Available online at www.sciencedirect.com

SCIENCE DIRECT®

Vision Research 45 (2005) 205–231

Vision Research

www.elsevier.com/locate/visres

Modeling the influence of task on attention

Vidhya Navalpakkam, Laurent Itti *

Departments of Computer Science, Psychology and Neuroscience Graduate Program, University of Southern California, Hedco Neuroscience Building, Room 30A, Mail Code 2520, 3641 Watt Way, Los Angeles, CA 90089-2520, USA

Received 6 October 2003; received in revised form 19 March 2004

Abstract

We propose a computational model for the task-specific guidance of visual attention in real-world scenes. Our model emphasizes four aspects that are important in biological vision: determining task-relevance of an entity, biasing attention for the low-level visual features of desired targets, recognizing these targets using the same low-level features, and incrementally building a visual map of task-relevance at every scene location. Given a task definition in the form of keywords, the model first determines and stores the task-relevant entities in working memory, using prior knowledge stored in long-term memory. It attempts to detect the most relevant entity by biasing its visual attention system with the entity's learned low-level features. It attends to the most salient location in the scene, and attempts to recognize the attended object through hierarchical matching against object representations stored in long-term memory. It updates its working memory with the task-relevance of the recognized entity and updates a topographic task-relevance map with the location and relevance of the recognized entity. The model is tested on three types of tasks: single-target detection in 343 natural and synthetic images, where biasing for the target accelerates target detection over twofold on average; sequential multiple-target detection in 28 natural images, where biasing, recognition, working memory and long term memory contribute to rapidly finding all targets; and learning a map of likely locations of cars from a video clip filmed while driving on a highway. The model's performance on search for single features and feature conjunctions is consistent with existing psychophysical data. These results of our biologically-motivated architecture suggest that the model may provide a reasonable approximation to many brain processes involved in complex task-driven visual behaviors.

Keywords: Attention; Top-down; Bottom-up; Object detection; Recognition; Task-relevance; Scene analysis

1. Introduction

There is an interesting diversity in the range of hypothetical internal scene representations, including the *world as an outside memory* hypothesis that claims no photographic memory for visual information (O'Regan, 1992), the *coherence theory* according to which only one spatio-temporal structure or coherent object can be represented at a time (Rensink, 2000), a limited memory of three or four objects in visual short-term memory (Irwin & Andrews, 1996; Irwin & Zelinsky, 2002), and finally, memory for many more previously attended objects in visual short-term and long-term memory (Hollingworth, 2004; Hollingworth & Henderson, 2002; Hollingworth, Williams, & Henderson, 2001). Together with studies in change detection (Kanwisher, 1987; Rensink, 2000, 2002; Rensink, O'Regan, & Clark, 1997; Watanabe, 2003), this suggests that internal scene representations do not contain complete knowledge of the scene. To summarize, instead of attempting to segment, identify, represent and maintain detailed memory of all objects in a scene, there is mounting evidence that our brain

* Corresponding author. Tel.: +1 213 740 3527; fax: +1 213 740 5687.

E-mail addresses: navalpak@usc.edu (V. Navalpakkam), itti@usc.edu (L. Itti).

0042-6989/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.visres.2004.07.042

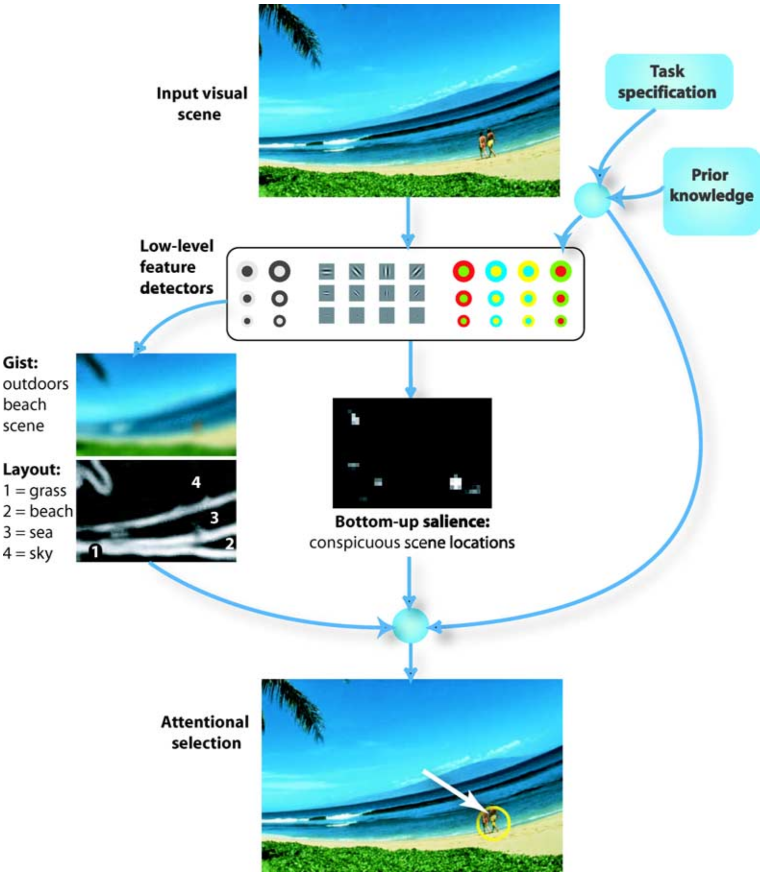


Fig. 1. Overview of current understanding of how task influences visual attention: Given a task such as “find humans in the scene”, prior knowledge of the target’s features is known to influence low-level feature extraction by priming the desired features. These low-level features are used to compute the gist and layout of the scene as well as the bottom-up salience of scene locations. Finally, the gist, layout and bottom-up salience map are somehow combined with the task and prior knowledge to guide attention to likely target locations. The present study attempts to cast this fairly vague overview model into a more precise computational framework that can be tested against real visual inputs.

may adopt a *need-based* approach (Triesch, Ballard, Hayhoe, & Sullivan, 2003), where only desired objects are quickly detected in the scene, identified and represented.

How do we determine the desired objects, and isolate them from within around 10^8 bits of information bombarding our retina each second? In this section, we provide a brief overview of some crucial factors. A detailed review of relevant literature can be found in Section 2. Studies of eye movements, physiology and psychophysics show that several factors such as bottom-up cues, knowledge of task, gist of the scene, [1] and nature of the target play important roles in selecting the focus of attention (see Fig. 1 for current understanding). Bottom-up processing guides attention based on image-based low-level cues. Such processes make a red ball more salient among a set of black balls. Gist and layout [2] guide attention to likely target locations in a top-down manner, e.g., if the task is to find humans in the scene and the gist is an outdoor beach scene, humans can be found by focusing attention near the water and the sand. Prior knowledge of the target also accelerates target detection in visual search tasks and this suggests that our visual system biases the attentional system with the known target representation so as to make the target more salient. Further, the classic eye movement experiments of Yarbus (1967) show drastically different patterns of eye movements over a same scene, depending on task. To summarize, task (with the aid of the gist and knowledge of the target) plays an important role in the selection of the focus of attention. As a consequence, eye movements vary depending on the task

[1] An abstract meaning of the scene that refers to semantic scene category, such as indoor office scene, outdoor beach scene etc.

[2] Division of the scene into regions in space based on semantic or visual similarity, e.g., a typical beach scene consists of three regions—sky on top, water in the middle, and sand at the bottom.

and humans attend to scene locations that are salient and relevant to their task.

Our goal in this paper is to model how task influences attention and to develop a better computational understanding of how different factors such as bottom-up cues, knowledge of task and target influence the guidance of attention. However, the neural implementation of several visual processes, such as computation of the gist and layout; object recognition; working of the short term memory and others is largely unknown. Rather than proposing a solution to each such open problem, we develop a working system to further our understanding of how these components may interact and interplay as a whole to fulfill task demands. To make such a large-scale integration feasible, we have focused on a few core issues, providing non-biological or black-box implementations for other components. In particular, we focus on four outstanding questions, namely determining task-relevance, biasing, recognizing, and memorizing, further introduced below.

Given a task and a visual scene, our model first determines what to look for. For this, we parse the task specification using an ontology (i.e., a knowledge base containing entities and their relationships) to yield the task-related entities and their relationships. Then, we determine the relevance of the task-related entities and simply look for the most task-relevant entity in the visual scene.

To detect a given target quickly and reliably in the scene, our model biases the low-level visual system with the known features of the target so as to make the target more salient, i.e., the bottom-up salience of the target is modulated in a top-down manner (hence, a combination of bottom-up and top-down attention). The most salient scene location is then chosen as the focus of attention. Due to biasing, the salience of the target should increase, making it more likely to draw attention.

Biasing is followed by the problem of recognition of the entity at the focus of attention. We employ a simple recognition model that shares its resources with the attention model by using the same pre-attentive features. Thus, an important aspect of our approach is to employ a common set of low-level visual primitives for bottom-up attention, object representation, top-down attention biasing, and object recognition. Further, we achieve recognition in a hierarchical manner wherein matching proceeds from a general representation of the object to a specific instance or view of the object.

Having detected and recognized the target in the scene, our model memorizes it for the purposes of scene understanding. We address an important problem in memorization and scene representation, which is the design and maintenance of an interface between symbolic knowledge of task-relevant targets and low-level visual representations based on retinotopic neural maps. For this, we propose a two-dimensional topographic map called the task-relevance map (TRM) that encodes the relevance of the scene entities. To memorize a target, the corresponding area or location in the TRM is highlighted with the target's relevance, and the target's visual features are stored in the visual working memory along with links to the symbolic knowledge of task-relevant targets. The TRM is dynamic and can be learned easily, and can be used to predict object properties such as their likely locations and sizes in a scene. To summarize, we propose, partially implement and test a computational model for the task-specific guidance of attention in visual scenes. An important aspect of the model is that its architecture is independent of the type of environment or task which it will face.

2. Motivation and related work

Visual attention has been often compared to a virtual spotlight through which our brain sees the world (Weichselgartner & Sperling, 1987). Attention has been classified into several types based on whether or not it involves eye movements (overt vs. covert attention), and whether its deployment over a scene is primarily guided by scene features or volition (bottom-up vs. top-down attention) (for review, see Itti & Koch, 2001a). The first biologically plausible architecture for controlling bottom-up attention was proposed by Koch and Ullman (1985). In their model, several feature maps (such as color, orientation, intensity) are computed in parallel across the visual field (Treisman & Gelade, 1980), and combined into a single salience map. Then, a selection process sequentially deploys attention to locations in decreasing order of their salience. We enhance this architecture by modeling the influence of task on attention.

