Cogn Comput (2010) 2:326–343
DOI 10.1007/s12559-010-9080-1

Where to Look Next? Combining Static and Dynamic Proto-objects in a TVA-based Model of Visual Attention

Marco Wischnewski · Anna Belardinelli · Werner X. Schneider · Jochen J. Steil

Received: 1 May 2010 / Accepted: 9 October 2010 / Published online: 6 November 2010

Abstract To decide "Where to look next ?" is a central function of the attention system of humans, animals and robots. Control of attention depends on three factors, that is, low-level static and dynamic visual features of the environment (bottom-up), medium-level visual features of proto-objects and the task (top-down). We present a novel integrated computational model that includes all these factors in a coherent architecture based on findings and constraints from the primate visual system. The model combines spatially inhomogeneous processing of static features, spatio-temporal motion features and task-dependent priority control in the form of the first computational implementation of saliency computation as specified by the "Theory of Visual Attention" (TVA, [7]). Importantly, static and dynamic processing streams are fused at the level of visual proto-objects, that is, ellipsoidal visual units that have the additional medium-level features of position, size, shape and orientation of the principal axis. Proto-objects serve as input to the TVA process that combines top-down and bottom-up information for computing attentional priorities so that relatively complex search tasks can be implemented. To this end, separately computed static and dynamic proto-objects are filtered and subsequently merged into one combined map of proto-objects. For each proto-object, attentional priorities in the form of attentional weights are computed according to TVA. The target of the next saccade is the center of gravity of the proto-object with the highest weight according to the task. We illustrate the approach by applying it to several real world image sequences and show that it is robust to parameter variations.

Keywords Modeling visual attention · TVA · Proto-objects · Static and dynamic features · Inhomogeneity · Natural scenes · Top-down control

M. Wischnewski (✉) · A. Belardinelli · W. X. Schneider
Center of Excellence - Cognitive Interaction Technology (CITEC) and Neuro-cognitive Psychology, Bielefeld University, Bielefeld, Northrhine-Westphalia, Germany
e-mail: marco.wischnewski@cit-ec.uni-bielefeld.de

A. Belardinelli
e-mail: anna.belardinelli@cit-ec.uni-bielefeld.de

W. X. Schneider
e-mail: wxs@uni-bielefeld.de

J. J. Steil
Research Institute for Cognition and Robotics (CoR-Lab) & Faculty of Technology, Bielefeld University, Bielefeld, Northrhine-Westphalia, Germany
e-mail: jsteil@cor-lab.uni-bielefeld.de

Introduction

"Where to look next ?" is a central function of visual saliency computations and attention selection. The difficulty lies in capacity limitations of the primate visual system in terms of object recognition and visuo-motor control [54]—limitations that call for selective mechanisms able to prioritize chunks of the fixated scene, possibly containing the best candidates for further processing. As human and non-human primates as well as artificial systems share this problem of limited resources, attention modeling has become essential to explain data of visual search and object recognition [54, 59, 61, 70] as well as for the synthesis of computer vision or robotic gaze orienting systems [5, 19, 42, 52, 56]. Many of the artificial systems have thereby been inspired by the human attentional system and tried to replicate a similar function at different degrees of biological and psychological plausibility [24].

However, none of the existing models comes even close to the apparent ease with which humans integrate bottom-up and top-down control of selective processing, e.g., for efficient visual search informed by task and context. There are three basic kinds of factors determining the outcome of the attentional processing that are heavily investigated in human and non-human primate vision and, consequently, are also subject to computational modeling: bottom-up low-level visual feature maps, visual proto-objects, and top-down task-based control. Bottom-up processing comprises both static and dynamic features and has been extensively studied at the computational level over the years, while more recently object-based [55] and task-based [35, 58] accounts of attention have been emphasized. We discuss the respective modeling approaches in turn and finally devise our model that combines all these aspects in a coherent architecture.

In the human and non-human primate visual system, static and dynamic features are processed via different pathways. Specifically, static features are processed along the ventral pathway while dynamic features follow the dorsal stream, the first being mostly devoted to identification of objects, the second to sensorimotor transformation, although also important in recovering object shape [26, 27]. Following this distinction, modeling of static bottom-up feature processing is inspired by the architecture of the ventral pathway and usually leads to the production of a saliency map from the weighted combination of different feature maps, reflecting the retinotopic structure of the input and considering single dimension conspicuousness at each location [34, 61]. Established basic visual features like intensity, color and orientation [70] are known to play a relevant role in different aspects of low-level visual processing that refer to segmentation and figure-ground discrimination [45]. Many respective computational models for static features have been developed in this direction [5, 19, 31, 32, 56] and aimed at reproducing selected facets of human and non-human primate feature processing to some extent, see [24] for a review. The representation as stack of basic feature maps has been refined and improved during the last years [41, 49, 52, 64]. Nevertheless, the account of attentional selection, when it comes to computational modeling, is mostly pixel-wise feature- and location-based.

In terms of the dorsal pathway that is involved in sensori-motor processing [27], computational modeling has focused on the aspect of motion perception [1, 67] and, more recently, also on attentional selection by motion [4, 6, 30, 37]. Despite biological evidence that the ventral and dorsal pathway share feedback connections to operate a figure-ground segmentation [17, 54], there are only a few approaches available that integrate both streams for the control of spatio-temporal attention [36, 38]. Still these rely on a pixel-to-pixel combination of the two pathways and therefore are difficult to reconcile with the object-based nature of attention.

Attentional selection in everyday tasks is often object-based. We look for something, we want to grasp or manipulate an object, or to navigate an environment while avoiding obstacles. This object-based account of attention has been recently substantiated by growing experimental evidence from highly controlled laboratory studies [8, 55] and it has also been picked up in some recent object-based computational approaches, such as [47, 57] or [65]. They share the idea to bind regions on the feature map level to proto-objects based on color/edge-based segmentation or extraction of coherent regions in one feature channel, respectively, and partially refer to Gestalt ideas for segmentation of such regions [47]. However, these approaches do not use proto-object-based features for further processing. The latter is proposed in the approach of [3], where regions have medium-level features such as size, symmetry, orientation and eccentricity. None of these approaches has included motion features, nor devised, how proto-objects can be included in an overall computational architecture for top-down task-specific control of attention. Finally, based on the growing evidence for task-dependent control of covert and overt attention [8, 16, 35, 58], computational bottom-up models have been extended, mostly by changing the weighting of features [39, 56] or by ex-post modification of the saliency map [43, 44]. Tsotsos' Selective Tuning Model [62] also implements a connectionist form of top-down biasing by enhancing target features and inhibiting distractor features. However, this kind of weighting can account only for simple preferences of basic visual feature channels over others ("look for red!"), but fail to model more complex tasks. On the other hand, within the psychological literature there is the well-established Theory of Visual Attention (TVA, [7–9]) that is capable of explaining a large range of behavioral and neurophysiological data on covert visual attention by means of a relatively simple mathematical model. TVA provides a psychologically plausible and elegant way to combine top-down control of priorities for certain features or categories and bottom-up computed visual information. Importantly, TVA assumes that visual units or proto-objects have been already formed when attentional control is computed. In other words, TVA implies an object-based account of visual attention. Surprisingly, TVA has neither been included in any computational attention model yet, nor has been subjected to stand-alone computational modeling. Hence, we present a computational model of attention centered around *proto-objects* to integrate all discussed factors of priority control: bottom-up static and dynamic features, object-based features in form, size, extension, orientation and location of proto-objects, and

task-dependent priority computation through TVA (see Fig. 1). The computation of proto-objects is the key step in this respect: proto-objects represent discrete units of attention, labeled by the features computed within their boundaries and by their position and extension in the field of view, and provide the input for the TVA stage. As we consider static and dynamic features that can actually be related to the same object, we have to create a single proto-object from different types of overlapping or conflicting proto-objects that we derive separately from the dorsal (dynamic) and ventral (static) computations. This is an instance of the *binding problem* and specifically touches the 'property', 'part' and 'location' binding types, as in the classification proposed by Treisman in [60]. The three types of binding consider respectively how to bind different object properties, how to bind different object parts and how to bind objects and locations. These issues are the focus of the section "Fusion of Ventral and Dorsal Proto-Objects", where we suggest a possible integration of objects and features. In the last step, according to the weight equation of TVA [7], an attentional weight (attentional priority) is computed for each proto-object. The weight determines the degree of priority in perceptual processing. We add the assumption that weights determine also where-to-look-next. The proto-object with the highest weight will be the target of the next saccade [10, 69]. Attentional weights depend on bottom-up influences such as the sensory evidence for visual features and on top-down influences such as the current task. Weights are represented in an attentional priority (saliency) map. It should be noted that the restriction of visual feature and weight computation to regions of proto-objects is computationally efficient and in contrast to pixel-based saliency maps: where there are no proto-objects, features do not have to be computed. Importantly, the proto-object with the highest attentional weight receives highest priority in perceptual processing and simultaneously becomes the target for the next saccade or camera shift [54]. Since any combination of low-level visual and medium-level proto-object features such as size or orientation of the principal axis can be included, relatively complex tasks like search for a "big red moving object" can be performed by our model. In other words, task-based control of attention is enhanced by allowing medium-level visual features to be part of the priority computations.

