attention in the human cortex. *Annual Review of Neuroscience*, 23, 2000.

[19] DC Knill and A Pouget. The bayesian brain: the role of uncertainty in neural coding and computation. *Trends Neurosci*, 27(12), 2004.

[20] B Leibe, N Cornelis, K Cornelis, and L Van Gool. Dynamic 3d scene analysis from a moving vehicle. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2007.

[21] A Levin, P Viola, and Y Freund. Unsupervised improvement of visual detectors using co-training. In *Proceedings of the ICCV*, pages 626–633, 2003.

[22] CJ McAdams and JHR Maunsell. Attention to both space and feature modulates neuronal responses in macaque area V4. *J Neurophysiol*, 83, 2000.

[23] T Michalke, J Fritsch, and C Goerick. A biologically-inspired vision architecture for resource-constrained intelligent vehicles. *Computer Vision and Image Understanding*, 114(5):548 – 563, 2010.

[24] K Murphy, A Torralba, D Eaton, and WT Freeman. Object detection and localization using global and local features. In J Ponce, editor, *Toward Category-Level Object Recognition*, Lecture Notes in Computer Science. Springer, 2005.

[25] V Navalpakkam and L Itti. Modeling the influence of task on attention. *Vision Res*, 45(2):205–231, Jan 2005.

[26] A Pouget, P Dayan, and RS Zemel. Inference and computation with population codes. *Annu Rev Neurosci*, 26:381–410, 2003.

[27] RD Reed and RJ Marks II. *Neural Smithing*. MIT Press, 1998.

[28] JH Reynolds, L Chelazzi, and R Desimone. Competitive mechanisms subserve attention in macaque areas V2 and V4. *The Journal of Neuroscience*, 19(5), 1999.

[29] P Roth, H Bischof, D Skočaj, and A Leonardis. Object detection with bootstrapped learning. In A Hanbury and H Bischof, editors, *Proc. 10th Computer Vison Winterworkshop*, pages 33–42, 2005.

[30] AL Rothenstein and JK Tsotsos. Selective tuning: Feature binding through selective attention. In *Proceedings of the International Conference on Artificial Neural Networks*, 2006.

[31] K Tanaka. Mechanisms of visual object recognition: Monkey and human studies. *Current Opinion in Neurobiology*, 7:523–529, 1997.

[32] S Treue. Neural correlates of attention in primate visual cortex. *Trends in Neuroscience*, 24, 2003.

[33] J Tsotsos, Y Liu, JC Martinez-Trujillo, M Pomplun, E Simine, and K Zhou. Attending to visual motion. *Computer Vision and Image Understanding*, 100(1-2), 2005.

[34] JK Tsotsos, SM Culhane, W Wai, Y Lai, N Davis, and F Nuflo. Modeling visual attention via selective tuning. *Artif. Intell.*, 78:507–545, 1995.

[35] J Vogel and O De Freitas. Target-directed attention: Sequential decision-making for gaze planning. In *International Conference on Robotics and Automation (ICRA)*, 2007.

[36] J Vogel and K Murphy. A non-myopic approach to visual search. In *Computer and Robot Vision*, volume 0, pages 227–234, Los Alamitos, CA, USA, 2007. IEEE Computer Society.

[37] D Walther, L Itti, M Riesenhuber, T Poggio, and C Koch. Attentional selection for object recongition - a gentle way. In *Lecture Notes in Computer Science*, volume 2525. Springer, 2002.

[38] H Wersing, S Kirstein, B Schneiders, U Bauer-Wersing, and E Körner. Online learning for boostrapping of object recognition and localization in a biologically motivated architecture. In *Proc. Int. Conf. Computer Vision Systems ICVS. Santorini, Greece.*, pages 383–392, 2008.

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

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

[41] RS Zemel, P Dayan, and A Pouget. Probabilistic interpretation of population codes. *Neural Comput*, 10(2):403–430, Feb 1998.