At the early stages of visual processing, task modulates neural activity by enhancing the responses of neurons tuned to the location and features of a stimulus (Buracas, Albright, & Sejnowski, 1996; Haenny & Schiller, 1988; Moran & Desimone, 1985; Motter, 1993, 1994a, 1994b; Treue & Maunsell, 1996; Wurtz, Goldberg, & Robinson, 1980). For example, area MT+ is more active during a speed discrimination task whereas area V1 shows increased activation during a contrast discrimination task (Huk & Heeger, 2000). In addition, psychophysics experiments have shown that knowledge of the target contributes to an amplification of its salience, e.g., white vertical lines become more salient if we are looking for them (Blaser, Sperling, & Lu, 1999). A recent study even shows that better knowledge of the target leads to faster search, e.g., seeing an exact picture of the target is better than seeing a picture of the same semantic type or category as the target (Kenner & Wolfe, 2003). These studies demonstrate the effects of biasing for features of the target. Other experiments (e.g., Treisman & Gelade, 1980) have shown that searching for feature conjunctions (e.g., color × orientation conjunction search: find a red-vertical item among red-horizontal and green-vertical items) are slower than "pop-out" (e.g., find a green item among red items).[3] These observations impose constraints on the possible biasing mechanisms and eliminate the possibility of generating new composite features on the fly (as a combination of simple features).

A popular model to account for top-down feature biasing and visual search behavior is *Guided Search* (Wolfe, 1994). It has the same basic architecture as proposed by Koch and Ullman (1985), but in addition, it achieves feature-based biasing by weighing feature maps in a top-down manner. For example, with the task of detecting a red bar, the red-sensitive feature map gains more weight, hence making the red bar more salient. However, it is not clear how the weights are chosen in that model. In our model, we learn a vector of feature weights (one weight per feature) from images containing the target (see Section 5). Further, we use the same feature vectors for attentional biasing, short-term memory representation, and object recognition. Thus, our model differs from *Guided Search* in that we learn internal target representations from images, and use these learned representations for top-down biasing. Our choice for target representation is influenced by the following three factors.

First, experiments have revealed several pre-attentive features, including orientation (Julesz & Bergen, 1983; DeValois, Albrecht, & Thorell, 1982; Tootell, Silverman, Hamilton, De Valois, & Switkes, 1988; Wolfe, Priedman-Hill, Stewart, & O'Connell, 1992), size (Treisman & Gelade, 1980), closure (Enns, 1986; Triesman & Souther, 1986), color (hue) (Bauer, Jolicoeur, & Cowan, 1996; Engel, Zhang, & Wandell, 1997; Luschow & Nothdurft, 1993; Nagy & Sanchez, 1990, 1992), intensity (Beck, Prazdny, & Rosenfeld, 1983; Leventhal, 1991; Treisman & Gormican, 1988), flicker (Julesz, 1971), direction of motion (Driver, McLeod, & Dienes, 1992; Nakayama & Silverman, 1986). In our current implementation, we use orientation, color and intensity. Second, while within-feature conjunctions are considered inefficient, color × color and size × size conjunctions are efficient in a part-whole setup (e.g., find a red house with yellow windows among red houses with blue windows and blue houses with yellow windows) (Bilsky & Wolfe, 1994). Low-level visual neurons with center-surround receptive fields and color opponence can help support such observations. If we represent the target in terms of center-surround features, information about the part can be obtained from the center, and information about the whole can be obtained from the surround. Besides, using center-surround features can make the system more robust to changes in absolute feature values that are typically associated with changing viewing conditions. This motivates us to represent the target by a vector of center-surround feature weights. Third, maintaining a pyramid of feature maps at different spatial scales is known to provide a compact image code (Burt & Adelson, 1983). Hence, we are motivated to maintain feature responses at multiple spatial scales.

In summary, our current implementation uses seven center-surround feature types: on/off image intensity contrast, red/green and blue/yellow double opponent channels, and four local orientation contrast (for implementation details, please see previous papers (Itti & Koch, 2000)). We compute the feature maps at six different pairs of center and surround spatial scales (Itti & Koch, 2000), yielding 42 feature maps in all. Non-linear interactions and spatial competition occur in each of these feature maps (see Section 2.4 in Itti & Koch, 2001b) before the maps are linearly combined into a salience map. This is a very important (though often overlooked) aspect of our previously proposed bottom-up attention model, also used here in the new model. The operational definition of salience implemented in this model is such that a feature map which is active at many locations is not considered a strong driver of attention (since one would not know to which of the active locations attention should be directed), while a feature map active at only one location is a strong driver. This is implemented in the bottom-up model (Itti & Koch, 2000, 2001b) as non-classical surround inhibition within each feature map, whereby neighboring active locations cancel each other out, while a unique active location would not be affected (or even is amplified in our model). Finally, in order to find the focus of attention, we deploy a Winner-Take-All (WTA) spatial competition in the salience map that selects the most salient location in the salience map (Itti, Koch, & Niebur, 1998).

Having selected the focus of attention, it is important to recognize the entity at that scene location. Many recognition models have been proposed that can be classified based on factors including the choice of basic primitives (e.g., Gabor jets (Wiskott, Fellous, Krüger, & von der Malsburg, 1997), geometric primitives like geons (Biederman, 1987), image patches or blobs (Weber, Welling, & Perona, 2000), and view-tuned units (Riesenhuber & Poggio, 1999)), the process of matching (e.g., self-organizing dynamic link matching (Lades et al., 1993), probabilistic matching (Weber et al., 2000)), and other factors (for reviews, see Arman & Aggarwal, 1993; Riesenhuber & Poggio, 2000). In this paper, we explore how the pre-attentive features used to guide attention may be re-used for object representation and recognition. Since we represent the target as a

[3] For interpretation of colours in all figures, the reader is referred to the web version of this article.

single feature vector, we do not handle complex or composite objects in the current model.

Recognition is followed by the problem of memorization of visual information. A popular theory, the *object file theory of trans-saccadic memory* (Irwin, 1992a, 1992b; Irwin & Andrews, 1996), posits that when attention is directed to an object, the visual features and location information are bound into an object file (Kahneman & Treisman, 1984) that is maintained in visual short term memory across saccades. Psychophysics experiments have further shown that up to three or four object files may be retained in memory (Irwin, 1992a; Irwin & Zelinsky, 2002; Luck & Vogel, 1997; Pashler, 1988; Sperling, 1960). Studies investigating the neural substrates of working memory in primates and humans suggest that the frontal and extrastriate cortices may both be functionally and anatomically separated into a "what" memory for storing the visual features of the stimuli, and a "where" memory for storing spatial information (Courtney, Ungerleider, Keil, & Haxby, 1996; Wilson, O Scalaidhe, & Goldman-Rakic, 1993). Based on the above, in our model, we memorize the visual representation of the currently attended object by storing its visual features in the visual working memory. In addition, we store symbolic knowledge such as the logical properties of the currently attended object and its relationship with other objects, in the symbolic working memory with help from the symbolic long-term memory. To memorize the location of objects, we extend the earlier hypothesis of a salience map (Koch & Ullman, 1985) to propose a two-dimensional topographic *task-relevance map* that encodes the task-relevance of scene entities. Our motivation for maintaining various maps stems from biological evidence. Single-unit recordings in the visual system of the macaque indicate the existence of a number of distinct maps of the visual environment that appear to encode the salience and/or the behavioral significance of targets. Such maps have been found in the superior colliculus, the inferior and lateral subdivisions of the pulvinar, the frontal-eye fields and areas within the intraparietal sulcus (Colby & Goldberg, 1999; Gottlieb, Kusunoki, & Goldberg, 1998; Kustov & Robinson, 1996; Thompson & Schall, 2000). Since these neurons are found in different parts of the brain that specialize in different functions, we hypothesize that they may encode different types of salience: the posterior parietal cortex may encode a visual salience map, while the pre-frontal cortex may encode a top-down task-relevance map, and the final eye movements may be generated by integrating information across the visual salience map and task-relevance map to form an attention guidance map possibly stored in the superior colliculus (Fig. 2).

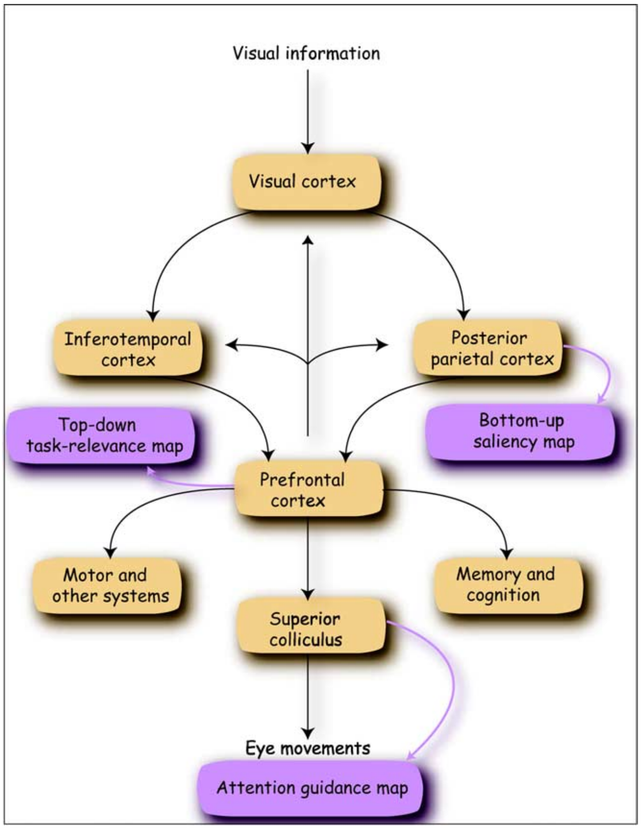


Fig. 2. We hypothesize the existence of different kinds of salience maps that encode different nature of information about the scene. In particular, we hypothesize that the posterior parietal cortex may encode a visual salience map, the pre-frontal cortex may encode a top-down task-relevance map, and the superior colliculus may store an attention guidance map that guides the focus of attention.