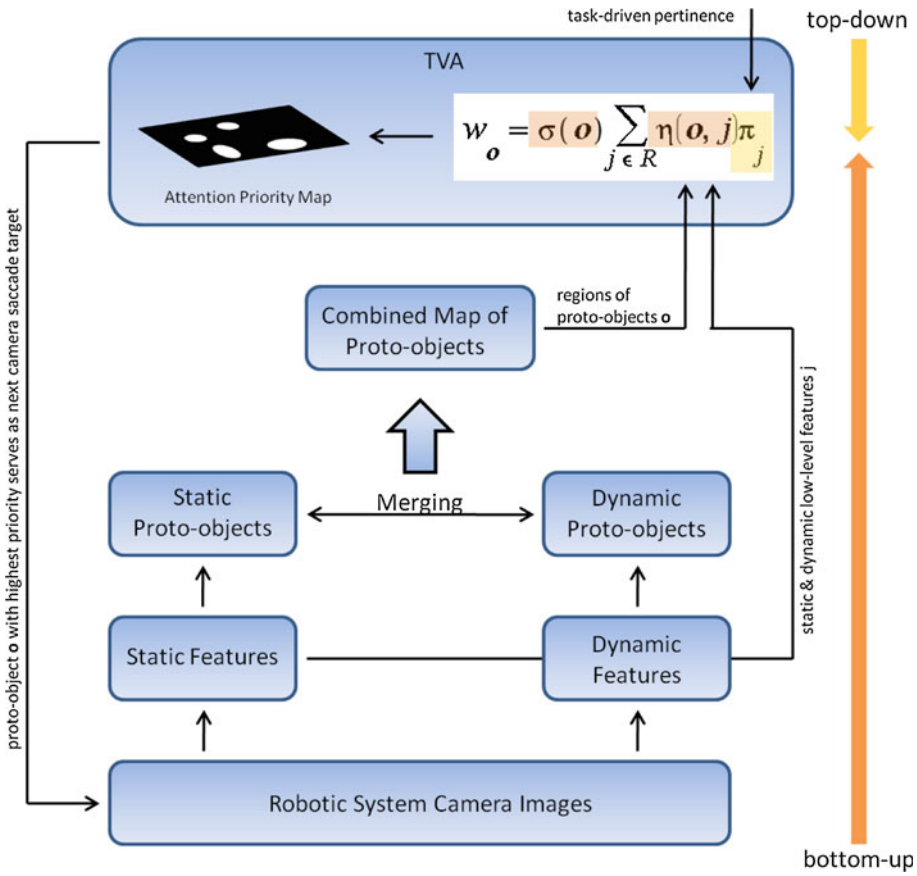

Fig. 1 Schematic overview of the model: A sequence of camera images is acquired as input. The model separately computes static and dynamic features, which allow for the static and dynamic proto-object detection, respectively. In the next step, proto-objects of both processing paths are merged into a combined map of proto-objects **o**. At this point, the static and dynamic low-level features j of the proto-objects **o** and the task-driven pertinence values π are available; that is, all input needed to compute attentional weights for proto-objects using the TVA weight equation. The outcome is the attention priority map of proto-objects holding candidate locations for saccades and possibly further object recognition. The camera system then saccades to the center of gravity of the proto-object with the maximum weight and the next processing cycle starts

Static Features and Proto-objects

This section describes the low-level feature processing and the subsequent proto-object formation based on static image features, which reflects the ventral pathway of processing. Figure 2 illustrates the corresponding part of the model, a more detailed description has been published by Wischnewski et. al [69].

Inhomogeneous Retinal/V1 Feature Processing

Starting from the retina up to higher brain areas, the human visual processing system is spatially inhomogeneously organized. This affects, on the one hand, the density of the retinal photoreceptors which decreases with increasing angle of eccentricity [48]. On the other hand, spatial inhomogeneity affects the filters of the brain area V1 where we can find simple cells for bar and edge detection. With increasing angle of eccentricity, the size of the receptive fields and the distance between adjacent fields increase, whereas their spatial frequency decreases.

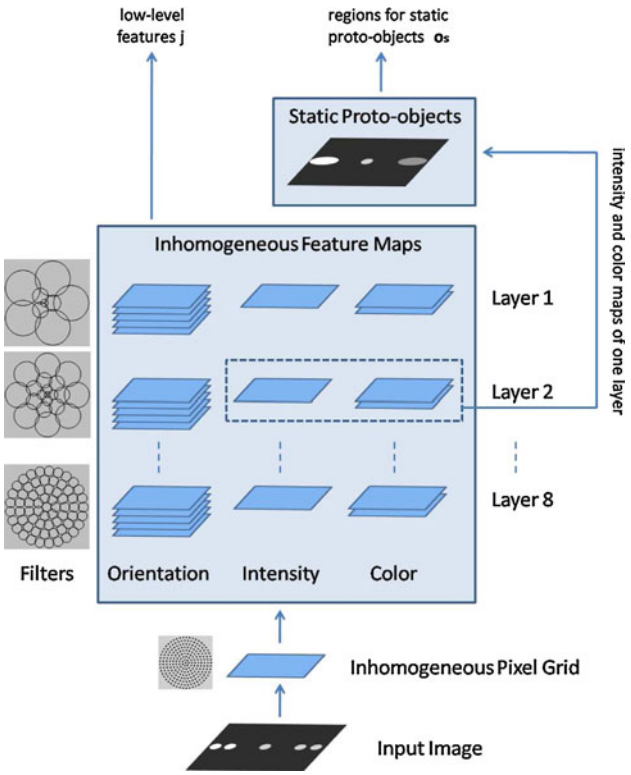

Fig. 2 An illustration of the processing pathway of static image features. First, a single frame from an input sequence is transformed into a retina-like inhomogeneous pixel grid. This grid serves as input for subsequent feature map computations, which are equally spatially inhomogeneous. On the one hand, these maps provide their data directly as static *features j* for the TVA weight equation. On the other hand, depending on simulated time pressure, we chose one of the eight layers and use its corresponding color/intensity maps as input for the static proto-object detection algorithm. The computed $proto-objects\ \mathbf{o}_s$ represent hypotheses about regions of real world objects

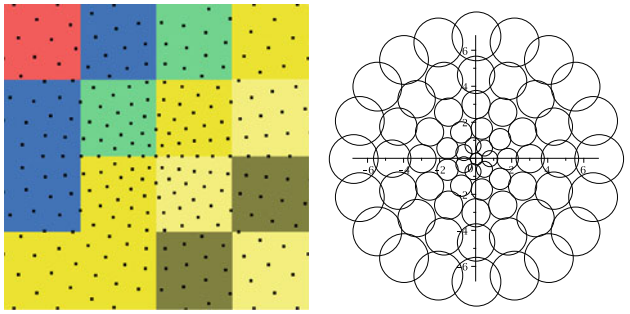

Fig. 3 *Left* An input image section of 4×4 pixels, where black dots mark the inhomogeneous pixels grid positions. *Right* The first five rings surrounding the central Gabor filter with $f_{center} = 1$ and $k = 0.4$. The axes of abscissae and ordinate reflect the angle of eccentricity. For illustration, the filter size was reduced

We model inhomogeneity by positioning filter centers in retina-like concentric rings (see Fig. 3, left). This structure relies on the physiological color/intensity space with the dimensions red/green, blue/yellow and black/white [65]. Following a model of findings in V1 developed by Watson [66], we scale the filters' receptive fields and vary the parameters (size, distance and frequency) with respect to a scaling factor s:

$$s = 1 + k * e, \tag{1}$$

where the factor s is linearly proportional to a scaling parameter k and e is the filter angle of eccentricity in degrees. In the human visual system, k is estimated to be around 0.4 [66] (see Fig. 3, right).

Feature maps are computed for orientation (5 different angles), color (R/G and B/Y) and intensity (B/W) as Gabor filters comprising a cosine function overlaid by a Gaussian, where θ represents the angle of orientation and ϕ the phase:

$$f(x,y) = e^{\frac{-4\ln(2)(x^2+y^2)}{w^2}} \cos(2\pi f(x\cos\theta + y\sin\theta) + \phi). \tag{2}$$

For the color and intensity maps, the filter operations are restricted to the Gaussian part of the equation. To cover the relevant frequency range, we use eight layers of these maps, each with a different parameter setting (see [69] for details). In total, we obtain 64 feature maps. The retinal model provides the data for all subsequent filter operations in the static (ventral) processing path.

We provide an open source C++ library called IIP (Inhomogeneous Image Processing), including a graphical frontend, which we used for all feature map computations.[1]

[1] http://www.uni-bielefeld.de/psychologie/ae/Ae01/IIP/.

Proto-object Formation and Selection

The pivotal units in our architecture are the proto-objects; that is, ellipsoidal visual units in a multidimensional feature space that already have the medium-level features of position, size, shape and orientation of the principal axis while still not being recognized as objects. We describe a static proto-object $\mathbf{o}_s$ formally by a location vector $\boldsymbol{\mu}$ and an inertia matrix $\mathbf{I}$:

$$\mathbf{o}_s = \{\boldsymbol{\mu}, \mathbf{I}\} \tag{3}$$

The eigenvectors and eigenvalues of the inertia matrix denote the axes of the ellipse and therefore its shape, orientation and size. To determine the ellipsoid proto-objects, we apply a standard multidimensional hierarchical cluster approximation algorithm developed by Forssén [21] that, like all pixel cluster algorithms, relies on distances in the respective feature space. Consequently, it operates on a homogeneous feature map pixel grid in the three-dimensional color/intensity space.

Unfortunately, the inhomogenous spacing of filters does not allow a straightforward clustering of feature map pixels, because – due to the particular spacing – close to the fovea several filters may have their center on the same pixel while other pixels in the periphery may not receive any filter values at all. If filters are spaced densely enough (e.g. in the example in Fig. 3), one option is to average filter responses over all filters per pixel. However, this is generally not the case, in particular not if a level with lower resolution is chosen for the initial filtering. We therefore proceed by virtually increasing the resolution of the image until the pixel distance is equal to the smallest possible distance of two filter positions in the highest resolution filter set, which is found between the central filter and the innermost ring. The large number of empty feature map pixels is then filled according to a next neighbor principle: each pixel receives the filter value of its next neighbor in the inhomogeneous set of actually computed filters. These thereby become centers of a Voronoi cell in the high-resolution grid. Figure 4 (middle) shows an exemplary input image processed by lower resolution filters and the respectively processed feature map, where the Voronoi effect is clearly visible in the periphery.

After this preprocessing of the feature map grid, Forssén's algorithm [21] (or any other algorithm that delivers ellipsoid approximations of multidimensional clusters), can be applied to generate the proto-objects. Which clusters, i.e. respective proto-objects, are found depends on the configuration and parameters of the algorithm. Table 1 gives the standard values and Fig. 5 shows how the proto-object configuration depends on value changes. Forssén's algorithm iteratively builds a hierarchy of labeled regions, where d_{max} and m_{thr} govern the region formation: regions r_1 and r_2 are merged if $||RGB_{r_1} - RGB_{r_2}|| < d_{max}$, where RGB is the vector of rgb-values in $[0, 1]^3$, and if the number of common boundary pixels is larger than $m_{thr}\sqrt{min(size_{r_1}, size_{r_2})}$. The parameter c_{min} determines the tendency to fuse regions of layers in the hierarchy, and the number of iterations determines noise reduction.

Forssén also uses a final filtering process to discard too large or small ellipsis, which we substitute taking into account the inhomogeneity again. We count the number of Voronoi cells n_{vc} that a proto-object covers to filter with respect to a combination of physical size and angle of eccentricity. Only proto-objects with n_{vc} in a range

$$n_{vc_{\min}} \leq n_{vc}(\mathbf{o}_s) \leq n_{vc_{\max}} \tag{4}$$

are retained. Larger ellipses can become proto-objects in the periphery because the number of Voronoi cells that are covered by these proto-objects decreases proportionally to the peripherally increasing size of the Voronoi cells.

Our experiments reveal that d_{max} is the most critical value and needs to be chosen with care (see Fig. 5);

Table 1 The standard parameters of the proto-object detection algorithm, see [21] for details

Par.	Value	Effect of variation
d_{max}	0.2	Higher values result in less regions.
c_{min}	0.5	Higher values result in smaller regions.
m_{thr}	0.5	Facilitates merging of overlapping regions.
m_{iter}	5	More iterations reduce noise.