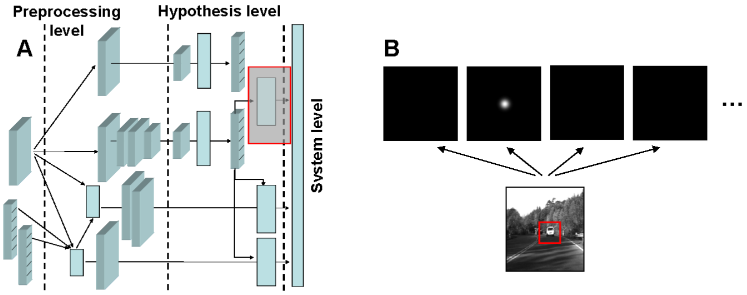


Figure 6: Size-dependent population encoding of hypothesis position. **A)** Embedding into processing system. **B)** Hypothesis size determines the nonzero level in the pyramid of population codes $z_i^0(\vec{p})$.

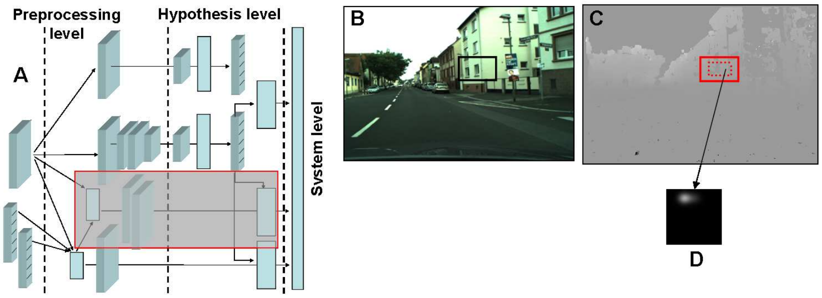


Figure 7: Examples of stereo processing for elevation and distance calculation. **A)** embedding into processing system **B)** video image with object hypothesis **C)** dense elevation map $F^2(\vec{x})$, similar to distance map $F^3(\vec{x})$. **D)** population-coded elevation $z^2(\vec{p})$ resulting from this measurement. An analogous processing generates population coded-distance $z^3(\vec{p})$. Such population codes may be more or less strongly multimodal, thus reflecting the uncertainty of the associated measurement.

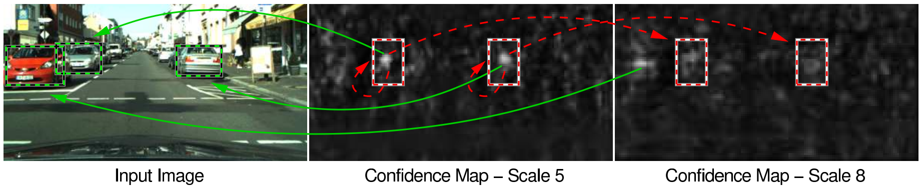


Figure 8: Competitive hypothesis selection in a resolution pyramid of confidence map produced by the appearance-based classifier. Maxima in the confidence maps (right) correspond to object hypotheses defined by rectangular areas in the input image (left). As indicated, a maximum with high confidence inhibits its neighborhood region across all scales.

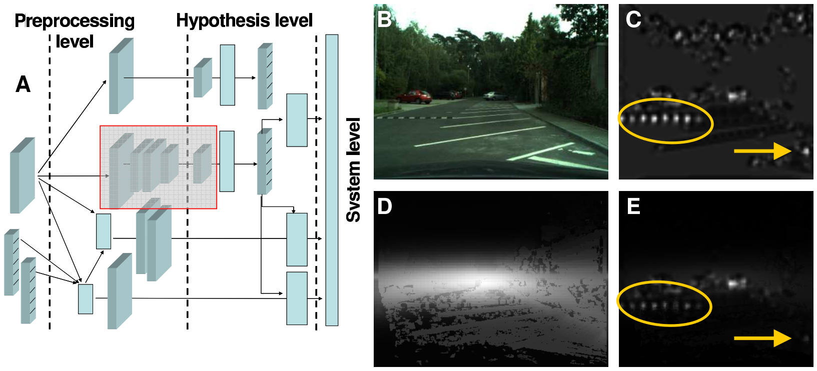
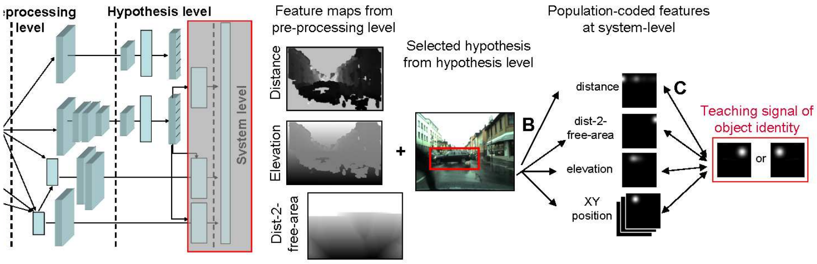