Our analysis so far has focused on the attentional pathway. As shown in Fig. 1, non-attentional pathways also play an important role; in particular, rapid identification of the gist (semantic category) of a scene is very useful in determining scene context, and is known to guide eye movements (Biederman, Mezzanotte, & Rabinowitz, 1982; Chun & Jiang, 1998; De Graef, Christiaens, & d'Ydewalle, 1990; Henderson & Hollingworth, 1999; Palmer, 1975; Rensink, 2000; Torralba, 2003). It is computed rapidly within the first 150 ms of scene onset (Thorpe, Fize, & Marlot, 1996), and the neural correlate of this computation is still unknown. Recently, Oliva and Torralba (2001) proposed a holistic representation of the scene based on spatial envelope properties (such as openness, naturalness etc.) that bypasses the analysis of component objects and represents the scene as a single identity. This approach formalizes the gist as a vector of contextual features (Torralba, 2003). By processing several annotated scenes, these authors learned the relationship between the scene context and categories of objects that can occur, including object properties such as locations, size or scale, and used it to focus attention on likely target locations (Torralba, 2002, 2003). This provides a good starting point for modeling the role of gist in guiding attention. Since the gist is computed rapidly, it can serve as an initial guide to attention. But subsequently, our proposed TRM that is continuously updated may serve as a better guide. For instance, in dynamic scenes such as traffic scenes where the environment is continuously changing and the targets such as cars and pedestrians are moving around, the gist may remain unchanged and hence, it may not be so useful, except as an initial guide.

The use of gist in guiding attention to likely target locations motivates knowledge-based approaches to modeling eye movements, in contrast to image-based approaches. One such famous approach is the scanpath theory which proposes that attention is mostly guided in a top-down manner based on an internal model of the scene (Norton & Stark, 1971). Computer vision models have employed a similar approach to recognize objects. For example, Rybak, Gusakova, Golovan, Podladchikova, and Shevtsova (1998) recognize objects by explicitly replaying a sequence of eye movements and matching the expected features at each fixation with the image features. In the present study, we focus on bottom-up guidance of attention and its top-down biasing, but we do not model such knowledge-based directed eye movements.

An interesting model for predicting eye movements during a search and copying task has been proposed by Rao, Zelinsky, Hayhoe, and Ballard (2002). These authors use iconic scene representations to predict eye movements during visual search. They compute salience at a given location based on the squared Euclidean distance between a feature vector containing responses of a bank of filters at that location, and the memorized vector of target responses. They validate their model against human data obtained in a search task and copying task and demonstrate some interesting center of gravity effects. This model is very interesting in that it suggests a highly efficient mechanism by which salience could be biased for the detection of a known target. However, this approach suffers from two shortcomings addressed by our model. First, since salience is computed as the distance between observed and target features, this model does not provide a mechanism by which attention could be directed in a purely bottom-up manner, when no specific target is being looked for. Hence, this model cannot reproduce simple pop-out, where a single vertical bar is immediately found by human observers within an array of horizontal bars, even in cases where observers had no prior knowledge of what to look for. Second, when target features are known, we will see in Section 7 that such template-based approach would predict that conjunction searches (Treisman & Gelade, 1980) should be as efficient as pop-out searches, which differs from empirical observations in humans. The biasing mechanism proposed in our model is less efficient but in better agreement with human data (see Section 7).

To summarize, we have motivated the components of our model which we believe are crucial for scene understanding. Ours is certainly not the first attempt to address this problem. For example, one of the finest examples of real-time scene analysis systems is *The Visual Translator* (VITRA) (Herzog & Wazinski, 1994), a computer vision system that generates real-time verbal commentaries while watching a televised soccer game. Their low-level visual system recognizes and tracks all visible objects from an overhead (bird's eye) camera view, and creates a geometric representation of the perceived scene (the 22 players, the field and the goal locations). This intermediate representation is then analyzed by series of Bayesian belief networks which evaluate spatial relations, recognize interesting motion events, and incrementally recognize plans and intentions. The model includes an abstract, non-visual notion of salience which characterizes each recognized event on the basis of recency, frequency, complexity, importance for the game, and other factors. The system finally generates a verbal commentary, which typically starts as soon as the beginning of an event has been recognized but may be interjected if highly salient events occur before the current sentence has been completed. While this system delivers very impressive results in the specific application domain considered, due to its computational complexity it is restricted to one highly structured environment and one specific task, and cannot be extended to a general scene understanding model. Indeed, unlike humans who selectively perceive the relevant objects in the scene, VITRA attends to and continuously monitors *all* objects and attempts to simultaneously recognize *all* known actions. Our approach differs from VITRA not only in that there is nothing in our model that commits it to a specific environment or task. In addition, we only memorize those objects and events that we expect to be relevant to the task at hand, thus saving enormously on computation complexity.

3. Overview of our architecture

In this section, we present a summary of our architecture which can be understood in four phases (Fig. 3).

3.1. Phase 1: eyes closed

In the first phase known as the "eyes closed" phase, the symbolic working memory (WM) is initialized by the user with a task definition in the form of keywords and their relevance (any number greater than baseline 1.0). Given the relevant keywords in symbolic WM, volitional effects such as "look at the center of the scene" could be achieved by allowing the symbolic WM to bias the TRM so that the center of the scene becomes relevant and everything else is irrelevant (but our current

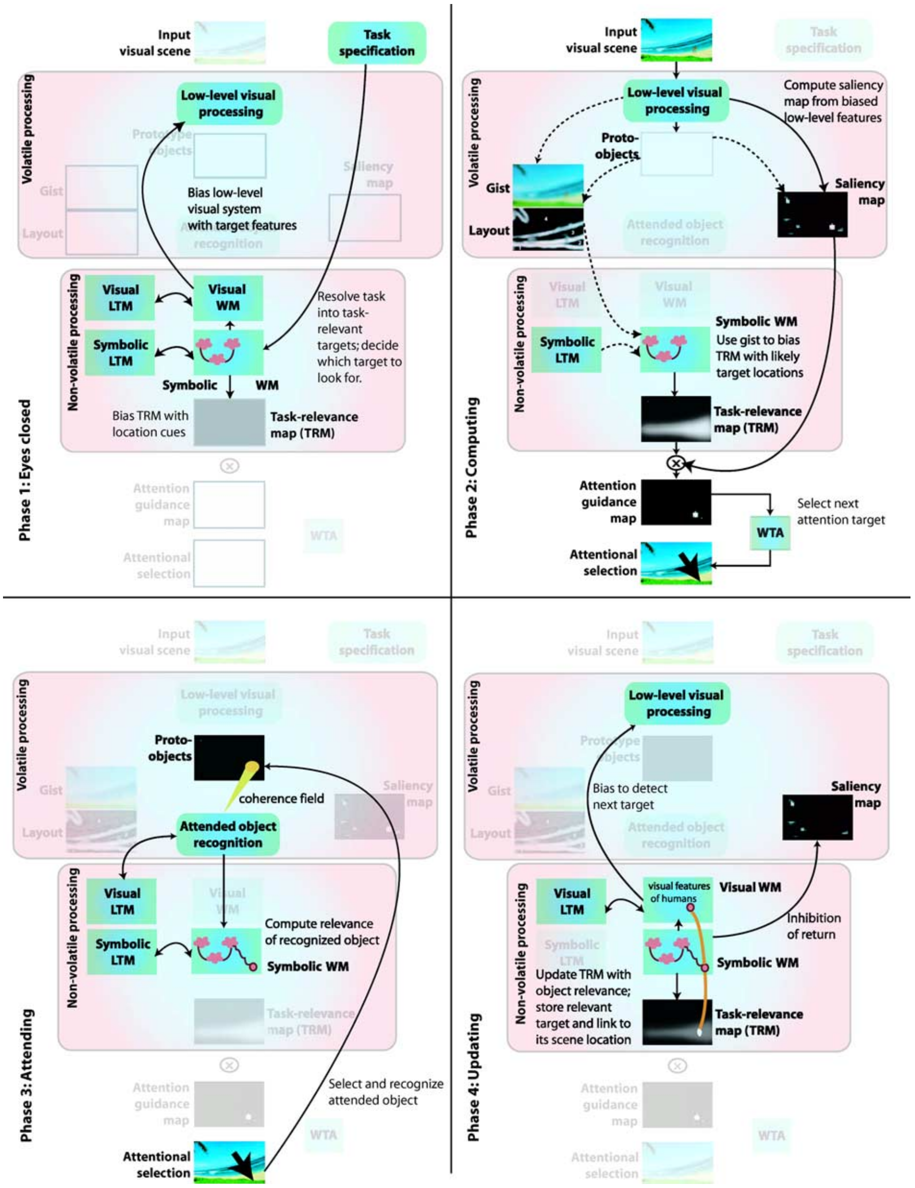


Fig. 3. Phase 1 (top left): Eyes closed, Phase 2 (top right): Computing, Phase 3 (bottom left): Attending, Phase 4 (bottom right): Updating. Please refer to Section 3 for details about each phase. All four panels represent the same model; however, to enable easy comparison of the different phases, we have highlighted the components that are active in each phase and faded those that are inactive. Dashed lines indicate parts that have not been implemented yet. Following Rensink's (2000) terminology, volatile processing stages refer to those which are under constant flux and regenerate as the input changes.

implementation has not explored this yet). For more complex tasks such as "who is doing what to whom," the symbolic WM requires prior knowledge and hence, seeks the aid of the symbolic long-term memory (LTM). For example, to find what the man in the scene is eating, prior knowledge about eating being a mouth and hand-related action, and being related to food items helps us guide attention towards mouth or hand and determine the food item. Using such prior knowledge, the symbolic WM parses the task and determines the task-relevant targets and how they are related to each other. Our implementation explores this mechanism using a simple hand-coded symbolic knowledge base to describe long-term knowledge about objects, actors and actions (Section 4). Next, it determines the current most task-relevant target as the desired target (Section 4). To detect the desired target in the scene, the visual WM retrieves the learned visual representation of the target from the visual LTM and biases the low-level visual system with the target's features (Section 5).

3.2. Phase 2: computing

In the second phase known as the "computing" phase, the eyes are open and the visual system receives the input scene. The low-level visual system that is biased by the target's features computes the biased salience map (Section 5). Apart from such feature-based attention, spatial attention may be used to focus on likely target locations, e.g., gist and layout may be used to bias the TRM to focus on relevant locations (but this is not implemented yet). Since we are interested in attending to locations that are salient and relevant, the biased salience and task-relevance maps are combined by taking a pointwise product to form the attention-guidance map (AGM). To select the focus of attention, we deploy a Winner-take-all competition that chooses the most active location in the AGM (Itti et al., 1998). It is important to note that there is no intelligence in this selection and all the intelligence of the model lies in the WM.