Fig. 4 Formation of static proto-objects for one frame of a video sequence (*left*). Due to the simulated time pressure we use the color and intensity features of just one layer to create the pixel-based voronoi cell mapping (*middle*). Subsequent color blob detection [21] yields the ellipsoid static proto-objects

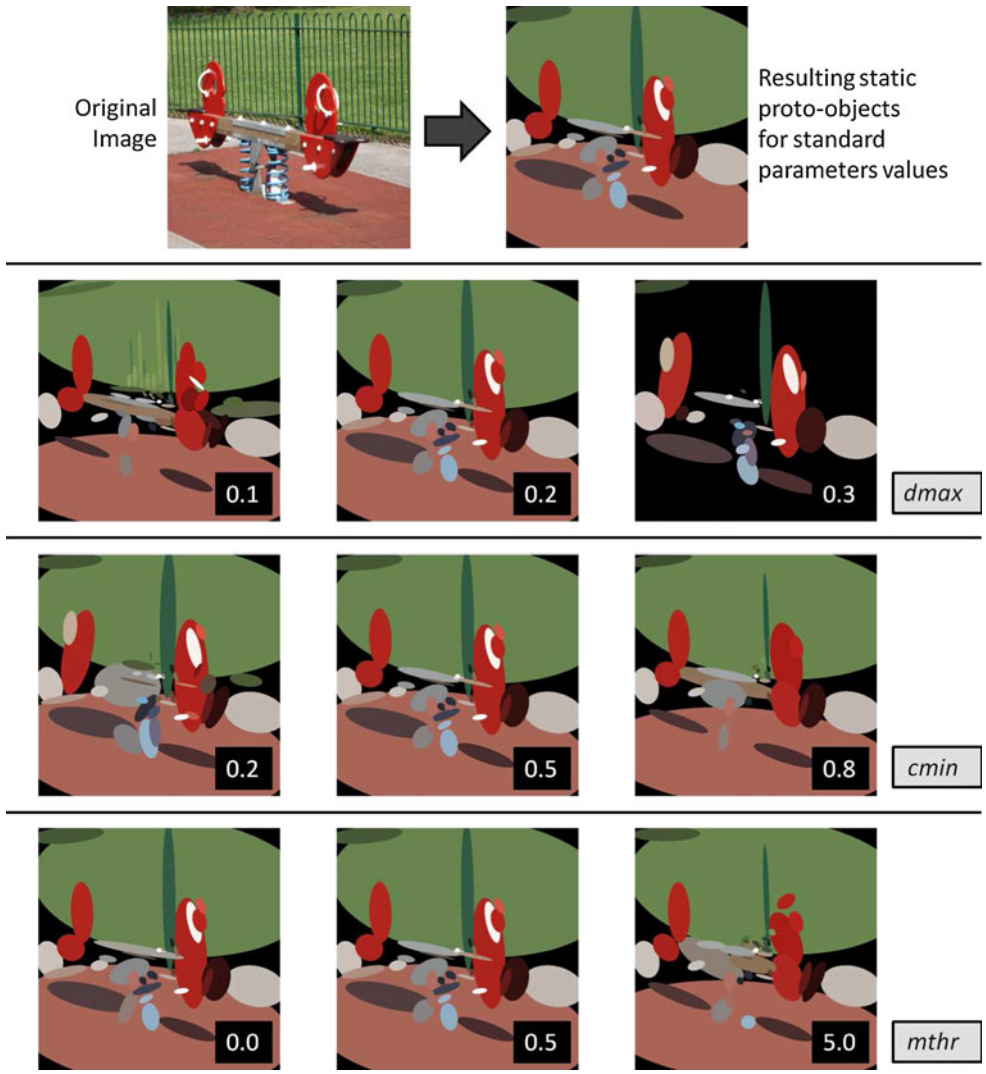

Fig. 5 Parameter variations. For an explanation of the parameters see Table 1. The values were varied as strongly as necessary to observe a clear difference to the outcome of the standard parameter setting. The image shows a playground scenario with a seesaw from the IIT-KGP Visual Saliency Data [22, 31, 32, 53] (http://www.facweb.iitkgp.ernet.in/∼jay/VS/Groundtruth.html). A useful proto-object representation of real world objects is characterized by (**a**) producing not too many proto-objects for a nearly homogeneously colored region (like the right seat of the seesaw). Too small values for d_{max} as well as too high values for m_{thr} yield an undesirable result. (**b**) producing not too many proto-objects for small regions, especially nearby the foveal area, like for too small values of d_{max} and c_{min} as well as for too high values for m_{thr}. (**c**) producing not too small proto-objects for larger regions (like the handle of the right seat of the seesaw) which is missing for too high values of c_{min} and m_{thr}. As a variation of the parameter m_{iter} has no effect on the proto-object configuration, it is skipped. In summarizing it can be stated that the proto-object detection algorithm works robustly because the modification of the parameters values only affects the outcome in detail but does not produce unreasonable results

however, the standard setting of Table 1 yields good results over a large range of input images (see Fig. 6). A further example of the overall performance after merging with the dynamic proto-objects is given in the results section.

Efficient Processing and the "Global Effect"

The inhomogeneity of the filter maps implies an implicit filtering mechanism because with decreasing resolution of the filters adjacent objects tend to become indistinguishable in the periphery of the visual field. This fusion effect in the periphery simulates the so-called "global effect" [20] in eye movement control: saccadic eye movements to two nearby objects in the periphery tend to land on the center of gravity of these objects. In our model, this corresponds to the landing of the gaze on a single proto-object, which is formed from the homogeneous answer of the low-resolution peripheral filter that covers both objects (see details in [69]).

In human vision, the "global effect" appears only under time pressure. In the model, the choice of the layer can

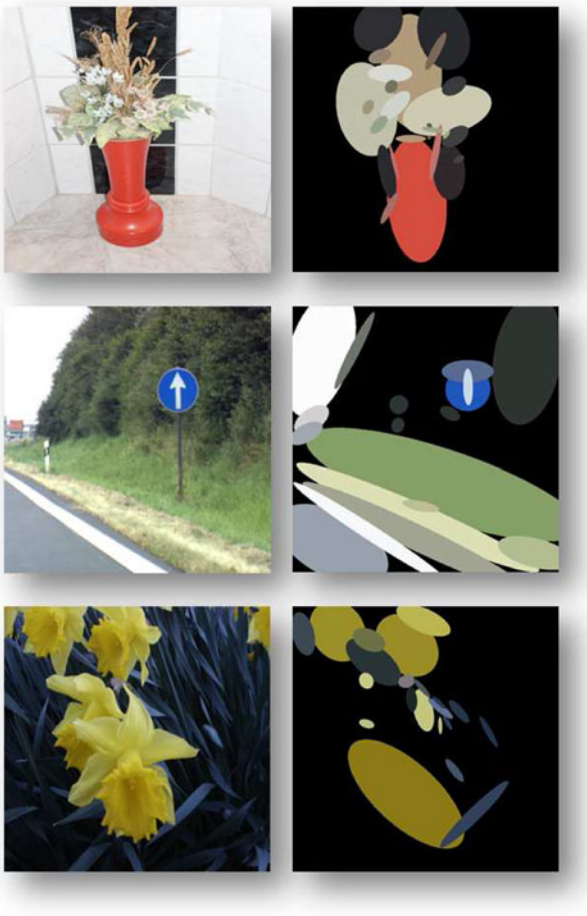

Fig. 6 Resulting proto-objects for different natural scenes using the standard parameters (see Table 1). The images come from the same image library as used in Fig. 5

simulate to what extent the system is under time pressure, because high-resolution layers take longer to be processed: the greater the time pressure, the lower the resolution of the chosen layer. This is also reasonable from the viewpoint of computational efficiency. The system effectively has to compute features only on the given resolution level and to generate the proto-objects w.r.t. that level. Computation of additional features for different resolutions can be restricted to the regions of those proto-objects that enter the attentional priority map after merging and further filtering on the proto-object stage.

The Dynamic Pathway

The extraction and processing of dynamic features in our architecture is described in this section and forms the equivalent of the dorsal stream of processing in the human visual system. Traditionally, motion is assumed to attain to the "where" pathway in human vision, hence defining spatial localizations required for action [26]. Nevertheless, there is also growing evidence that the ventral and dorsal streams share feedback connections to operate a figure-ground segmentation [17], which we functionally implement by a further information fusing stage that will be discussed in the section "Fusion of Ventral and Dorsal Proto-Objects".

Motion Feature Extraction

In our system, the motion sensing relies on an extension of the energy model by [1] that was introduced in [4]. The overall sketch of the motion processing pathway is depicted in Fig. 7. The basic idea is that coherent motion can be selected within an intensity frame buffer by filtering the diagonal oriented edges and bars produced by objects moving in the spatiotemporal volume and computing their energy (while vertical and horizontal edges correspond to static and flickering objects, respectively, see [68]). Energy features have been shown to represent a simple and biologically plausible model of how our visual system can identify features like edges and corners in static images. In these points, the energy function, given by the root square of the responses to quadrature pairs of linear oriented filters, squared and summed, has its local maxima [40].

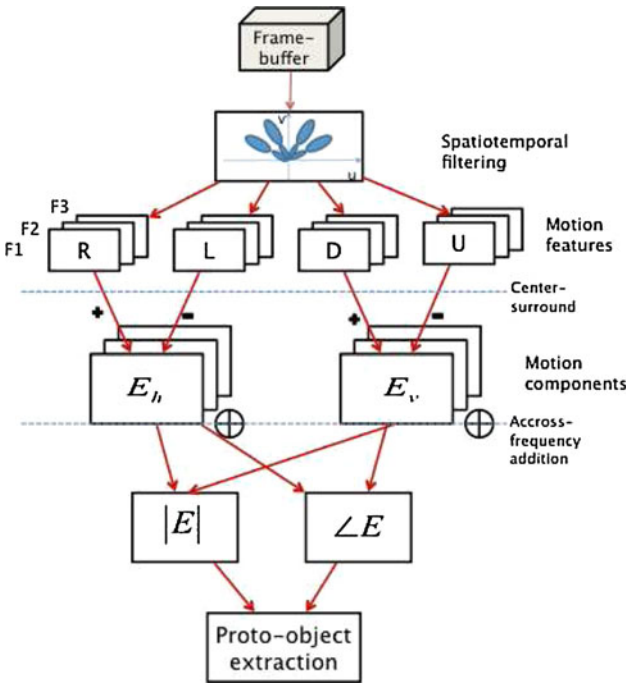

Fig. 7 The processing flow for motion extraction and moving object formation. First, a frame buffer is filtered by a purposely designed Gabor filter bank and direction-based feature maps are obtained. Afterward, horizontal and vertical components of motion energy are computed, and, from those, energy magnitude and phase are achieved. This allows extraction of proto-objects to be further merged with static objects and weighted for attentional selection

Equivalently, the energy can be used in spatiotemporal planes to extract moving edges.