Figure 9: Typical effects of attentional modulation on classifier. **A)** Embedding into processing system. **B)** Sample input image. **C)** confidence map of classifier at scale 5. Note the strong (but incorrect) maxima indicated by the ellipse and the arrow. **D)** Top-down modulation image at scale 5. **E)** Modulated confidence map. Note that the local maxima indicated by the arrow and the ellipse have been merely attenuated; especially the maximum indicated by the arrow may still be selected since there are no competing maxima nearby. In contrast, local maxima close to the upper border of the image have been almost eliminated. Selection behavior depends strongly on the number of allowed hypotheses, H, and the selection threshold θ.



processing level
Hypothesis level
System level
Feature maps from pre-processing level
Distance
Elevation
Dist-2-free-area
+
Selected hypothesis from hypothesis level
B
Population-coded features at system-level
distance
dist-2-free-area
elevation
XY position
C
Teaching signal of object identity
or

Figure 10: System-level learning of object models. **A)** Embedding into processing system **B)** Encoding of system-level quantities at the hypothesis level. **C)** Learning of the mapping between object identity and population-coded system-level quantities. Note that both directions of the mapping are learned, i.e., one can determine the expected identity given a feature, but just as easily the expected feature distribution given an identity. The latter case is used for generating attentional modulation.

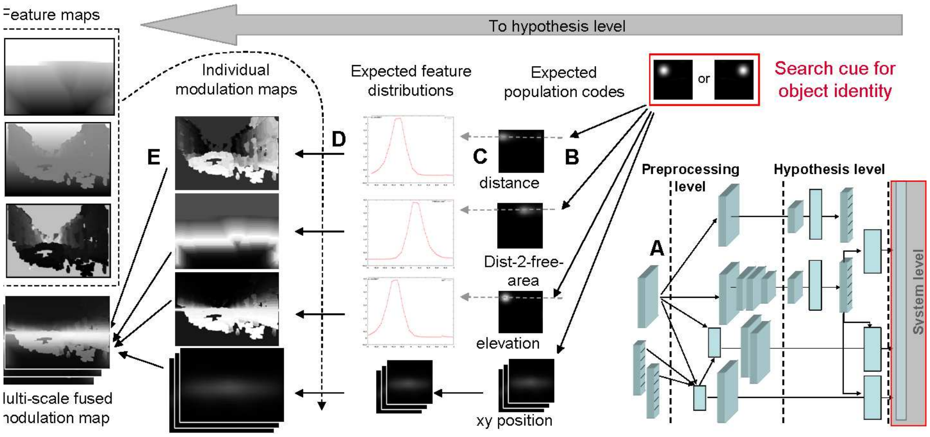


Figure 11: Generation of object-specific attentional modulation. **A)** Embedding into processing system **B)** Learned reverse mapping from an object identity representation (the search cue) to population-coded feature value distributions. **C)** Decoding of the population-coded feature value distributions. For all features except xy-position, this involves a "cutting" of the population code along the line indicated by the gray dashed arrows. **D)** Generation of individual attentional modulation maps. For all features except xy-position, this involves a lookup operation, substituting values found in individual feature maps by corresponding confidence values from expected distributions generated in step C. **E)** Fusion of modulation maps by simple addition and normalization.