3.3. Phase 3: attending

In the third phase known as the "attending" phase, the low-level features or prototype objects are bound into a mid-level representation (in our implementation, this step simply extracts a vector of visual features at the attended location). The object recognition module determines the identity of the entity at the currently attended location (Section 6), and the symbolic WM estimates the task-relevance of the recognized entity (Section 4).

3.4. Phase 4: updating

In the final phase known as the "updating" phase, the WM updates its state (e.g., records that it has found the man's hand). It updates the TRM by recording the relevance of the currently attended location (Section 4). The estimated relevance may influence attention in several ways. For instance, it may affect the duration of fixation (not implemented). If the relevance of the entity is less than the baseline 1.0, it is marked as irrelevant in the TRM, and hence will be ignored by preventing future fixations on it (e.g., a chair is irrelevant when we are trying to find what the man is eating. Hence, if we see a chair, we ignore it). If it is somewhat relevant (e.g., man's eyes), it may be used to guide attention to a more relevant target by means of directed attention shifts (e.g., look down to find the man's mouth or hand; not implemented). Also if it is relevant (e.g., man's hand), a detailed representation of the scene entity may be created for further scrutiny (e.g., a spatio-temporal structure for tracking the hand; not implemented). The WM also inhibits the current focus of attention from continuously demanding attention (inhibition of return in SM). Then, the symbolic WM determines the next most task-relevant target, and the visual WM retrieves the target's learned visual representation from visual LTM, and uses it to bias the low-level visual system.

This completes one iteration. The computing, attending and updating phases repeat until the task is complete. Upon completion, the TRM shows all task-relevant locations and the symbolic WM contains all task-relevant targets.

As mentioned earlier (Section 1), our focus in this paper is on determining task-relevance, biasing, recognizing, and memorizing. Accordingly, we have designed symbolic LTM and WM modules for estimating task-relevance (Sections 4.1 and 4.2) and also for computing and learning task-relevant locations in a TRM (Sections 4.2 and 7); visual WM and LTM modules for learning object representations (Section 5.1), reusing the learned target representations to compute the biased saliency map for object detection (see Section 5.2), and matching against learned representations for object recognition (see Section 6). Implementation of other components (such as gist, layout, object trackers) and their interactions is still under progress and we do not include their details in this paper.

4. Estimating the task-relevance of scene entities

In this section, we propose a computational framework for estimating the task-relevance of scene locations. This is essentially a top-down process requiring prior knowledge about the world and some semantic processing. Hence, we recruit symbolic LTM and WM modules. Our current architecture is based on research in artificial intelligence and knowledge representation (Brachman & Levesque, 1985) and is not biological.

4.1. Symbolic long-term memory (LTM)

The symbolic LTM acts as a knowledge base. It contains entities and their relationships. For consistency with the vocabulary used in knowledge representation research, we refer to it as ontology from now on. We currently address tasks such as "who is doing what to whom" and accept task specifications in the form of object, subject and action keywords. Hence, we maintain object, subject and action ontologies. Each ontology is represented as a graph with entities as vertices and their relationships as edges. Our entities include real-world concepts as well as abstract ones. In our current implementation, we consider simple relationships such as *is a*, *includes*, *part of*, *contains*, *similar*, and *related*. The following examples motivate the need to store more information in the edges. Consider the case when we want to find a hand. Suppose we find a finger (hand contains finger) and a man (hand is part of man), how should we determine which of them is more relevant? Clearly, the finger is more relevant than the man because if the finger is found, it implies that the hand has been found. However, if the man is found, we still require a few eye movements before finding the hand within the man. To incorporate this, we create a partial order on the set of relationships by ranking them according to the priority or granularity of a relationship $g(r(u,v))$, where $r(u,v)$ is the relationship between entity (u and v). In general,

- $g(\mathit{contains}) > g(\mathit{part\ of})$,
- $g(\mathit{is\ a}) > g(\mathit{includes})$,
- $g(\mathit{related}) > g(\mathit{similar})$.

Let us consider another case where we still want to find the hand, but we find a pen and a leaf instead, and wish to estimate their relevance. This situation is unlike the previous one since both entities are hand-related objects and hence, share the same relationship with the hand. Yet, we consider the pen to be more relevant than the leaf because in our daily lives, the hand holds a pen more often than it holds a leaf (unless we are considering gardeners!). Thus, the probability of joint occurrence of entities seems to be an important factor in determining relevance. Hence, we store co-occurrence of the entities $c(u,v)$.

Apart from storing information in the edges, we also store information in the nodes. Each node maintains a list of properties in addition to the list of all its neighbors. To represent conjunctions and disjunctions or other complicated relationships, we maintain truth tables that describe the probabilities of various combinations of parent entities. An example is shown in Fig. 4. Currently, our ontology is not learnable. For the purposes of testing the model, we have hand-coded the ontology with hand-picked values of co-occurrence and granularity.

4.2. Symbolic working memory (WM)

The symbolic WM creates and maintains task graphs for objects, subjects and actions that contain task-relevant entities and their relationships. After the entity at the current fixation (fixated entity) is recognized, symbolic WM estimates its task-relevance as follows. First, it checks whether the fixated entity is already present in the task graph, in which case, a simple lookup gives the relevance of the fixated entity. If it fails to find the fixated entity in its task graph, then it seeks the help of symbolic LTM in the following: the symbolic WM requests the symbolic LTM to check whether there exists a path in the ontology from the fixated entity to any of the entities in the task graph. If so, the nature of the path reveals how the fixation is related to the current task graph. If no such path exists, the fixated entity is declared to be irrelevant to the task. In the case of the object task graph, an extra check is performed to ensure that the properties of the fixated entity are consistent with the object task graph (see Fig. 5 for examples). If the tests succeed and the fixated entity is determined to be relevant, the symbolic LTM returns the discovered paths (from the fixated entity to the entities in the task graph) to the symbolic WM.

The symbolic WM computes the relevance of the fixated entity as a function of the relevance of its neighboring entities (in the task graph) and the nature of its connecting relations. Let us consider the influence of entity u (whose relevance is known) on entity v (whose relevance is to be computed). This depends on

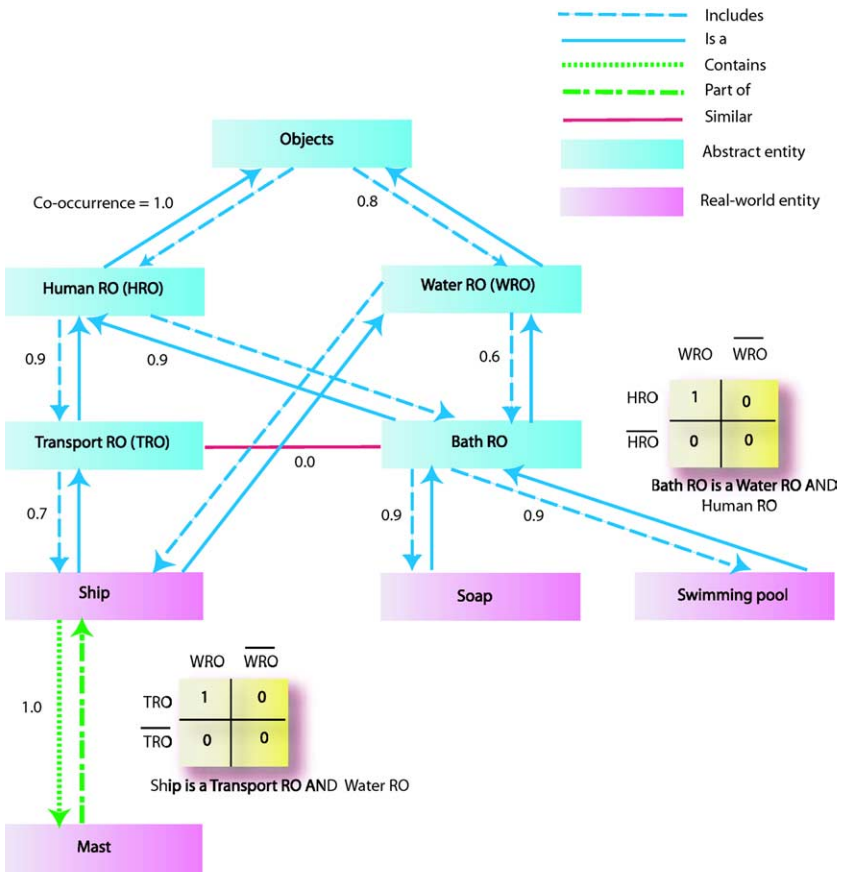
- the relevance of entity u (R_u),
- the granularity of the relationship $r(u,v)$ ($g(r(u,v))$).
- the conditional probability $P(v\ \mathit{is\ relevant}|u\ \mathit{is\ relevant})$. For the purposes of visual scene analysis, v is considered to be relevant if it helps us find u. Hence the conditional probability can be estimated from previous experience as $P(u\ \mathit{will\ be\ found}|v\ \mathit{is\ found})$ or $P(u\ \mathit{occurs}|v\ \mathit{occurs})$. This is the same as $c(u,v)/P(v)$, where $c(u,v)$ is the co-occurrence of u and v.

To model the decaying influence with increasing path length [4] between the entities, we introduce a *decay_factor* that lies between 0 and 1. Thus we arrive at the following expression for computing relevance of entity v (R_v):

$$R_v = \max_{u:(u,v)\ \mathit{is\ an\ edge}} (R_u * g(r(u,v)) * c(u,v)/P(v) * \mathrm{decay\_factor}) \qquad (1)$$

The relevance of a new entity depends on the task-relevant entities already present in the task graph. Hence, creation of the initial task graph is important. In our implementation, the initial task graph consists of task keywords and their relevance as input by the user. For instance, given a task specification such as "what is the man catching", the user inputs "man" as the subject keyword and "catch" as the action keyword, along with their relevance (any number greater than baseline 1.0). After adding these keywords to the task graph, we further expand the task graph through the *is a* relation. Our new task graph contains "man *is a* human", "catch *is a* hand-related action". As a general rule, upon addition of a new entity into the task graph, we expand it to the *related* entities (entities connected through the *related* relation). In this example, we expand the initial task graph to "hand-related action is *related* to hand and hand-related object". Thus even before the first fixation, we know that we are looking for a hand-related object, i.e., we have an idea about what entities are expected to be relevant. Such expansion of the task into task-relevant targets allows the model to compute the relevance of fixated entities in the manner explained above. For example, if the fixation is

[4] Path length between two nodes A and B of a graph is calculated as the number of edges in the path between A and B.