To this end, we employ a Gabor filter bank to extract motion information at different spatio-temporal scales (3 frequency bands) and velocities/directions (4 filter orientations). Thereby we generate features to represent motion in the spatiotemporal frequency domain included in the window u, $v \in [0, 0.5]$ cyc/pixel, in order to comply with the sampling theorem. Gabor filters have long been known to resemble orientation sensitive receptive fields present in our visual cortex and to represent band-pass functions conveniently localized both in the space and in the frequency domain [13]. ICA (Independent Component Analysis) basis learning on natural image sequences has also produced receptive fields resembling 3D Gabor filters at different orientations and scales [28], similar to those of motion sensitive neurons [15]. 3D Gabor and log-Gabor features are established motion descriptors already used for optical flow and motion segmentation computations [18, 29]. Here, we code each voxel in a Gabor wavelet space, according to its oriented energy response to the 12 filters. 2D filters are convolved with horizontal and vertical spatio-temporal planes, in order to give a measure of rightward/leftward energy and upward/downward energy, respectively. Filtering is carried out on a bidimensional basis for the sake of computational load due to convolution operations. The frequency and orientation bandwidths determine the dimension of the filters and the central frequencies. Considering $s = h$ for filtering along the row-temporal (horizontal) planes, and $s = v$ for filtering along the column-temporal (vertical) planes, we obtain thus, for a given frame buffer, 24 3D feature maps, for some specific frequency and orientation bandwidths (in this case $b_f = 1$ octave and $b_\theta = 30°$):

$$E_{s,f,\theta}(x,y,t) \text{ where } \begin{cases} s = \{h, v\} \\ f = \{0.0938, 0.1875, 0.3750\} \\ \theta = \{\pi/6, \pi/3, 2/3\pi, 5/6\pi\} \end{cases}$$

Thereby, we tessellate the frequency domain so to span different ranges of velocities and spatiotemporal scales with a finite number of filters. Motion opponency is then used to recover direction by comparison of opponent filter pairs (i.e filters with same slope but opposite orientation, θ and $(\pi - \theta)$). Right-sensitive filters ($\theta_r = \{\pi/6, \pi/3\}$) span the second and the fourth quadrant in the frequency domain, while left-sensitive filters ($\theta_l = \{(\pi - \pi/6), (\pi - \pi/3)\}$) span the first and the third quadrant. A measure of the total rightward (leftward) energy at a specific frequency can hence be obtained by summing rightward (leftward) energy across velocities:

$$R_f = \sum_i \left| \frac{E_{h,f,\theta_{r_i}} - E_{h,f,\theta_{l_i}}}{E_{h,f,\theta_{r_i}} + E_{h,f,\theta_{l_i}}} \right|_{\geq 0} \tag{5}$$

$$L_f = \sum_i \left| \frac{E_{h,f,\theta_{r_i}} - E_{h,f,\theta_{l_i}}}{E_{h,f,\theta_{r_i}} + E_{h,f,\theta_{l_i}}} \right|_{\leq 0} \tag{6}$$

where the $|\cdot|$ operator selects points greater/less than zero, corresponding to rightward/leftward motion. The same can be done for upwards (downwards) energy computation, by taking $s = v, \theta_u = \theta_r$ and $\theta_d = \theta_r$. In this way we obtain 4 feature volumes R, L, U, D at different frequencies.

At this point, we have an effective motion feature detector, but no attentional modulation, as in other motion detection models. Therefore, we apply normalization and center-surround operators to the frames of each feature buffer. This is again motivated by the findings in the human visual system, where ganglion cells have been described as firing more strongly whenever a central location is more contrasted with respect to its surroundings [14]. This holds in the motion domain as well, as shown by [45] and [51], and it has been shown to occur at a very early stage even in some retina ganglion cells of rabbits, and probably of primates too [46]. Center-surround filtering is performed by taking the difference between each location and the mean of its neighborhood at different scales (see [23]).

Normalization to the same range and weighting according to the number of occurring local maxima is realized in a biological plausible manner by iteratively filtering the feature frames with a DoG (Difference of Gaussians) filter and taking each time just the non-negative values [31]. By doing so, feature maps with few activation peaks are enhanced, as most informative. Mono-directional features are then combined and summed across frequencies to obtain a measure of horizontal and vertical energy:

$$E_h = \sum_f (\mathcal{N}(CS(R_f)) + \mathcal{N}(CS(L_f))) \tag{7}$$

$$E_v = \sum_f (\mathcal{N}(CS(U_f)) + \mathcal{N}(CS(D_f))) \tag{8}$$

Here, $\mathcal{N}(\cdot)$ and $CS(\cdot)$ denote the normalization and center-surround operators, respectively, applied to each $x - y$ frame of the feature buffers.

E_h and E_v can be regarded as the projection on the x and y axes of the salient motion energy present in the frame buffer. Hence, from these components we can achieve, for every voxel, magnitude and phase of the salient energy:

$$|E(x,y,t)| = \sqrt{E_h(x,y,t)^2 + E_v(x,y,t)^2} \tag{9}$$

$$\angle E(x,y,t) = atan2(E_v(x,y,t), E_h(x,y,t)) \tag{10}$$

The energy magnitude represents the overall strength of the receptive fields responses to the moving stimulus, while the phase gives an idea of the stimulus direction on the 2D plane of the frame.

Proto-Object Construction and Selection

So far, we have shown how to obtain a saliency map enhancing relevant motion zones. Such a map is yet pixel-based and, in the perspective of perception for action, an object-based map would best help subsequent processing for object recognition and action selection. We need to evaluate the priority of an object as a whole and with respect to the surrounding background, not just by considering each single pixel it is composed of. Indeed, as said, attentional processes can modulate segregation and grouping of the visual input into "object tokens" across both the dorsal and the ventral pathways [54].

Proto-object patches can be extracted by relying only on motion features if we define them as blobs of consistent motion in terms of module and direction. This is consistent with the Gestalt law of common fate, stating that points moving with similar velocity and direction are perceptually grouped together in a single object. We take the middle frame of the filtered buffers (magnitude and phase), as the one containing the maximal response to the filters. Afterward, the magnitude map $|E(x, y)|$ is thresholded to discard points with too low energy. Null energy points are given a phase value outside the interval $(-\pi, \pi]$. The thresholding is done retaining only points having a saliency amount equal to a share θ_{energy} of the maximum. Increasing θ_{energy} from 10% up to 30% reduces the spreading of the objects due to center-surround operations but can also deliver just parts of a moving object. We chose a threshold of 0.2 as standard value, since it delivers most refined object regions without splitting objects or limiting the region to the pixels cumulating the maximum of Gabor responses. Effects of modulation of this parameter along with a static one are presented in the results section.

Finally, the mean shift algorithm is applied upon module, phase, and locations data. The mean shift algorithm is a kernel-based mode-seeking technique, broadly used for data clustering and segmentation [12]. Being non-parametric, it has the advantage that it does not need the number of clusters to be specified previously, even though a scale factor in the form of the dimension of the kernel window must be indicated. After segmentation, points lying in a connected neighborhood with conspicuous motion energy and coherent motion direction are clustered together.

To the aim of combination of dynamic and static objects, the clusters formed via Mean Shift segmentation can again be approximated by ellipses. Like previously described for the static proto-objects, we compute for every cluster the inertia matrix $\mathbf{I}$ from the covariance matrix and the mean $\boldsymbol{\mu}$ of the points forming the cluster. Each object $\mathbf{o}_d$ is further characterized by the mean energy of its points and the mean direction weighted w.r.t. the energy magnitude:

$$\mathbf{o}_d = \{\boldsymbol{\mu}, \mathbf{I}, \mathbf{x_d}\} \tag{11}$$

where $\mathbf{x_d} = \{|\bar{E}|, \angle\bar{E}\}$. An example of the whole processing from energy computation to object extraction is given in Fig. 8. The motion saliency system is tested again some sequences of the dataset for spatiotemporal saliency used in [37]. First, the motion saliency map is presented (Fig. 8, second column). The threshold for the motion energy allows to obtain clusters that most correspond to the real object shape (third column in the figure) and hence ellipsoids, which do not extend too much over the real object boundaries (fourth column). Ground truth (fifth column) is given by mask frames corresponding to foreground object segmentation manually annotated by human subjects. Even though the task for the subjects was different from that of attentional selection, where only significant moving things are retained, while uninteresting are discarded or inhibited, we computed a ROC curve on the motion saliency maps at different thresholds. Although our motion selection mechanism, indeed, does not distinguish between foreground and background motion or between self and relative motion, thus a high rate of false positives could be generated, the overall performance was quite good. In Fig. 9 it is shown that when most of the salience is retained, almost

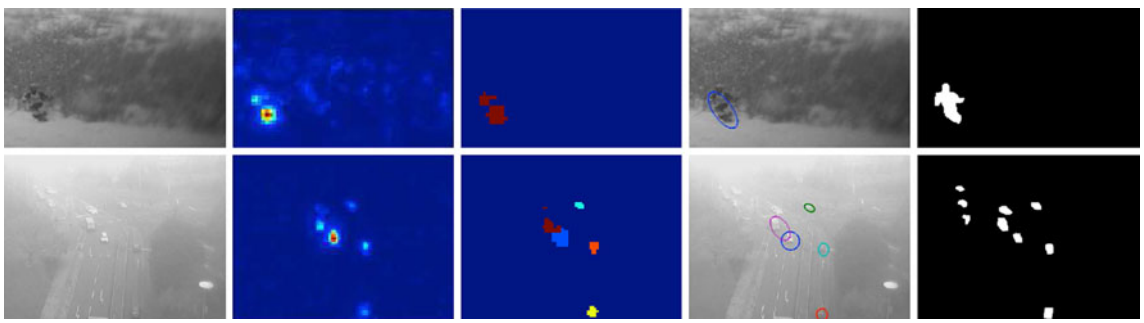

Fig. 8 Saliency maps and selected objects. Two sequences of the dataset presented in [37]. From the left, the central frame of the frame buffer, the corresponding saliency map, the segmentation results, and the extracted objects are shown. In the last column, the ground truth, as segmented by human subjects performing foreground object segmentation, is shown

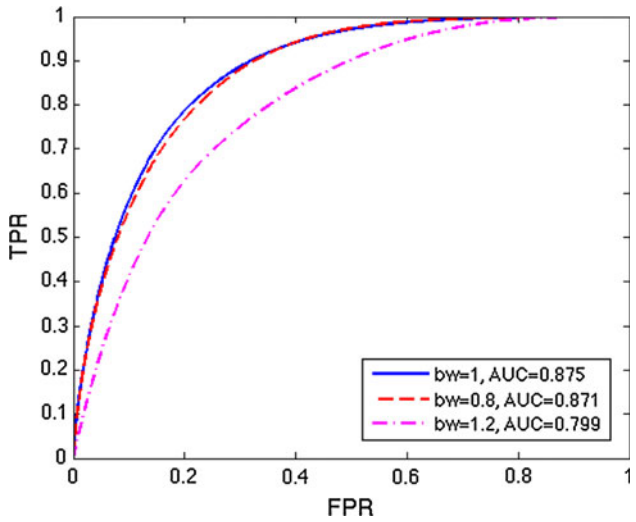

Fig. 9 ROC curves of the motion saliency maps against ground truth for the whole dataset of [37]. For different frequency bandwidths the system performs always reasonably, yet the best classification rate is obtained for a bandwidth of 1 octave

all the points selected by humans are also enhanced in the saliency map. The three curves were produced by taking different frequency bandwidths for the filter bank (namely 0.8, 1, and 1.2 octaves). The best classification performance, i.e an AUC (Area-Under-the-Curve) of 0.8751, was obtained by taking $b_f = 1$ octave; thus, we chose this value for the experiments in the following sections.