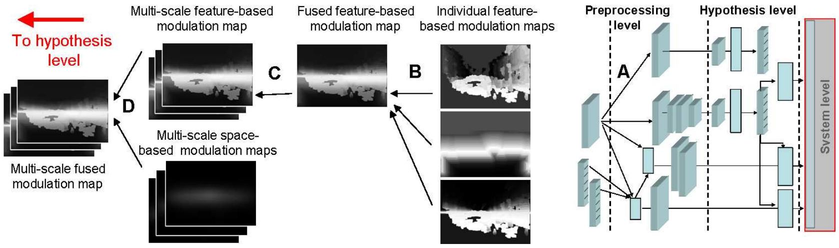

To hypothesis level
Multi-scale feature-based modulation map
Fused feature-based modulation map
Individual feature-based modulation maps
Preprocessing level
Hypothesis level
A
B
C
D
Multi-scale fused modulation map
Multi-scale space-based modulation maps
System level

Figure 12: Fusion of space- and feature-based modulation signals. **A)** Embedding into processing system. **B)** Normalization and summation of feature-based attentional modulation signals. **C)** Duplication across scales **D)** Summation of multiscale feature- and space-based modulation signals with subsequent maximum normalization. The resulting multiscale modulation map therefore fulfills $m_i(\vec{x}) \in [0,1] \forall i \forall \vec{x}$.

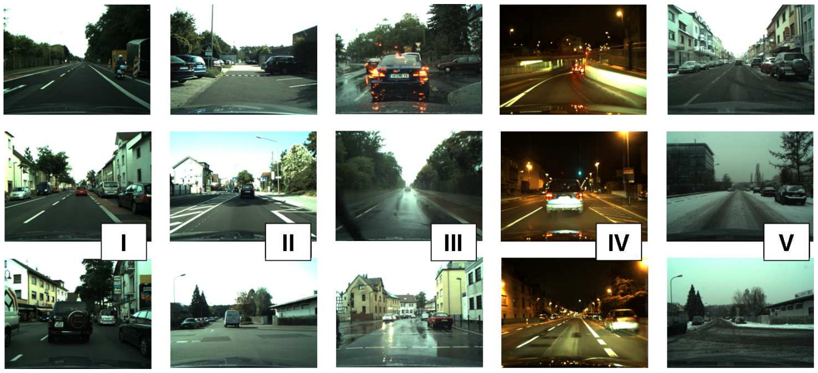


Figure 13: Selected example images from streams I-V. All videos were taken in RGB color using a MatrixVision mvBlueFox camera at a resolution of 800x600. Used frame rates were 10Hz except for video II where a setting of 20Hz was used. Aperture was always set to 4.0 except for video IV where we used a value of 2.4. A self-implemented exposure control was used on both cameras, manipulating the gain and exposure settings of each camera.

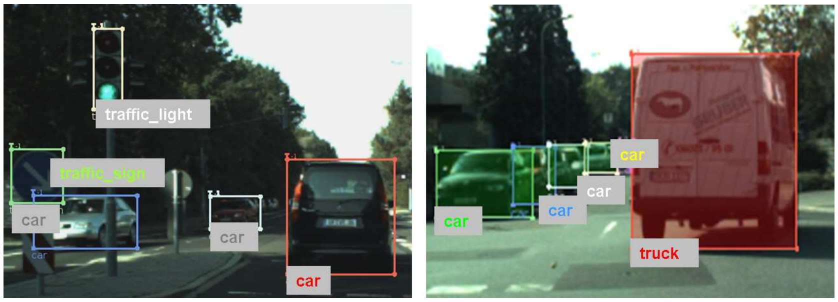


Figure 14: Examples of recorded streams and annotated information. Each annotation consists of a rectangular area, an identity and an occlusion value (not shown). In order to reduce the annotation effort, only every tenth image in a video sequence was annotated. We annotated positive examples for a number of different object classes. Since this study focuses on vehicles, we ensured that really all vehicles present in a given image are covered by an annotation. As can be seen from the images, we use what we term *semantic annotations*, which means that is has been tried to mark the whole area containing an object even if it is partially occluded.