	WRO	$\overline{\text{WRO}}$
HRO	1	0
$\overline{\text{HRO}}$	0	0

Bath RO is a Water RO AND Human RO

	WRO	$\overline{\text{WRO}}$
TRO	1	0
$\overline{\text{TRO}}$	0	0

Ship is a Transport RO AND Water RO

Fig. 4. Sample ontology, as used to represent long-term knowledge in our model. The relations include is *a*, *includes*, *part of*, *contains*, *similar*, *related*. While the first five relations appear as edges within a given ontology, the *related* relation appears as edges that connect the three different ontologies. The relations *contains* and *part of* are complementary to each other as in *Ship contains Mast, Mast is part of Ship*. Similarly, *is a* and *includes* are complementary. Hand-picked co-occurrence measures are shown on each edge and the conjunctions, disjunctions are shown using the truth tables. In the figure, RO refers to related object.

recognized as an object belonging to the car category, then it is determined to be irrelevant as it is not a hand-related object (Fig. 5).

To summarize, our proposed architecture expands a given task into task-relevant entities and determines the task-relevance of scene entities. Once the task-relevant entities or targets are known, the next step is to efficiently detect them in the scene.

5. Top-down biasing for object detection

With just the elementary information available at the pre-attentive stage in the form of low-level feature maps tuned to color, intensity and orientation, our model learns representations of objects in diverse, complex backgrounds. The representation starts with simple vectors of low-level feature values computed at different locations on the object, called views. We then recursively combine these views to form instances, in turn combined into simple objects, composite objects, and so on, taking into account feature values and their variance. Given any new scene, our model uses the learned representation of the target object to perform top-down biasing on the attentional system, such as to render this object more salient by enhancing those features that are characteristic of the object. The details of how our model learns and detects targets are explained in the following subsections.

5.1. Learning the object representation

During the learning phase, the model operates in a free-viewing mode. That is, in the absence of any task, there are no top-down effects, the TRM is uniform (baseline 1.0 everywhere), and the AGM is the same as the salience map. Thus, in the absence of task, our model deploys attention according to the bottom-up salience model (Itti & Koch, 2000). To guide the model to the location of the target, we use a binary target mask that serves as a location cue by highlighting the targets in the input image. It should be noted that we do not use the target mask to segment the target from its background. In fact, we attempt to learn not only the object

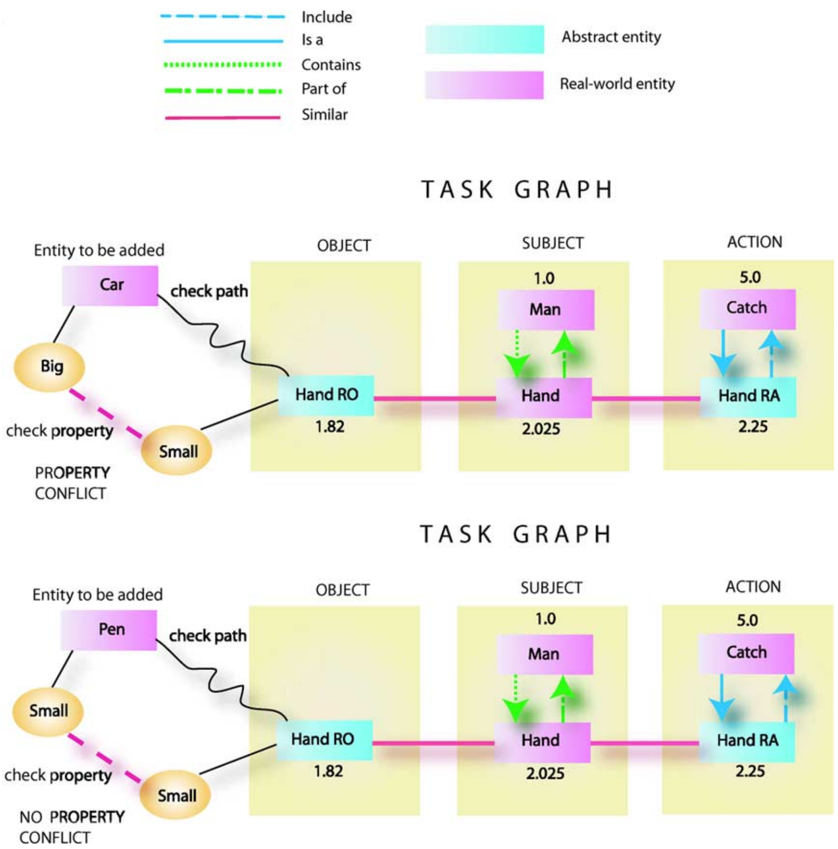


Fig. 5. To estimate the relevance of an entity, we check the existence of a path from the entity to the task graph and check for property conflicts. To find "what is the man catching", we are looking for a hand related object that is small and holdable, hence a big object like car is considered irrelevant; whereas a small object like pen is considered relevant.

properties, but also local neighborhood properties. This is useful since in several cases, the object and its background may co-occur and hence, the background information may aid in the detection of the object.

When the model attends to the target, a few locations are chosen around that salient location (currently, the model chooses nine locations from a 3×3 grid of fixed size centered at the salient location). For each chosen location, the visual WM learns the center-surround features at multiple spatial scales and stores them in the visual LTM. The coarser scales include information about the background while the finer scales contain information about the target. Specifically, a 42-component feature vector extracted at a given location represents a view (red/green, blue/yellow, intensity and four orientations at six center-surround scales). Thus, we obtain a collection of views contained in the current instance of the target.

The visual WM combines the different views obtained above to form a more stable, general representation of an instance of the object that is robust to noise. It repeats this process by retrieving the stored instances from the visual LTM and combining them to form a general representation of the object and so on. The following rules are used for combination of several object classes (equally likely, mutually exclusive) to form a general representation of the super-object class. Let X_i be the event that the ith object class occurs, where $i \in 1, 2, \ldots, n$. Let Y be the event that the super-object class occurs. We define Y as follows:

$$Y = \bigcup_i X_i \tag{2}$$

In other words, an observation is said to belong to the super-object class if and only if it belongs to any of the object classes (e.g., an observation belongs to an object category if and only if it belongs to any of the object instances).

Let O be the random variable denoting an observation and $O = o$ be the event that the value o is observed. $P(O = o|X_i)$ refers to the class conditional density, i.e., the probability of observing $O = o$ given that the ith object class has occurred. Let $P(O = o|X_i)$ follow a normal distribution $N(\mu_i, \Sigma_i)$ where $\mu_i = (\mu_{i1}\, \mu_{i2} \cdots \mu_{i42})^{\mathrm{T}}$, i.e., a vector of the mean feature values, and Σ_i is the covariance matrix. Due to our assumption that the different features are independent, the covariance matrix reduces to a diagonal matrix, whose diagonal entries equal the variance in feature values, represented as $\sigma_i^2 = (\sigma_{i1}^2\, \sigma_{i2}^2 \cdots \sigma_{i42}^2)^{\mathrm{T}}$.

Our aim is to find the distribution of $O|Y$. As shown in Appendix A, we obtain the following:

$$P(O=o \mid Y) = \sum_i P(O=o \mid X_i) w_i \quad (3)$$

$$\text{where } w_i = P(X_i) \Big/ \sum_j P(X_j) \quad (4)$$

$$= 1/n \quad (\text{since } X_i \text{ are equally likely}) \quad (5)$$

$$\mu = E[O \mid Y] \quad (6)$$

$$= \sum_i w_i \mu_i \quad (7)$$

$$\sigma^2 = E[(O \mid Y)^2] - (E[O \mid Y])^2 \quad (8)$$

$$= \sum_i w_i(\sigma_i^2 + \mu_i^2) - \mu^2 \quad (9)$$

In general, $O|Y$ has a multi-modal distribution. But as a first approximation and to achieve recursion in our implementation, we consider only up to the second moment and approximate this multi-modal distribution by a normal distribution $N(\mu, \sigma^2)$.

By processing several images containing different poses and sizes of an object, the visual WM, along with the help of visual LTM, learns the representation of the views, instances and combines them to form a representation of the object (Fig. 6).

5.2. Object detection using the learned visual representation

To detect a specific target object in any scene, the visual WM uses the learned representation stored in the visual LTM to bias the combination of different feature maps to form the salience map. A feature f is considered to be relevant and reliable if its mean feature value is high and its feature variance is low. Hence, we determine the weight by which this feature will contribute to the salience map (feature weight) as $R(f)$.

$$R(f) = \textit{relevance of feature } f = \frac{\mu(f)}{1+\sigma(f)}$$

where

$\mu(f) =$ mean response to feature f,

$\sigma^2(f) =$ variance in response to feature f

We compute several classes of features in several visual processing channels (Section 2) and create a channel hierarchy H as follows. $H(0)$ (leaves): the set of all features at different spatial scales; $H(1)$: the set of subchannels formed by combining features of different spatial scales and the same feature type; $H(2)$: the set of channels formed by combining subchannels of same modality; … $H(n)$: the salience map (where n is the height of H). In order to promote the target in all the feature channels in the channel hierarchy, each parent channel promotes itself proportionally to the maximum feature weight of its children channels.

$$\forall p \in \bigcup_{k=0}^{n} H(k), \quad R(p) \propto \max_{c \in \text{children}(p)} (R(c))$$

For instance, if the target has a strong horizontal edge at some scale, then the weight of the 0° subchannel increases and so does the weight of the orientation channel. Hence, those channels that are irrelevant for this target are weighted down and contribute little to the salience map (e.g., for detecting a horizontal object, color is irrelevant and hence the color channel's weights are decreased). At each level of the channel hierarchy, weighted maps of the children channels (Map_c) are summed into a unique map at the parent channel (Map_p), resulting in the salience map at the root of the hierarchy.

$$\forall p \in \bigcup_{k=0}^{n} H(k), \quad \text{Map}_\text{p}(x,y) = f\left(\sum_{c \in \text{children}(p)} R(c) * \text{Map}_\text{c}(x,y)\right)$$

where f refers to the spatial competition. For details regarding its implementation, please see Section 2.4 in Itti and Koch (2001b); as mentioned earlier, its role is to prune those feature maps where many locations are strongly active (and hence none may be considered a stronger attractor of attention than any other), while promoting maps where a single or a few locations are active (and tend to pop-out). This aspect of the saliency model (Itti & Koch, 2001b; Itti et al., 1998) is also further discussed in Section 7 and Figs. 11 and 12. In the salience map thus formed by biasing the combination of all feature maps, all scene locations whose local features are similar to the target's relevant features become more salient and likely to draw attention (Fig. 7). The false positives at this stage can be removed at the recognition stage.