Fusion of Ventral and Dorsal Proto-Objects

After the formation of static and dynamic proto-objects, it remains to decide which of these candidates shall be evaluated by the TVA weight equation. This task lends itself to a more detailed investigation into the nature of proto-objects. In our model, proto-objects are homogeneous regions in the feature space, which are approximated by ellipses. A set of TVA features is assigned to each proto-object (see "Task Dependency by Means of TVA (Theory of Visual Attention)" for an overview). These features refer to the proto-objects as a whole. So there is no difference how the filter responses, e.g. for color, are distributed within a proto-object region - the model computes an arithmetic mean for every feature. This means, on the one hand, that our proto-objects are more than just a set of pixels extracted from a number of (possibly weighted) features, as usual in standard saliency models [39, 43], because they have an ellipsoid shape including a variety of filter responses for each feature. On the other hand, they are much coarser than required as input for proper object recognition algorithms, which have to analyze the data in a more detailed and therefore more time-consuming way (e.g. [50]). This status "in between" is characteristic as well as essential for proto-objects. The complexity level of proto-objects enables to quickly compute sufficiently complex hypotheses on real world objects. Hence, the idea of proto-objects is to optimize the trade-off between latency and accuracy of object-based covert visual attention and related saccadic eye movements. In a way, our proto-objects attain to the second scheme for perceptual representation proposed in [11], e.g. the one between "feature-placing" and "full-blooded objects". The merging of ventral and dorsal proto-objects has to respect this condition: computation still has to be fast and no information shall be lost. Our filtering and merging algorithm complies with these conditions in a two-stage process.

Stage 1

A moving object often consists of different color regions that are treated as separate proto-objects in the static processing stream. Similarly, just a single part of one object could be moving, while the rest is static. In this case, we can obtain multiple static proto-objects located within the region of one dynamic proto-object, or viceversa, one or more moving objects can lie within a bigger static object or overlap it substantially. Our model merges dynamic proto-objects with strongly overlapping static proto-objects, or viceversa, to a new elliptical dynamic proto-object. Afterward, all old proto-objects, which have been merged to new proto-objects, are deleted. That is, we discard proto-objects that represent only parts of a real world object and decrease the computational load by decreasing the overall number of proto-objects.

We devise two criteria, for merging overlapping proto-objects: First, if the center of a static/dynamic proto-object is located within the boundaries of a dynamic/static proto-object. Secondly, if the midpoint between the centers of both proto-objects is located within the larger proto-object and one proto-object is significantly smaller than the other:

$$\mathcal{A}(\mathbf{o}_1) < \mathcal{A}(\mathbf{o}_2) * th_1 \quad with \quad 0 < th_1 \ll 1 \tag{12}$$

where $\mathcal{A}(\cdot)$ denotes the area of the proto-object within the visual field and th_1 denotes the threshold parameter for stage 1.

Stage 2

Because of the merging process in stage one, some of the dynamic proto-objects may have changed their shape. This can lead to a strong overlap of two dynamic proto-objects. Because a real world object cannot have two different mean directions of motion in this overlapping region at the same time, one of the two proto-objects is deleted. Two criteria determine if a strong overlap exists. First, the center

of the dynamic proto-object A is located within the region of the proto-object B. Second, A is significantly smaller than B:

$$\mathcal{A}(\mathbf{o}_{d_1}) < \mathcal{A}(\mathbf{o}_{d_2}) * th_2 \quad with \quad 0 < th_2 \ll 1 \tag{13}$$

Due to our strong overlap criteria, it is likely that 1 and 2 refer to the same real world object. So we lose no information about the spatial formation of real world objects by deleting the dynamic proto-object 1. Moreover, we again decrease the computational load by decreasing the number of proto-objects. Figure 11 illustrates the mechanisms of both stages. An example, based on a sequence of natural frames, is shown in Fig. 10. Furthermore, in the result section is shown that this merging heuristic produces robust results in terms of the search task target, even for different parameter values used for the static and dynamic proto-object detection.

Finally, we obtain a combined map of ventral (static) and dorsal (dynamic) proto-objects. At this point, as the number of proto-objects and their elliptical shapes are fixed, the features for each proto-object can be computed, where the dynamic properties, energy and direction, are set to zero for all static proto-objects. For each proto-object, we obtain the geometric features location, size, shape and orientation. Geometric orientation here denotes the orientation of the principal axis whereas static orientation denotes an averaged response of all Gabor filters that are located within the region of a proto-object. Formally, each resulting proto-object comprises a center vector $\boldsymbol{\mu}$, an inertia matrix $\mathbf{I}$, a vector of dynamic properties $\mathbf{x_d}$, a vector of static properties $\mathbf{x_s}$ and a vector of geometric properties $\mathbf{x_g}$:

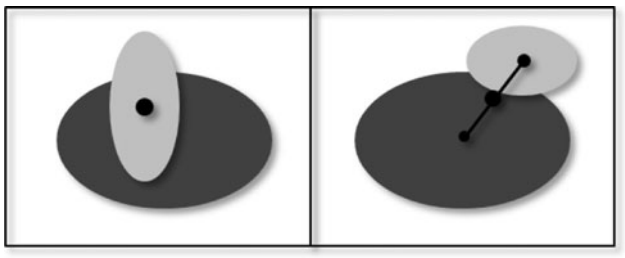

Fig. 11 Merging/deleting of proto-objects. *Left*: in stage one, two proto-objects are merged if the center of one proto-object (*light*) is located within the other (*dark*) and one of both proto-objects is static and the other one dynamic. In stage two, if both proto-objects are dynamic, the light proto-object would be deleted if it is significantly smaller than the dark one. *Right*: If we assume that one of both proto-objects is static and the other one is dynamic, then, if the midpoint of the two proto-object centers is located within the region of the bigger proto-object and the other proto-object is significantly smaller, both proto-objects are merged

$$\mathbf{o} = \{\boldsymbol{\mu}, \mathbf{I}, \mathbf{x_d}, \mathbf{x_s}\mathbf{x_g}\} \tag{14}$$

These remaining proto-objects serve as arguments for the subsequent TVA computations and thus as potential candidates for the next saccade landing point.

Task Dependency by Means of TVA (Theory of Visual Attention)

With the merging of proto-objects and the respective feature computation, all bottom-up computations are completed. The combined map of proto-objects provides a set $\mathbf{O}$ of static and dynamic proto-objects. Each element

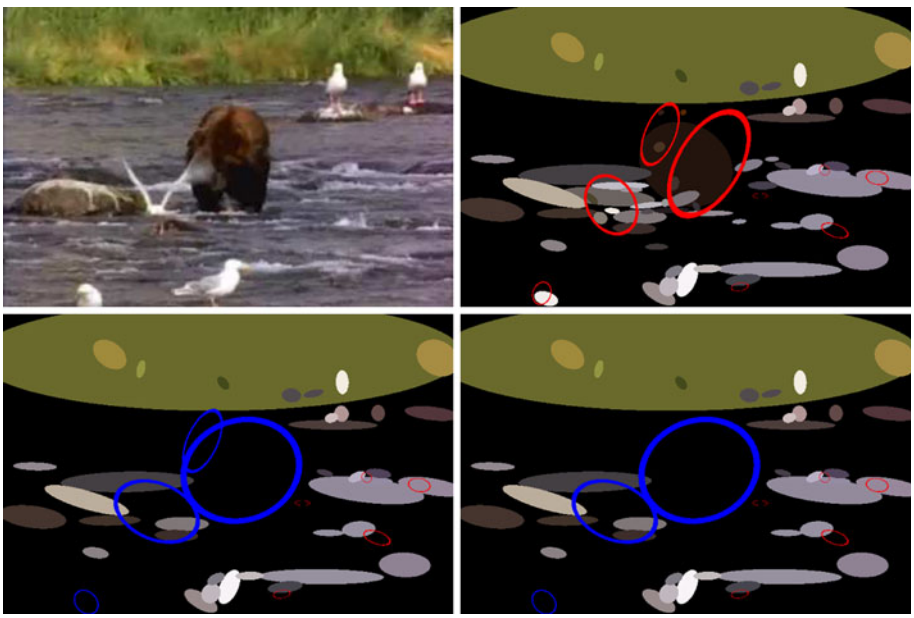

Fig. 10 Merging and filtering to obtain the combined map of proto-objects. *Top left*: a single frame of the input sequence. *Top right*: *Blobs* represent static proto-objects whereas ellipses represent dynamic proto-objects. *Bottom left*: Dynamic proto-objects have been merged with strong overlapping static proto-objects (*blue colored*). Bottom right: The smaller one of two dynamic proto-objects, which overlap strongly, is deleted

(proto-object) $\mathbf{o} \in \mathbf{O}$ consists of a set of properties serving as input for the subsequent TVA computations.

The Weight Equation

In the following step, the model computes a task-dependent attentional weight $w_{\mathbf{o}}$ for each proto-object based on a modified version of the *weight equation* of TVA [7]:

$$w_{\mathbf{o}} = \sum_{j \in R} \sigma(\mathbf{o}) \eta(\mathbf{o}, j) \pi_j = \sigma(\mathbf{o}) \sum_{j \in R} \eta(\mathbf{o}, j) \pi_j \quad (15)$$

Each $w_{\mathbf{o}}$ value is computed as the sum over all features j which are elements of R, the set of the task features. $\eta(\mathbf{o}, j)$, called the *sensory evidence*, denotes to what extent the proto-object $\mathbf{o}$ has the feature j weighted by top-down task-dependent pertinence π_j. The $\eta(\mathbf{o}, j)$ values thus restrict feature computation to proto-object regions, while the π_j implement a standard feature channel weighting as also present in other saliency models.