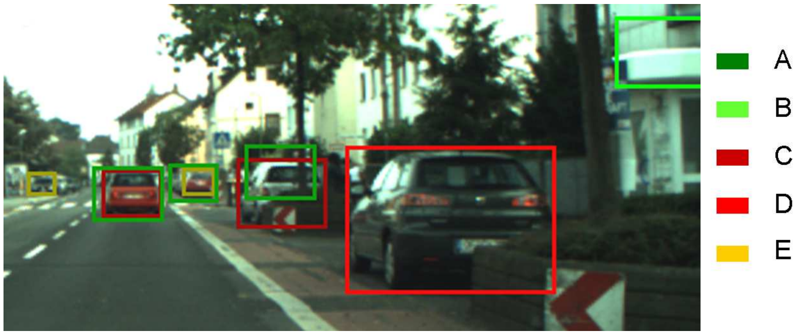


Figure 15: Example of single-image performance evaluation. A) Hypothesis matching an annotation (true positive case) B) hypothesis not matching any annotation in the current image (false positive case) C) annotation matched by one or more hypotheses D) annotation not matched by any hypothesis (false negative case) E) annotation that is not considered due to size constraints, see text. Such annotations do not constitute a false negative case when matched by a hypothesis, but neither a true positive case otherwise.

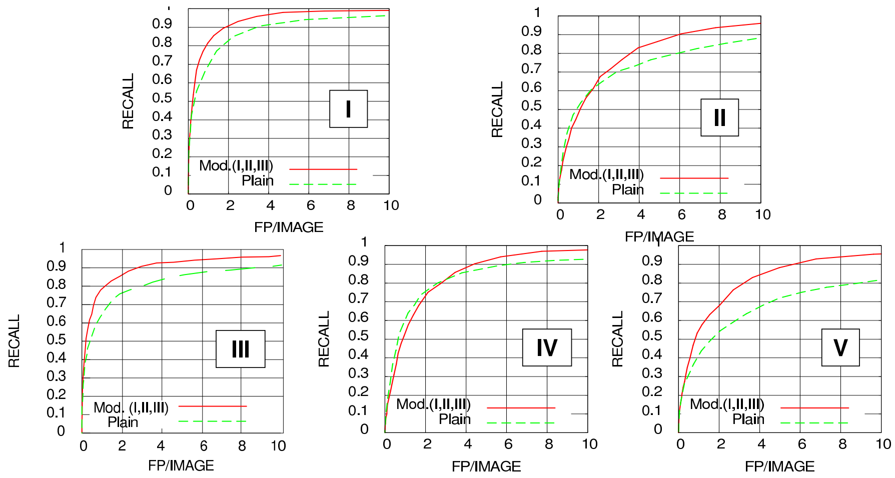


Figure 16: Assessment of performance improvement by attentional modulation for video streams I-V by ROC-like plots. The dashed green curves give the performance of the appearance-based classifier without attentional modulation, the solid red curves show the performance when using attentional modulation. System-level models were trained on streams I-III using blocking. A clear improvement can be observed for all streams.

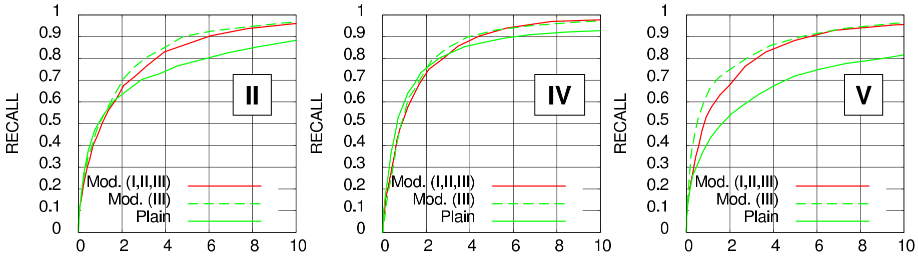


Figure 17: Assessment of generalization performance of top-down modulation using stream II,IV,V. System-level models were trained on stream I,II,III (baseline) and on stream III, both times using blocking. Results were very similar in nature on streams I,III (not shown). Solid green curves: appearance-based classifier without modulation. Solid red curves: appearance based classifier using attentional modulation trained on streams I,II,III. Dashed green curves: appearance based classifier using attentional modulation trained on stream III. As can be seen from the plots, training system-level models only on stream III does not affect performance significantly in any direction.