6. Using attention for object recognition

Our current implementation for object recognition is aimed at re-using pre-attentive features used to guide attention. Hence, we adopt the simplest approach and treat the object as a feature vector, with no explicit representation of structure. While this imposes limitations on the complexity of objects that our model can recognize, it is fast and may serve to prune the search space, thus acting as a filter that may feed into more complex, slower recognition systems.

To recognize an object, our model attends to any location in the object, and extracts the center-surround feature vector from that location. We try to recognize the entity at the current fixation by matching the extracted feature vector ($\mathbf{f}$) with those already learned ($\mathbf{O} = \{\mathbf{o}_1, \mathbf{o}_2 \ldots \mathbf{o}_n\}$) and stored in the visual LTM (see Fig. 8). We use a maximum likelihood estimation technique to find the match between $\mathbf{f}$ and $\mathbf{O}$, i.e., find the object $\mathbf{o}_i$ that maximizes $P(\mathbf{f}|\mathbf{o}_i)$. Let $\text{Match}(\mathbf{f}, k)$ denote the

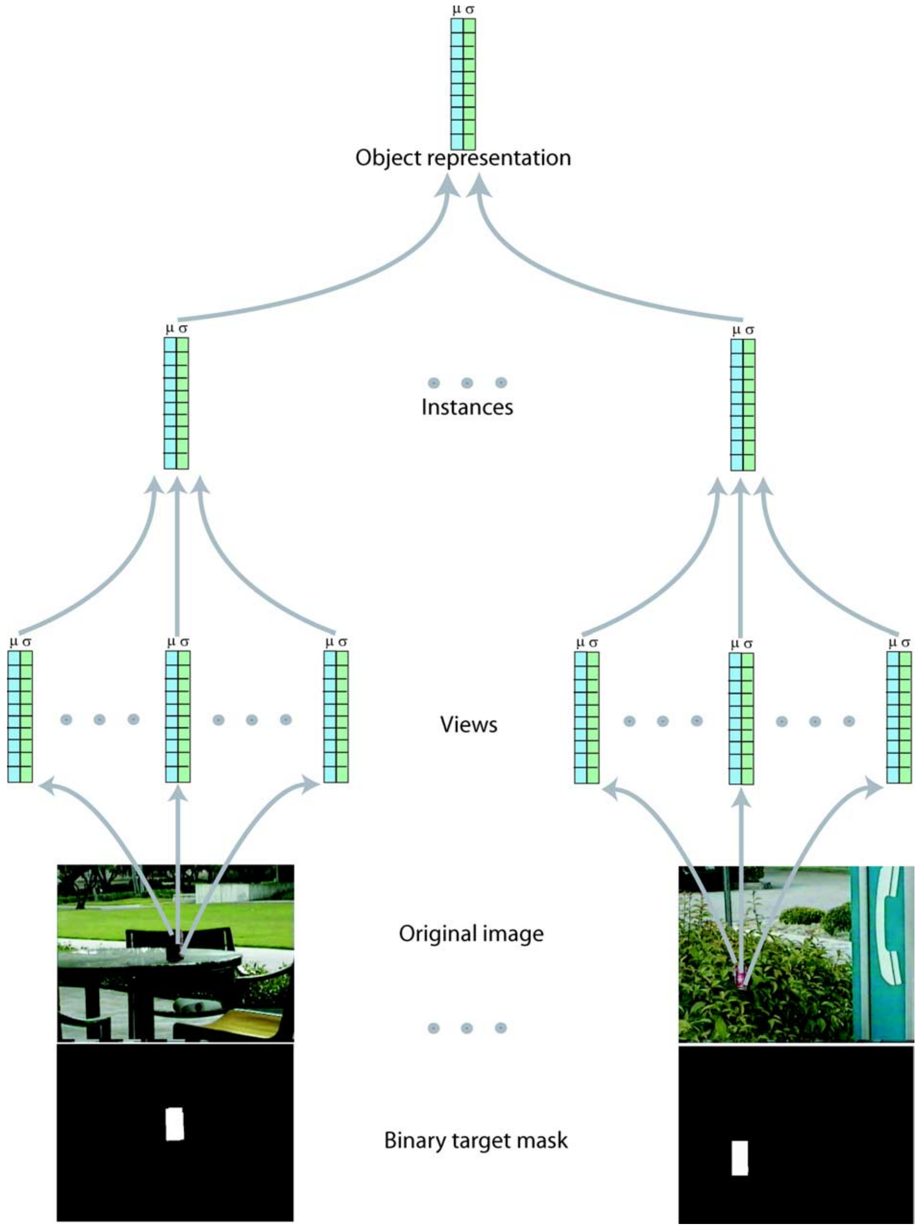


Fig. 6. Learning a general representation of an object. The model uses a binary target mask (target is 1 and background is 0) to serve as a location cue. The model learns the views by extracting the center-surround feature vectors at different spatial scales from a few locations within the target. Next, it combines the views to form instances. The instances are in turn combined to form a general representation of the object.

set of nodes that provide a good match among all nodes from the root (level 0) to some desired level k of specificity. We compute it progressively in increasing levels of specificity by first finding Match($\mathbf{f}$, 0), then finding Match($\mathbf{f}$, 1), and so on up to Match($\mathbf{f}$, k), i.e., by first comparing against general object representations and then comparing against more specific representations such as a particular object or instance or view. At each level, we narrow our search space and improve the speed of recognition by pruning those subtrees rooted at nodes that do not provide a good match, and selectively expanding those nodes that provide a good match. We find a good match among a set of nodes by comparing the likelihood estimates of the nodes to find a unique maximum which is twice higher than the second maximum. If we find a unique maximum, the corresponding node provides a good match. Else in the presence of ambiguity, all nodes whose likelihood estimates are greater than or equal to the mean likelihood estimate are considered to provide a good match. Given Match($\mathbf{f}$, x), we find Match($\mathbf{f}$, $x + 1$) as follows:

6.1. Case 1: $|Match(\boldsymbol{f}, x)| = 1$: Unique match at level x

If level x is the deepest level in the object hierarchy, then we have successfully found the most specific

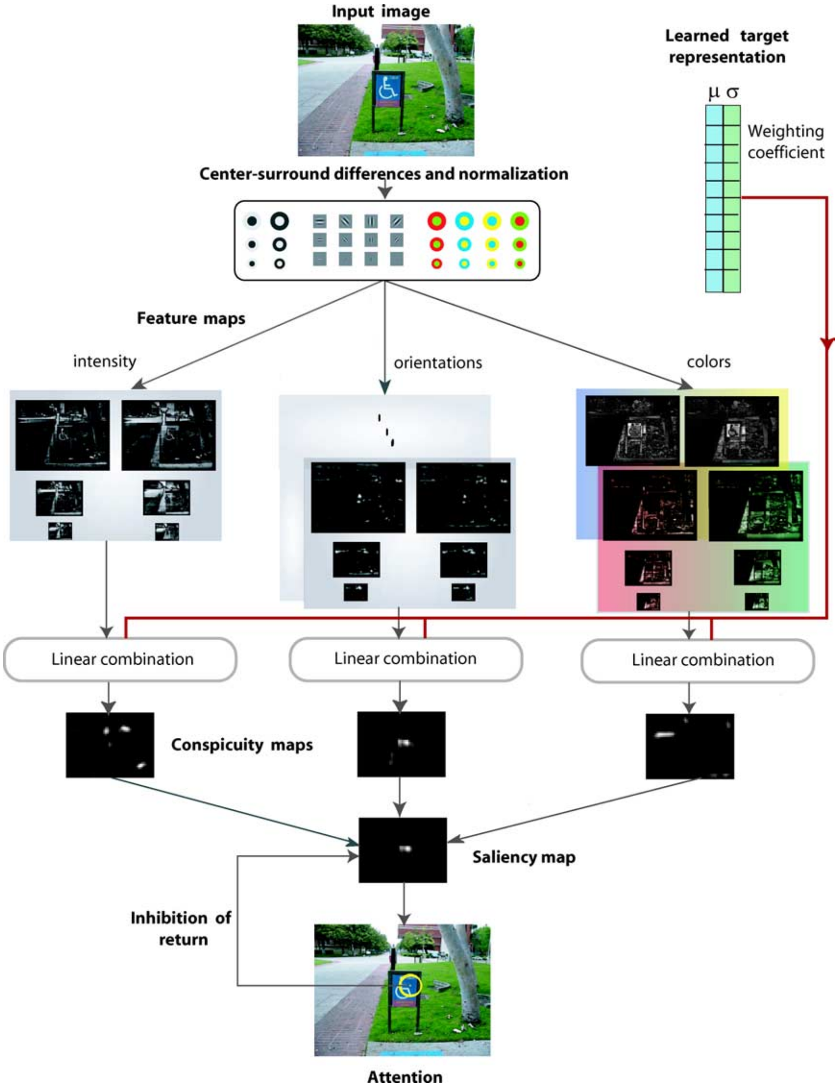


Fig. 7. Top-down biasing model for object detection. To detect a specific target object in any scene, we use the learned target representation to bias the linear combination of different feature maps to form the salience map. In the salience map thus formed, all scene locations whose features are similar to the target become more salient and are more likely to draw attention.

representation that matches the fixated entity. We output Match(**f**, x) and terminate our search. Else, given that the general object representation at level x provides a good match, we proceed deeper into the object hierarchy to find a better match among more specific representations. We accomplish this by expanding the matching node at level x into its children nodes at level $x + 1$. If the parent node provides a better match than the children nodes (e.g., a gray stimulus may match the gray parent better than its white or black children), we prune the subtree rooted at the parent node and Match(**f**, $x + 1$) = Match(**f**, x). Else, Match(**f**, $x + 1$) equals the set of children nodes that provide a good match.