The modification of the TVA equation concerns the sensory evidence, which not only depends on the averaged filter output (the η value) but also on the size and location of a proto-object. The larger and the more foveally located a proto-object, the higher the sensory evidence. Based on the inhomogeneous structure of the static feature maps (see Fig. 2), both criteria (size and eccentricity) can likewise be quantified by the number of static filters located in the area of a proto-object, which leads to the σ value:

$$\sigma(\mathbf{o}) = f(\mathbf{o})^i \quad (16)$$

where $f(\mathbf{o})$ denotes the number of static filters and i determines how strong the number of these filters influences the sensory evidence. The influence can be eliminated by setting i to zero or we obtain a linear dependence if i equals one. We chose i to be 0.2.

Defining a TVA-Based Task

In total, our model provides nine different TVA features, which fall into three categories:

- static (low-level): color, orientation and intensity
- dynamic (low-level): energy and direction
- geometric (medium-level): location, size, shape and orientation

Formally, we define a task as a set of quadruples $T = \{j, \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j, \pi_j\}$. j denotes a task-relevant TVA feature (e.g. color or energy). $\boldsymbol{\mu}_j$ and $\boldsymbol{\Sigma}_j$ are mean and variance of a Gaussian, defining the value searched for and a tolerance interval. Two features, color and location, are represented by 2D variables; hence, in this case, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ are mean and variance of a bivariate Gaussian. The degree to which the proto-object $\mathbf{o}$ has the feature j is now expressed as

$$\eta(\mathbf{o}, j) = p(x_{\mathbf{o}_j} | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j), \quad (17)$$

where $p(\cdot | \boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ is the normalized Gaussian probability obtained by dividing the Gaussian function by its maximum; that is, the η values are confined to the range [0..1]. This is important because the weighting between features shall be limited to the pertinence values. The parameter $x_{\mathbf{o}_j}$ denotes the value of the feature j of object $\mathbf{o}$. This probabilistic computation of the evidence for "$\mathbf{o}$ having j" is motivated by experimental evidence showing that Gaussian tuning curves are in general a good approximation of cortical neurons' response behavior in the visual system [25] (Fig. 12).

To implement a certain task, some features and their pertinence values have to be selected to highlight the object the system is looking for. Each TVA feature can be used several times. For instance, the use of two location features j_{loc1} and j_{loc2} makes it possible to prioritize two different locations if the system is unsure where to find an object. We take only task-relevant features into account because all non-relevant features have a pertinence value of zero.

The variances $\boldsymbol{\Sigma}_j$ determine how accurately the search is performed. A small variance implies to exactly search for "that red" given by the mean value. Such a strict search carries the risk of getting a very low attentional weight for all proto-objects making the system too sensitive to noise

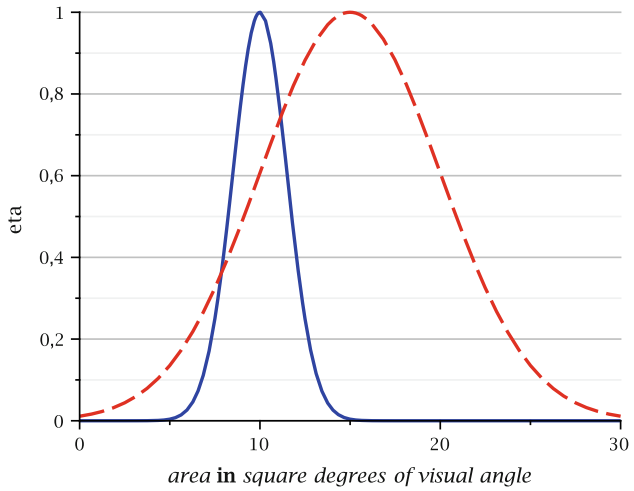

Fig. 12 An example of η-value computation. The figure shows a solid blue ($\mu = 10; \sigma = 1.5$) and a dashed red ($\mu = 15$; $\sigma = 5$) one-dimensional Gaussian each reflecting a size feature of TVA. The measure for size is the area a proto-object covers within the visual field in square degree of visual angle. Both Gaussians are divided by their maximum to obtain a range of [0..1]. Now, the η-values can easily be computed as $\eta(\mathbf{o}, size) = p(x_{\mathbf{o}_{size}} | 10, 1.5)$ and $\eta(\mathbf{o}, size) = p(x_{\mathbf{o}_{size}} | 15, 5)$ where $x_{\mathbf{o}_{size}}$ is the size feature value of object $\mathbf{o}$

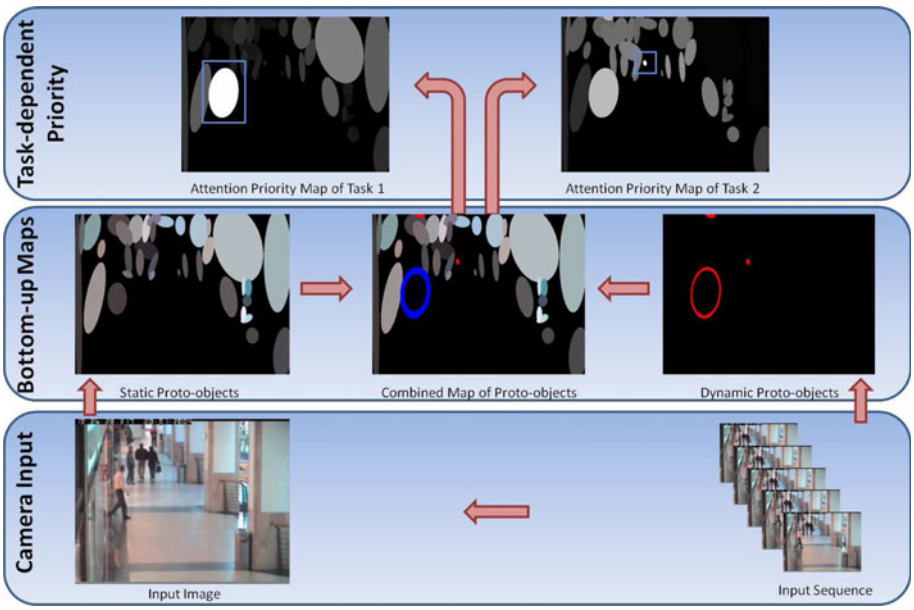

Fig. 13 An example of the model data flow - starting with the input sequence up to the task-dependent priority maps. The rectangle in the attention priority maps marks the proto-object with highest priority

in the input data. On the other hand, too large variance makes it difficult for the system to distinguish between relevant and non-relevant proto-objects and therefore also produces an undesired behavior.

Finally, the pertinence values have to be defined. As the absolute value of the attentional weights $w_{\mathbf{o}}$ is of no relevance, the pertinence values can be chosen on a relative scale like $\pi_{size} = 2 * \pi_{color}$.

The Attention Priority Map

The computed $w_{\mathbf{o}}$ values of the weight equation (15) are stored in a retinotopically organized *attention priority map* (APM). The proto-object with the highest attentional weight (priority) within this map serves as next camera saccade target. After the saccade has been executed, the next processing cycle can be started.

Results

In this section, we present some results by elucidating the whole computational process step by step. For illustration, we first use a sequence from the Caviar surveillance data set.[2] This real world sequence provides rich and typical data: there are people moving or staying and different objects at different scales. Even without a particular task, our gaze orienting system must therefore deal with multiple stimuli competing for selection. In the selected sequence, a man leaves a shop while some other people are walking in the distance.

The whole process consists of the following steps as shown in Fig. 13:

- *Image buffer acquisition*. n frames are acquired and buffered. The middle frame is fed into the static feature processing thread, while the whole buffer is used in the dynamic feature processing thread.
- *Static feature processing*. The input frame undergoes foveation. Color, intensity and orientation responses are computed on the foveated image. The color and intensity feature maps of one layer provide the input for the subsequent blob detection algorithm. Size filtering delivers the most plausible blobs as static proto-objects. In our example sequence, one brown thick ellipse is produced for the man in foreground, while the three small dark ellipses on the background correspond to the three people further away.
- *Dynamic feature processing*. The whole frame buffer is spatio-temporally filtered by the Gabor filter bank. Directional motion energy features are used to enhance locations of conspicuous motion. Proto-object formation is obtained via segmentation upon energy and direction on the last frame of the buffer. Dynamic blobs are labeled with their mean energy and direction. One bigger blob is produced for the man in foreground, while a smaller ellipse corresponds to the bag of the woman moving away. Just these two objects are distinguished because of their motion contrast with respect to the surroundings.
- *Object combination and fusion*. At this point, static and dynamic proto-objects possibly corresponding to the same entity are merged. This delivers a combined map, where a single bigger ellipse has subsumed the static and the dynamic ones corresponding to the man leaving

[2] http://groups.inf.ed.ac.uk/vision/CAVIAR/CAVIARDATA1/.

Table 2 The task parameters for the surveillance camera sequence presented in Fig. 13

	Task 1			Task 2		
	μ	σ	π	μ	σ	π
Intensity	0.9	0.3	1	–		
Energy	40	10	2	30	10	2
Area	5	2.5	1	0.33	2.5	1
Orientation	90	30	1	—		

Both thresholds th_1 and th_2 of the combining stage are equal to 0.1

Table 3 The task parameters for the football stadium sequence presented in Fig. 14, with $th_1 = th_2 = 0.1$

	Task 1			Task 2		
	μ	σ	π	μ	σ	π
Intensity	–	–	–	0.0	0.3	1
Color	1.0/0	0.5/0.5	1	–	–	–
Area	0.6	0.4	1	0.1	0.05	1
Energy	30	10	2	–	–	–

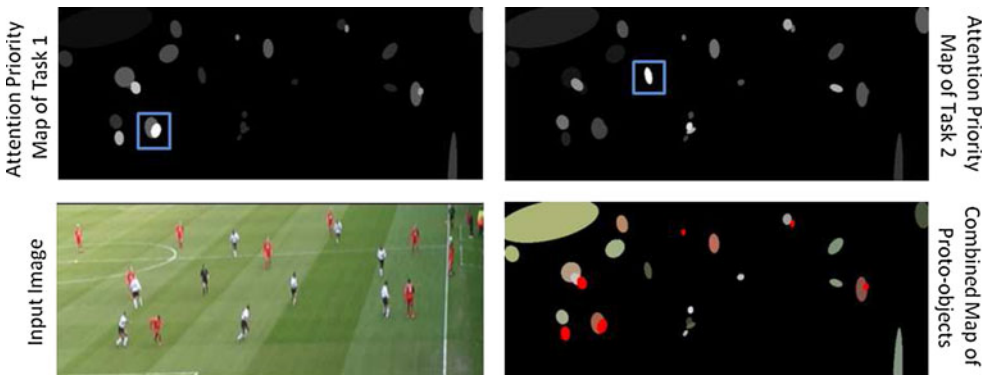

Fig. 14 The relevant data flow images of a sequence recorded in a football stadium. We cut the lower half of the images to accentuate the important areas. The labels are identical to those in Fig. 13

the shop. Each object is labeled with its features, static, dynamic and geometric.