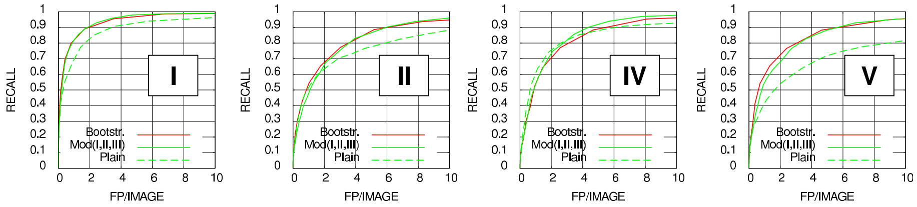


Figure 18: Performance comparison of attentional modulation using bootstrapped and annotated training (stream I not shown). Dashed green curves: plain classifier performance without attentional modulation. Solid green curves: effects of attentional modulation trained on streams I+II+III. Red curves: effect of system-level models trained on streams I+II+III using bootstrapping. Performance using bootstrapped training is very similar to annotated training and markedly superior to the ”plain” classifier (except on stream IV).

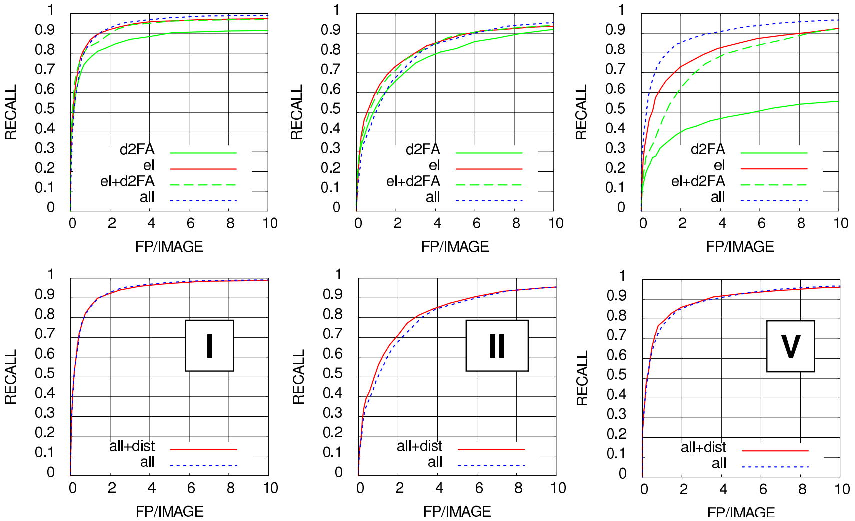


Figure 19: Improvement of detection performance by the fusion of attentional modulation signals. For space limitations, we show only streams I, III,IV. Upper row: effect of fusing informative modulation signals on detection performance. Solid red curves (”el”): elevation only, solid green curves (”d2fa”): distance-to-free-area only, dashed green curves (”el+d2fa”): fusion of distance-to-free-area and elevation, dotted blue curve (”all”): fusion of position/size (not shown), distance-to-free-area and elevation signals. In streams I and IV, good performance is mainly obtained through the elevation signal. In stream V, the free area computation often fails due to laser reflections, resulting in meaningless distance-to-free-area measurements and impaired elevation measurements. As can be seen, the fusion of modulation signals makes performance robust against failure (documented by poor distance-to-free-area performance) or deterioration (documented by impaired elevation performance) of individual modulation signals. Lower row: robustness of the system against addition of uninformative modulation signals. The inclusion or omission of the distance-based modulation signal only has a negligible effect. Solid red line (”all+dist”): detection performance when using distance-to-free-area, elevation, position/size and distance. Dotted blue line (”all”): detection performance when omitting distance.