6.2. Case 2: $|Match(f, x)| > 1$: Ambiguity at level x

If level x is the deepest level in the object hierarchy, then we declare ambiguity in recognition and output the node that provides the best match among Match(**f**, x). Else, we resolve the ambiguity at this level by seeking better matches at the next level $x + 1$. We expand each matching node at level x into its children nodes at level $x + 1$, taking care to prune the subtree if the parent node matches better than its children. Among the nodes thus obtained, Match(**f**, $x + 1$) equals the set of nodes that provide a good match.

Although simple and limited, this object recognition scheme has proven sufficiently robust to allow us to test the model with complex natural scenes, as described in the following section.

7. Results

As a first test of the model, we consider a search task for a known target and wish to detect it as fast as possible.

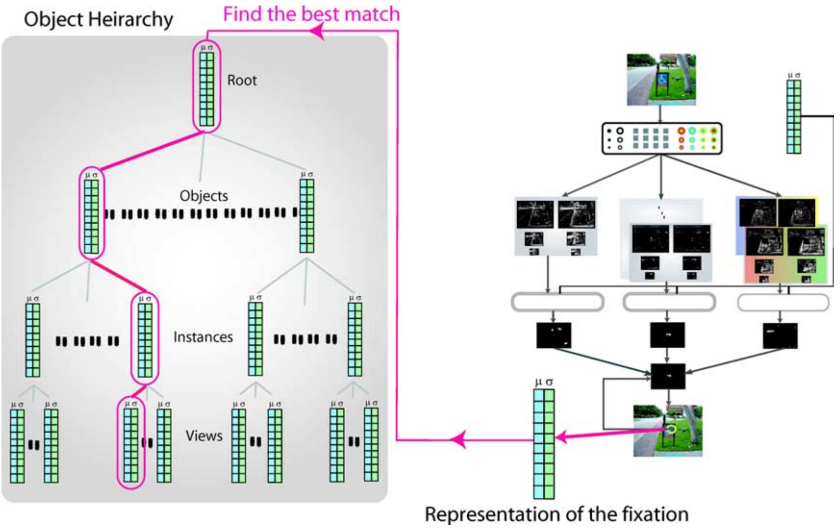


Fig. 8. Architecture for object recognition. Our model recognizes the object at the attended scene location by extracting a center-surround feature vector from that location and finding the best match by comparing it against representations stored in the object hierarchy.

This test aims at evaluating our model's efficiency against a naive bottom-up model by comparing the speed of detection and the salience of the target. We allowed our model to learn the visual features of the target from training images (12 training images per target object on average, 24 target objects) and their corresponding target masks (the target mask highlighted the target and served as a location cue for training only). To detect the target in a new scene, the visual WM biased the bottom-up attentional system to enhance the salience of scene locations that were similar to the target. Attention was guided to locations whose biased salience was high. We tested the model on 343 new scenes and measured the improvement in performance of our top-down biased model over the naive, bottom-up model (Itti & Koch, 2000). There was a significant improvement in detection and in the salience of the target in many but not all cases, verified as follows by statistical testing for a significance level of 0.05 (Fig. 9). The null hypothesis H_0 (mean improvement of 2.00 in target salience normalized by maximum salience in the image) was compared to alternate hypotheses H_1 (mean improvement in normalized target salience <2.00) and H_2 (mean improvement in normalized target salience >2.00). In some scenes, the distractors were similar to the target, making the search tasks difficult (e.g., detect a circle among ellipses). In such cases, biasing for the target led to an increase in salience of the target as well as the distractors that shared the target's features. Due to the spatial competition that followed, the salience of the target was modulated and there was no significant improvement in detection time or salience of the target, hence supporting the alternative hypothesis H_1. A particularly interesting case occurred when we tried to detect a circle among circles with vertical bars. The bottom-up salience of the circle was very low and biasing improved its salience by a large factor. But biasing also boosted the salience of all the circles with vertical bars and due to the spatial competition, the biased salience of the target became low and hence it did not pop-out (just like this search is always difficult for humans, whether or not they know the target (Treisman & Gormican, 1988)). But in the opposite case where we tried to detect a circle with a vertical bar among circles, biasing did not affect the performance since the target was already the most bottom-up salient item and popped out. In most scenes, despite interference from the distractors, biasing improved target salience and detection time (data supported H_0 or H_2). For example, biasing accelerated the detection of a square among rectangles 15.56-fold on average. An example of a comparison between the number of fixations taken by the biased vs. unbiased models is shown in Fig. 10.

This first set of results suggested that the spatially global (Saenz, Buracas, & Boynton, 2002) (one weight per feature map) biasing mechanism implemented here and similar in spirit to Guided Search (Wolfe, 1994) may or may not improve search performance, depending on the presence of shared features between target and distractors. To further explore the validity of such a mechanism, we compared our biased model's predictions with existing psychophysical data and other models such as a random model, the bottom-up or unbiased model (Itti & Koch, 2000), and the top-down search model proposed by Rao et al. (2002). As mentioned in Section 2, Rao et al.'s model assumes a much stronger biasing mechanism, whereby salience at every location reflects similarity between the local low-level features and the target features provided top-down (with

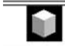
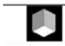
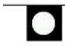
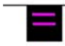
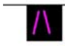
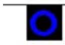
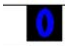
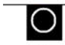
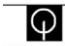
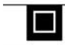
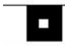
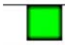
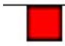
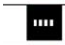
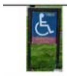
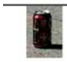
Target	Distractor	95% $conf_{salience}$	95% $conf_{time}$	95% $conf_{shifts}$	Hypothesis	Remarks
		[0.78, 2.20]	[0.00, 14.87]	[0.86, 1.06]	H_0	Biasing improves detection time
		[1.21, 3.84]	[0.72, 1.25]	[0.70, 1.09]	H_0	Biasing does not affect detection time
		[0.13, 0.32]	[0.10, 0.20]	[0.08, 0.16]	H_1	Biasing increases detection time
		[1.11, 1.40]	[1.02, 1.04]	[1.00, 1.00]	H_1	Pop-out
		[1.06, 2.30]	[1.76, 3.06]	[1.91, 3.88]	H_0	Biasing improves detection time
		[0.26, 1.00]	[0.09, 0.21]	[0.18, 0.36]	H_1	Biasing increases detection time
		[0.90, 1.01]	[1.00, 1.00]	[1.00, 1.00]	H_1	Pop-out
		[17.73, 964.89]	[0.37, 1.11]	[0.73, 1.09]	H_2	Biasing does not affect detection time
		[1.00, 1.11]	[1.00, 1.00]	[1.00, 1.00]	H_1	Pop-out
		[0.00, 3319.3]	[7.85, 23.26]	[11.61, 19.18]	H_0	Biasing improves detection time
		[2.09, 8.72]	[4.47, 10.46]	[5.69, 13.30]	H_2	Biasing improves detection time
		[1.02, 1.19]	[1.03, 1.55]	[1.00, 1.00]	H_1	Biasing improves detection time
		[0.97, 1.18]	[1.00, 1.46]	[1.03, 1.77]	H_1	Biasing improves detection time
		[3032.20, 7060.60]	[16.02, 17.85]	[20.00, 20.00]	H_2	Biasing improves detection time
	natural	[2.48, 23.79]	[0.49, 1.06]	[0.53, 1.15]	H_2	Biasing does not affect detection time
	natural	[0.00, 15.13]	[0.47, 1.37]	[0.49, 1.53]	H_0	Biasing does not affect detection time
	natural	[1.00, 4.39]	[1.07, 2.17]	[1.09, 2.66]	H_0	Biasing improves detection time
	natural	[1.88, 2.77]	[1.74, 2.59]	[1.79, 2.39]	H_0	Biasing improves detection time

Fig. 9. Our model's results for top-down biasing results for a sample from our database of objects. The first column is the target object that we biased the model for; the second column shows the distractor object when in a search array setup, or "natural" means that a natural cluttered scene was the background or distractor; the third column shows the 95% confidence interval for improvement in target salience normalized by maximum salience in the display (biased over naive models); the fourth column shows the 95% confidence interval for improvement in detection time (naive over biased models); the fifth column shows the 95% confidence interval for improvement in number of attentional shifts before detection of the target (naive over biased models); the sixth column shows the hypothesis supported by the salience data. The null hypothesis H_0 (mean improvement in normalized target salience = 2.0) or alternative hypothesis H_2 (mean improvement in normalized target salience >2.0) was supported by a majority of the target objects. In some cases where the distractors were very similar to the target, the alternative hypothesis H_1 (mean improvement in normalized target salience <2.0) was supported. The final column shows some remarks on the effect of biasing on detection time. Note that in the case of pop-out, improvement in normalized target salience is approximately 1.0 because the target is already the most salient item in the display (hence, target salience normalized by maximum salience equals 1.0), and biasing maintains the target as the most salient item.

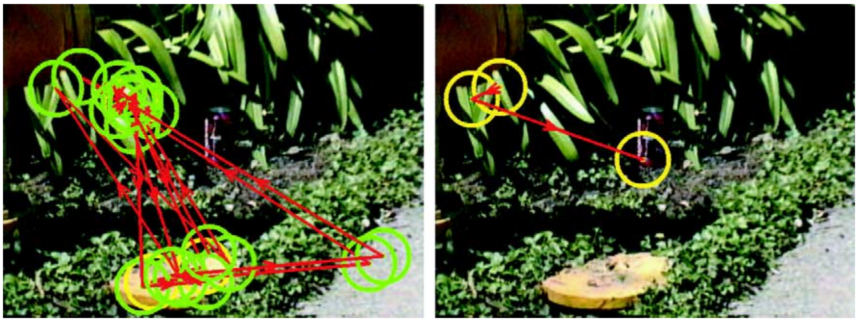


Fig. 10. The example on the left shows the attentional trajectory during free examination of this scene by the naive, bottom-up salience model (yellow circles represent highly salient locations, green circles represent less salient locations, red arrows show the scanpath). Even after 20 fixations, the model did not attend to the coke can, simply because its salience was very low compared to that of other conspicuous objects in the scene. Displayed on the right is the attentional trajectory after top-down biasing for the coke can object class (built from instances and views of the coke can from other photographs containing the can in various settings). Our model detected the target as early as the third fixation.

similarity is based on the Euclidean distance between feature vectors).