- *TVA weighting computation and APM production*. By setting a task, different sets of features can now be taken into account to compute the weights of the proto-objects according to TVA. From the same combined map, hence, different priority maps can be generated. For example, in task 1, we set the Gaussian parameters of the sensory evidence for the features intensity, energy, area and orientation. Proto-objects ranking is visualized through luminosity; hence, the man in foreground wins the competition. In the second case, right on the top of Fig. 13, the task was defined upon energy and area, so the blob of the swinging bag obtains the highest priority. Table 2 shows the task parameters in detail.

In Fig. 14, a second sequence is presented, taken from a fixed camera in a football stadium[3]. The scene shows a throw-in for the red team and players trying to get free from their opponents. In task 1, we wanted to find the player who was more likely to get the ball, hence the reddish one with a consistent energy amount, that is red color and energy get a high priority. In the second case, the task was defined so to find the referee, that is, motion is not a relevant feature anymore, whereas intensity gets a high priority, since the referee is usually dressed in black. Table 3 was used to define the task parameters.

A further sequence is presented in Fig. 15. The sequence stems from the dataset presented in [2][4] for crowd flow segmentation. Here, different sized vehicles, cyclists and pedestrians are moving in a crossroad. We tested robustness of our system by varying parameter configuration. As discussed in previous sections, d_{max} and θ_{energy} are the most critical parameters for proto-object formation. Higher values of both parameters lead to smaller and fewer objects as depicted in the right lower part of the maps matrix presented in Fig. 15. The given task is formalized in the table of the same figure and corresponds to "look for a bright moving van"; that is, the target is characterized by two low-level features (one static and one dynamic) and by one medium-level feature (area). In two thirds (16/24) of the cases, the winning proto-object refers to the same real world object as the proto-object in the central map that was produced with the standard parameter setting. Also, the remaining cases show similar results by selecting other bright moving cars. So the model shows robustness as its functionality is not limited to our standard parameters.

[3] testing, camera 3, ftp://ftp.pets.rdg.ac.uk/pub/VS-PETS/.

[4] http://www.cs.cmu.edu/~saada/Projects/CrowdSegmentation/.

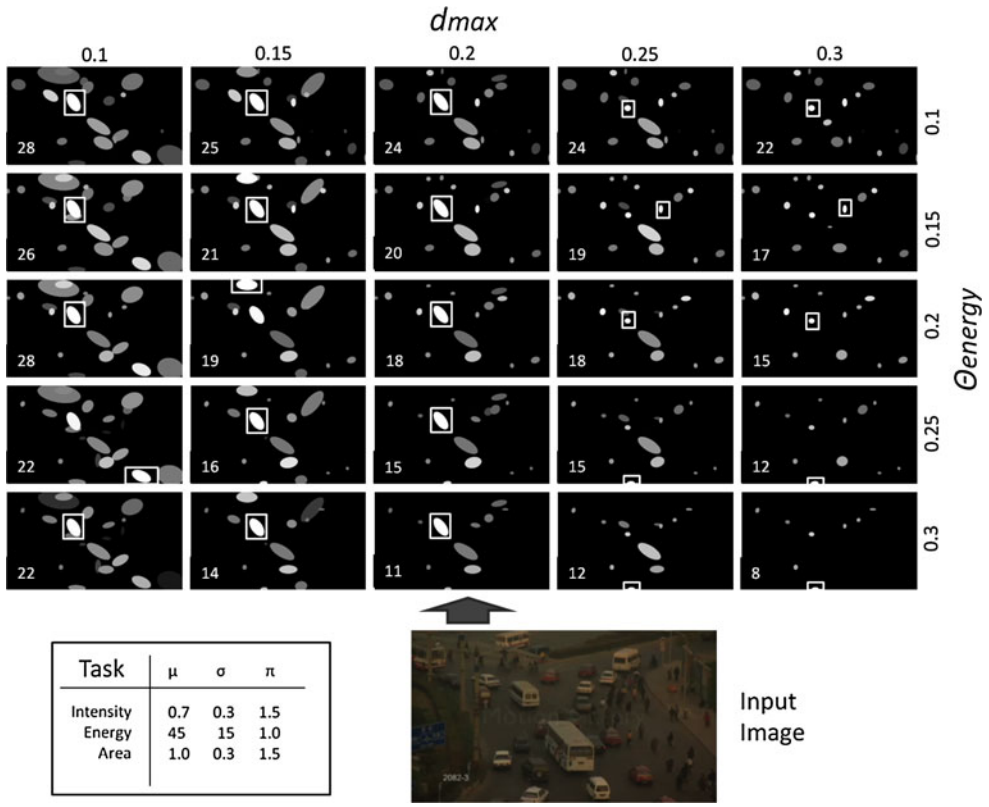

Fig. 15 Resulting attention priority maps depending on the values of the parameters d_{max} (static) and θ_{energy} (motion). Each parameter was varied fivefold. The central map of the 5×5 matrix shows the outcome for the standard parameters. In the left-bottom corner of every map, the total number of present proto-objects is indicated. The winning proto-object is highlighted by a rectangular box. The task consists of two low-level (one static, one dynamic) and one medium-level feature. In the given scenario, the task definition complies with the search for a bright moving van. The results are relatively robust against even strong changes of the parameters. The parameter setting was discussed in "Static Features and Proto-Objects" and "The Dynamic Pathway"

Discussion

We introduced a computational model of attention that is strongly inspired and constrained by experimental findings and theories from neuroscience and psychology (for overviews, see [8, 54, 60]) and that contains a number of novel ingredients and novel combinations of known ingredients.

First, inhomogeneous processing of visual features is a main feature of the primate visual system [63] that has been hardly implemented in computational models of attention. Exceptions are presented in [33, 47, 57]. Humans and other primates move their eyes several times per second because the retina is not homogeneous and potential interesting objects of the environment have to be analyzed by the fovea, the high-resolution part of the retina. A peripheral representation of an object is often not sufficient for efficient object recognition. Inhomogeneous processing in our model extends beyond the visual feature level up to the level of proto-objects within the attentional priority map.

Second, most computational models of attention are restricted to processing of static visual features and following priority computation (e.g., [23, 32, 43]). Recently, dynamic features such as motion have been modeled within the saliency map framework [4, 6, 30, 37] and a very few combinations of static and dynamic features for determining saliency and attentional priorities have been devised [36, 38] on a pixel-wise basis.

Third, the integration of static and dynamic features occurs in our model at the level of proto-objects that are used for determining attentional priorities in an object-based way. Despite converging evidence for the necessity of such an object-based account of visual attention, and more precisely, for the necessity of an attentional priority (saliency) map that contains medium-level visual proto-objects [8, 55], just a few computational models implemented this important ingredient of attentional control [47, 57, 65]. Our model introduces and implements proto-object computations in a number of novel ways. Proto-objects refer to a medium level within the hierarchy of the visual system at which dynamic and static feature processing are combined for the first time for a common priority-based representation of the visual environment. Moreover, computation of these proto-objects implies a new type of medium-level visual features, that is, location of

proto-objects within a priority map, their size and rough shape including orientation of the principal axis of the ellipsoids. Rough shape means, for instance, that a circular object can be distinguished from an elongated one.

Fourth, a further new aspect of proto-object computation refers to modeling attentional processes according to the well-established and prominent "Theory of visual Attention" [7, 9] that implies a relatively sophisticated form of task-based control. TVA presupposes that attentional priorities for perceptual processing are computed at the level of proto-objects within a priority (saliency) map. A pixel-based representation of attentional priorities that is standard in most computational models would not allow to implement such an object-based account of attention. Object-based attention means in TVA that attentional weights are computed for proto-objects. The weight determines the degree of priority of these objects and their features in perceptual processing. Our model adds the assumption that the proto-object with the highest weight will be the target of the next saccade [10, 69]. Following TVA, attentional weights depend on bottom-up influences such as the sensory evidence for visual features and on top-down influences such as the current task [7]. Weights are represented in an attentional priority (saliency) map [9]. In terms of a computational model of attention, the restriction of visual feature and weight computation to regions of proto-objects is computationally efficient and in contrast to pixel-based saliency maps. Moreover, following [54], our model assumes that the proto-object with the highest attentional weight receives highest priority in perceptual processing and simultaneously becomes the target for the next saccade or camera shift (see, also, [16]).

Importantly, the novel medium-level features of proto-objects such as size or rough shape allows a more sophisticated form of task-based control of attention. Traditional computational models of attention (e.g., [5, 39, 43]) allow only to specify the current task at the level of low-level basic visual features such as color, motion, orientation. Depending on the task and the corresponding search target, these basic feature channels are weighted. For instance, when the task is to find a human face, then those color channels that match skin color are weighted higher than other channels. Our model adds a further and novel level of task-based control of selective vision by computing medium-level features of proto-objects such as size or rough shape of the proto-object. Consequently, more complex task-dependent search tasks can be specified and these are able to influence the computation of attentional priorities for perception and sensori-motor actions such as saccadic eye movements. Specifying the size or the rough shape of an object – in our model in terms of the ratio of the axes of the ellipsoids (e.g, distinguishing a circle from a stripe) – adds a further constraint for finding task-relevant information. The search task of Fig. 15 (traffic scene with 3 search features and 2 parameters varied) implemented such a scenario by involving size as a further task-relevant medium-level feature. Models that rely on low-level features could not use this size information and will therefore show a less efficient search behavior.

The results and examples show that despite its apparent complexity the overall architecture delivers very good results with respect to a large variety of real world images and image sequences for standard parameters, where we have resorted to benchmark sequences provided by several other authors. The main parameter sensitivity is hidden in the clustering algorithm that is the basis for proto-object formation. In principle, any clustering scheme that delivers compact regions approximable by ellipsoids may be used and future work could aim at learning a respective function from experience. However, we have used previously proposed standard algorithms that have not been specifically tailored toward our problem, which again shows the robustness of the overall approach. Our answer to "Where to look next ?": to the proto-object with the highest TVA-based attentional priority.

Acknowledgments This research was supported by grants of the Cluster of Excellence - Cognitive Interaction Technology (CITEC).