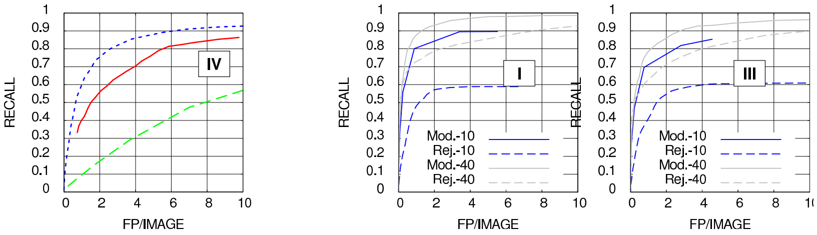


Figure 20: **Left graph:** Effect of attentional modulation on symmetry-based object detection. Dashed green curve: symmetry-based detection without modulation. Solid red curve: symmetry-based object detection with attentional modulation. Dotted blue curve: appearance-based classifier without attentional modulation (for comparison). As can be seen, attentional modulation improves the almost chance-level performance of symmetry-based car detection to a level close to the much more powerful appearance-based classifier. **Right two graphs**: Comparison of early modulation and late rejection approaches under moderate ($H = 40$) and strong ($H = 10$) resource constraints, shown for streams I and III. Solid curves: attentional modulation with strong (blue curve) and moderate (gray curve) constraints. Dashed curves: late rejection with strong (blue curve) and moderate (gray curve) constraints. Please observe the marked difference between resource-constrained object detection performance using attentional modulation (solid blue curve) or late rejection (dashed blue curve). Especially on the recall axis, the late rejection approach achieves a much poorer performance when simulating constrained resources. This effect was observed also on streams II,IV and V (not shown).

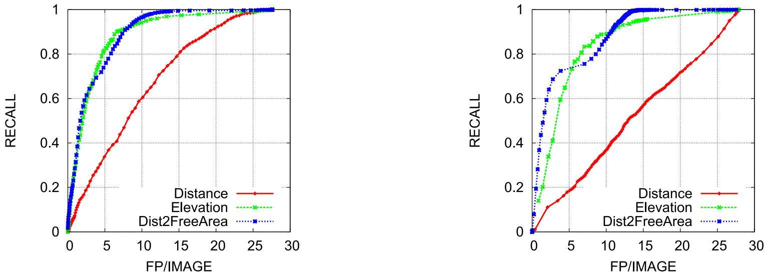


Figure 21: Checking system-level models for overfitting. We compare performance of system-level models evaluated on disjunct training and evaluation sets from stream I. Left: performance on training set. Right: performance on evaluation set. Training set performance is somewhat superior but significant performance is still achieved on the evaluation set. In case of overfitting, the performance on the evaluation set should differ much more strongly.

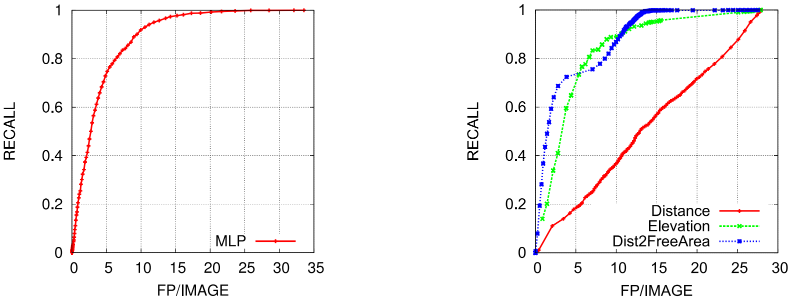


Figure 22: Direct comparison of system-level models and MLP classifier performed on evaluation set from stream I. Left: multilayer perceptron using population-coded distance, elevation, size and distance-to-free-area as input. Right: Individual system-level models for each feature, see Sec. 2.8. MLPs performance is slightly superior overall which is unsurprising since it is three-layered, can use nonlinearities and combines all its input features. In contrast, the system-level models directly map each population-coded input to object identity (no combination). Training and evaluation of MLP and system-level models was performed as described in Sec. 2.11.