References

1. Adelson EH, Bergen JR. Spatiotemporal energy models for the perception of motion. J Opt Soc Am A. 1985;2(2):284–99.
2. Ali S, Shah M. A lagrangian particle dynamics approach for crowd flow segmentation and stability analysis. In: IEEE conference on computer vision and pattern recognition, 2007. CVPR '07. 2007. p. 1–6.
3. Aziz M, Mertsching B. Fast and robust generation of feature maps for region-based visual attention. IEEE Trans Image Process. 2008;17(5):633 –44.
4. Belardinelli A, Pirri F, Carbone A. Motion saliency maps from spatiotemporal filtering. Attention in Cognitive Systems 2009. p. 112–23.
5. Breazeal C, Scassellati B. A context-dependent attention system for a social robot. In: IJCAI '99. San Francisco: Morgan Kaufmann Publishers Inc.; 1999. p. 1146–53.
6. Bruce NDB, Tsotsos JK. Saliency, attention, and visual search: an information theoretic approach. J Vis. 2009;9(3), 1–24.
7. Bundesen C. A theory of visual attention. Psychol Rev. 1990;97(4):523–47.
8. Bundesen C, Habekost T. Principles of visual attention: linking mind and brain. Oxford: Oxford University Press; 2008.
9. Bundesen C, Habekost T, Kyllingsbaek S. A neural theory of visual attention: bridging cognition and neurophysiology. Psychol Rev. 2005;112(2):291–328.
10. Carbone E, Schneider WX. Gaze is special: the control of stimulus-driven saccades is not subject to central, but visual attention limitations. Atten Percept Psychophys. (in press).
11. Clark A. Feature-placing and proto-objects. Philos Psychol. 2004;17(4):443+.

12. Comaniciu D, Meer P. Mean shift: a robust approach toward feature space analysis. IEEE Trans Pattern Anal Mach Intell. 2002;24(5):603–19.
13. Daugman JG. Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. J Opt Soc Am A. 1985;2(7):1160–9.
14. De Monasterio FM, Gouras P. Functional properties of ganglion cells of the rhesus monkey retina. J Physiol. 1975;251(1): 167–95.
15. DeAngelis GC, Ohzawa I, Freeman RD. Spatiotemporal organization of simple-cell receptive fields in the cat's striate cortex. i. general characteristics and postnatal development. J Neurophysiol. 1993;69(4):1091–117.
16. Deubel H, Schneider WX. Saccade target selection and object recognition: evidence for a common attentional mechanism. Vis Res. 1996;36(12):1827–37.
17. Domijan D, Šetić M. A feedback model of figure-ground assignment. J Vis. 2008;8(7):1–27.
18. Dosil R, Fdez-Vidal XR, Pardo XM. Motion representation using composite energy features. Pattern Recognit. 2008;41(3):1110–23.
19. Driscoll J II, RP Cave K. A visual attention network for a humanoid robot. In: Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, 1998. p. 12–6.
20. Findlay JM. Global visual processing for saccadic eye movements. Vis Res. 1982;22(8):1033–45.
21. Forssén PE. Low and medium level vision using channel representations. Ph.D. thesis, Linköping University, Sweden, SE-581 83 Linköping, Sweden (2004). Dissertation No. 858, ISBN 91-7373-876-X.
22. Frey HP, Konig P, Einhauser W. The role of first- and second-order stimulus features for human overt attention, perception and psychophysics. Percept Psychophys. 2007;69(2):153–61.
23. Frintrop S, Klodt M, Rome E. A real-time visual attention system using integral images. In: Proceedings of the 5th international conference on computer vision systems (2007).
24. Frintrop S, Rome E, Christensen HI. Computational visual attention systems and their cognitive foundations: a survey. ACM Trans Appl Percept. 2010;7(1):1–39.
25. Geisler WS, Albrecht DG. Visual cortex neurons in monkeys and cats: detection, discrimination, and identification. Vis Neurosci. 1997;14:897–919.
26. Goodale MA, Milner AD. Separate visual pathways for perception and action. Trends Neurosci. 1992;15(1):20–5.
27. Goodale MA, Westwood DA. An evolving view of duplex vision: separate but interacting cortical pathways for perception and action. Curr Opin Neurobiol. 2004;14(2):203–11.
28. van Hateren JH, Ruderman DL. Independent component analysis of natural image sequences yields spatio-temporal filters similar to simple cells in primary visual cortex. Proc Biol Sci. 1998;265(1412):2315–20.
29. Heeger DJ. Optical flow using spatiotemporal filters. Int J Comput Vis. 1988;1(4):279–302.
30. Itti L, Baldi P. Bayesian surprise attracts human attention. Vis Res. 2009;49(10):1295–306.
31. Itti L, Koch C. Feature combination strategies for saliency-based visual attention systems. J Electron Imag. 2001;10(1):161–9.
32. Itti L, Koch C, Niebur E. A model of saliency-based visual attention for rapid scene analysis. IEEE Trans Pattern Anal Mach Intell. 1998;20(11):1254–9.
33. Kehrer L, Meinecke C. A space-variant filter model of texture segregation: parameter adjustment guided by psychophysical data. Biol Cybern. 2003;88(3):183–200.
34. Koch C, Ullman S. Shifts in selective visual attention: towards the underlying neural circuitry. Hum Neurobiol. 1985;4(4):219–27.
35. Land M, Tatler B. Looking and acting: vision and eye movements in natural behaviour. Oxford: Oxford University Press; 2009.
36. Le Meur O, Le Callet P, Barba D. Predicting visual fixations on video based on low-level visual features. Vis Res. 2007;47(19): 2483–98.
37. Mahadevan V, Vasconcelos N. Spatiotemporal saliency in dynamic scenes. IEEE Trans Pattern Anal Mach Intell. 2009;32: 171–7.
38. Marat S, Ho Phuoc T, Granjon L, Guyader N, Pellerin D, Guérin-Dugué A. Modelling spatio-temporal saliency to predict gaze direction for short videos. Int J Comput Vis. 2009;82(3):231–43.
39. Moren J, Ude A, Koene A, Cheng G. Biologically based top-down attention modulation for humanoid interactions. Int J HR. 2008;5(1):3–24.
40. Morrone MC, Burr DC. Feature detection in human vision A phase-dependent energy model. Proc R Soc Lond B Biol Sci. 1988;235(1280):221–45.
41. Nagai Y. From bottom-up visual attention to robot action learning. In: Proceedings of 8 IEEE international conference on development and learning. IEEE Press; 2009.
42. Nagai Y, Hosoda K, Morita A, Asada M. A constructive model for the development of joint attention. Conn Sci. 2003;15(4): 211–29.
43. Navalpakkam, V, Itti L. An integrated model of top-down and bottom-up attention for optimal object detection. In: Proceedings of IEEE conference on computer vision and pattern recognition (CVPR), New York, NY. 2006. p. 2049–56.
44. Navalpakkam V, Itti L. A goal oriented attention guidance model. In: Biologically Motivated Computer Vision, pp. 81–118. Springer (2010).
45. Nothdurft H. The role of features in preattentive vision: comparison of orientation, motion and color cues. Vis Res. 1993; 33(14):1937–58.
46. Olveczky Bence P, Baccus SA, Meister M. Segregation of object and background motion in the retina. Nature. 2003;423:401–8.
47. Orabona F, Metta G, Sandini G. A proto-object based visual attention model. In: Attention in cognitive systems. Theories and systems from an interdisciplinary viewpoint. 2008. p. 198–215.
48. Palmer SE. Vision science. Cambridge: MIT; 1999.
49. Park S, Shin J, Lee M. Biologically inspired saliency map model for bottom-up visual attention. In: Biologically motivated computer vision. Springer; 2010. p. 113–45.
50. Riesenhuber M, Poggio T. Hierarchical models of object recognition in cortex. Nat Neurosci. 1999;2(11):1019–25.
51. Rosenholtz R. A simple saliency model predicts a number of motion popout phenomena. Vis Res. 1999;39(19):3157–63.
52. Ruesch J, Lopes M, Bernardino A, Hornstein J, Santos-Victor J, Pfeifer R. Multimodal saliency-based bottom-up attention a framework for the humanoid robot icub. In: International conference on robotics and automation, Pasadena, CA, USA. 2008. p. 962–7.
53. Schaefer G, Stich M. UCID - An Uncompressed Colour Image Database. In: Storage and retrieval methods and applications for multimedia 2004. Proceedings of SPIE, vol. 5307. 2004. p. 472–80.
54. Schneider WX. VAM: A neuro-cognitive model for visual attention control of segmentation, object recognition, and space-based motor action. Vis Cogn. 1995;2(2–3):331–76.
55. Scholl BJ. Objects and attention: the state of the art. Cognition. 2001;80(1–2):1–46.
56. Steil, JJ, Heidemann G, Jockusch J, Rae R, Jungclaus N, Ritter, H.: Guiding attention for grasping tasks by gestural instruction: The gravis-robot architecture. In: Proceedings IROS 2001, IEEE 2001. p. 1570–7.
57. Sun Y, Fisher R, Wang F, Gomes HM. A computer vision model for visual-object-based attention and eye movements. Comput Vis Image Underst. 2008;112(2):126–42.
58. Tatler B (2009) Current understanding of eye guidance. Vis Cogn. 777–89.

59. Torralba A, Oliva A, Castelhano MS, Henderson JM. Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. Psychol Rev. 2006;113(4):766–86.
60. Treisman A. The binding problem. Curr Opin Neurobiol. 1996;6(2):171–8.
61. Treisman AM, Gelade G. A feature-integration theory of attention. Cogn Psychol. 1980;12(1):97–136.
62. Tsotsos JK, Culhane SM, Winky WYK, Lai Y, Davis N, Nuflo F Modeling visual attention via selective tuning. Artif Intell. 1995;78(1–2):507–45.
63. Van Essen D, Anderson C. Information processing strategies and pathways in the primate visual system. In: Zornetzer S, Davis J, Lau C, McKenna T (eds.), An introduction to neural and electronic networks. Academic Press, New York; 1995. p. 45–76.
64. Walther D, Itti L, Riesenhuber M, Poggio T, Koch C. Attentional selection for object recognition—a gentle way. In: Biologically motivated computer vision, Springer; 2002. p. 251–67.
65. Walther D, Koch C. Modeling attention to salient proto-objects. Neural Netw. 2006;19(9):1395–407.
66. Watson AB. Detection and recognition of simple spatial forms. Technical report, NASA Ames Research Center; 1983.
67. Watson AB, Albert Jr J. Model of human visual-motion sensing. J Opt Soc Am A. 1985;2(2):322–41.
68. Wildes RP, Bergen JR. Qualitative spatiotemporal analysis using an oriented energy representation. In: ECCV '00: Proceedings of the 6th European conference on computer vision-part II. 2000. p. 768–84.
69. Wischnewski M, Steil JJ, Kehrer L, Schneider WX. Integrating inhomogeneous processing and proto-object formation in a computational model of visual attention. In: Human centered robot systems. 2009. p. 93–102.
70. Wolfe JM, Horowitz TS. What attributes guide the deployment of visual attention and how do they do it? Nat Rev Neurosci. 2004;5(6):495–501.