To develop an intuitive understanding of the comparison between both models, consider a conjunction search array with red and blue vertical and horizontal elements (and a single red-vertical target) like in Fig. 11. In our model, biasing for the features of the target means giving a high weight to red color (the red/green feature maps) and to vertical orientation (the vertical feature maps). Because each of these feature maps contains many active locations (the target, but also half of the distractors), the spatial competition in each feature map (Itti & Koch, 2001b) is expected to drive those maps to zero, no matter how strongly biased they may be (remember that the spatial competition tends to promote maps which contain a unique active location and to demote those which contain many active locations). In the end, biasing is rather ineffective because it increased the weights of feature maps that were basically noise and not attractors of attention. It is not totally ineffective, though, because the target is amplified twice (once in the red/green maps and once in the vertical maps) and hence exhibits slightly increased salience, though still very low. In contrast, a template matching algorithm like that of Rao et al. would predict that biasing for the target should render it salient, since the target will exhibit a feature distance near zero (perfect match between local features and top-down biasing features, corresponding to highest salience), while distractors will exhibit non-zero distances (mismatch in at least one feature value). Whether the difference between target and distractor salience values is sufficient to yield pop-out can be controlled in Rao et al.'s model by a softmax parameter, λ, which determines how dominantly the location of maximum salience attracts attention compared to locations of lesser salience. To decide on a fair value for λ, we chose the one which barely allowed our re-implementation of Rao et al.'s model to find the target in constant time on simple pop-out search arrays (red-vertical bar among red-horizontal distractors, and red-vertical bar among blue-vertical distractors). To further allow a fair comparison, our re-implementation of Rao et al.'s model used the same set of features and center-surround scales as our model.

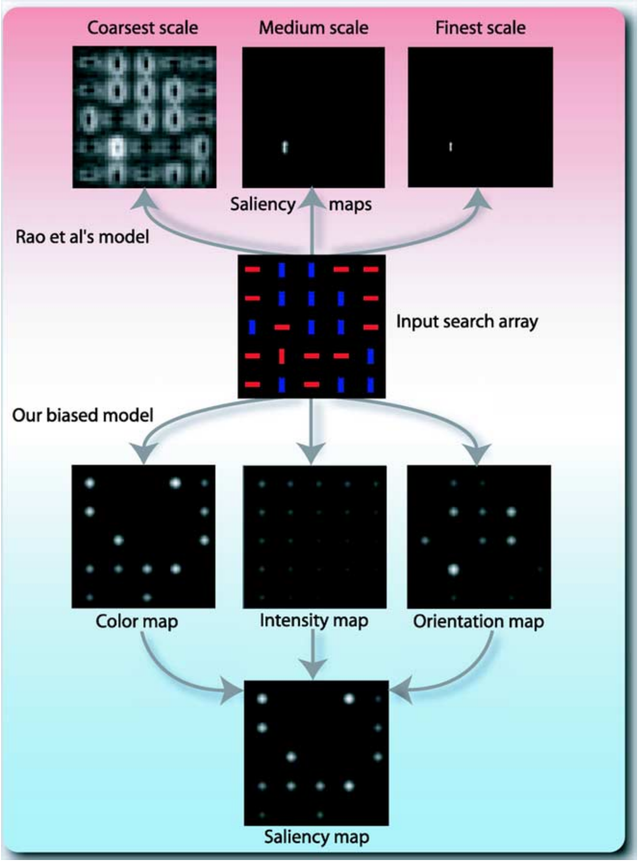


Fig. 11. Difference between our biased model and Rao et al.'s model. Consider searching for a red-vertical item among red-horizontal and blue-vertical items. Rao's model computes salience of each scene location based on the Euclidean distance between the target and that location in feature space, by progressively considering the information at coarse-to-fine scales. The corresponding salience maps obtained for the first three fixations are shown here. As early as the third fixation, the salience map including the finest scale clearly shows the target to be the single most salient location in the scene. Thus, Rao's model predicts that conjunction searches are efficient (see Section 7 for details on our re-implementation of that model). On the other hand, in our model, biasing promotes the red and vertical features. In the resulting color feature map, the target as well as red-horizontal distractors become active. Similarly, in the orientation feature map, the target as well as blue-vertical distractors become active. Due to spatial interactions within each feature map, the target and the distractors cancel each other. In the resulting salience map, the salience of the target and the distractors are comparable, hence, leading to an inefficient search.

We then tested all models on 100 color-feature searches (where the target differed from the distractors only in color), 100 orientation-feature searches (where the target differed from the distractors only in orientation), and 100 conjunction searches (where the target differed from the distractors in either color or orientation). In each category, we plotted the reaction time (time taken by the models to detect the target) against increasing number of items in the display (density of display was maintained a constant while the display size was varied). As shown in Fig. 12, while the random model and Rao et al.'s model showed no difference in performance across search categories, our biased and unbiased models correctly predicted pop-out in single-feature searches and confirmed the linear increase in reaction time with increasing set size, as it typical in conjunction searches. That is, as soon as Rao et al.'s model was able to reliably detect pop-out targets (by tuning λ), it had become sensitive enough so as to also reliably detect conjunction targets. This result casts doubt on the fact that a template-matching computation like that proposed in Rao et al.'s model may occur in the primate brain. Our biased model, as expected from our intuitive

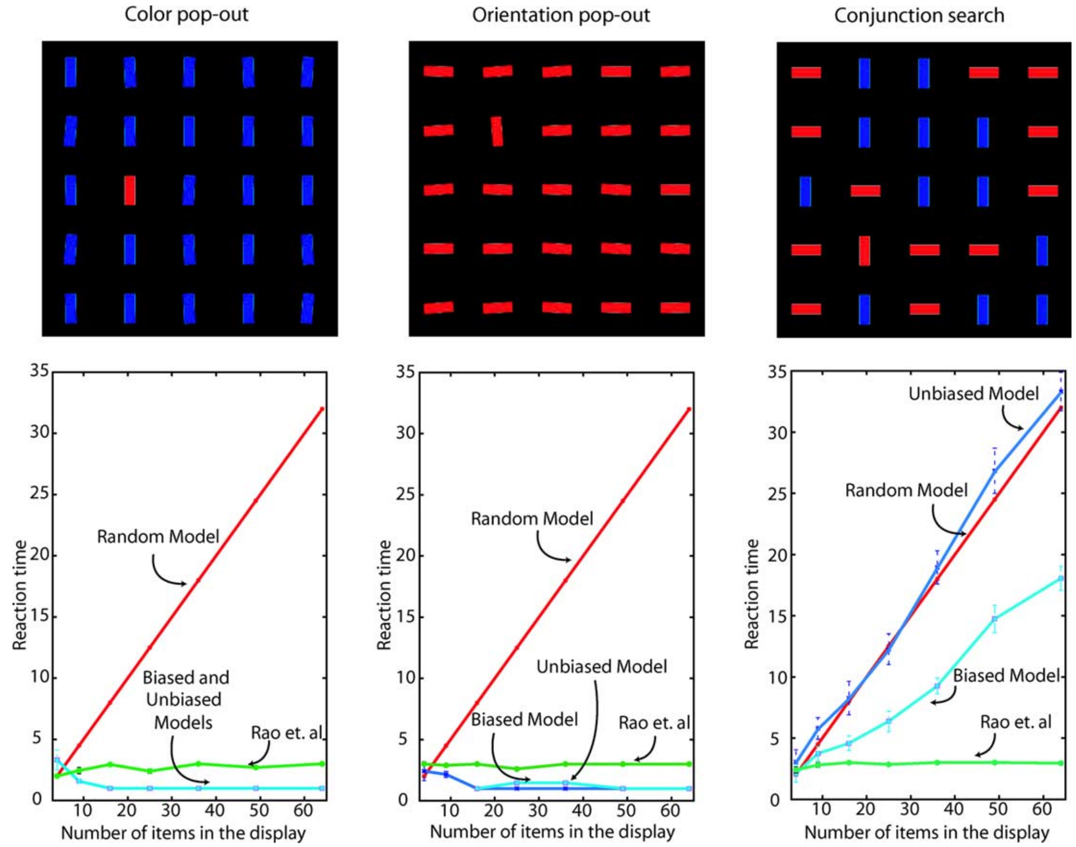


Fig. 12. Comparison between the performance of different models: This figure shows a comparison between the performance of a random model, our unbiased model, our biased model, and a top-down model as proposed by Rao et al. The performance of the models is compared on search arrays creating pop-out in color (first column), pop-out in orientation (second column), and serial, conjunction searches (third column). The *x*-axis shows the number of items in the display and the y axis shows the reaction time (RT) measured as the number of fixations engaged by the model before target detection. The random model assumes uniform probability of attending to each item in the display, hence, on an average, it attends to half the total number of items in the display before finding the target. In single feature searches, our unbiased (unknown target) and biased (known target) models, along with Rao's model (known target) correctly predict efficient search as shown in columns 1 and 2. However, in conjunction searches as shown in column 3, Rao's model continues to predict efficient search (slope = 0, reaction time does not change with increasing number of items in the display), while our unbiased and biased models show an approximately linear increase in reaction time with increasing number of items in the display, which is typical of inefficient searches.

analysis (target salience must increase as it was amplified in two feature maps but distractors only in one), performed slightly better in the conjunction searches than the unbiased model.

Next, we determined our model's ability to perform one-shot learning. An example is shown in Fig. 13 where the model learned a specific instance of a handicap sign from one image and used the learned instance to detect new handicap signs in novel poses, sizes and backgrounds. We tested this one-shot-learning mechanism on 28 test images and as shown by the statistics in Tables 1 and 2, the model accelerated detection over twofold on average. When we allowed the model to learn all instances and combine them to form a general target representation, it allowed for greater variance in the possible target shapes and sizes. While, on the one hand, increased variance in feature values allows detection and categorization of modified targets under the same general object category, on the other hand, it decreases detection speed due to the uncertainty in the exact target features. Hence, biasing for the general object representation led to a small drop in efficiency as compared to biasing for the learned instance. Finally, when we allowed the model to detect the same instance that it had learned, it was most efficient. These results support studies in psychophysics suggesting that better or more exact knowledge of the target leads to better searches (Kenner & Wolfe, 2003).

For multiple target detection, the visual WM used the target representations previously learned and stored in the visual LTM (as stated earlier, for learning, we used 12 training images per target object). The model biased for the multiple task-relevant targets sequentially in decreasing order of their relevance. As mentioned earlier in this section (exemplified with the conjunction search arrays of Fig. 12), biasing is likely, but not guaranteed to make the target most salient. Hence, a less relevant target may be detected while biasing for the most

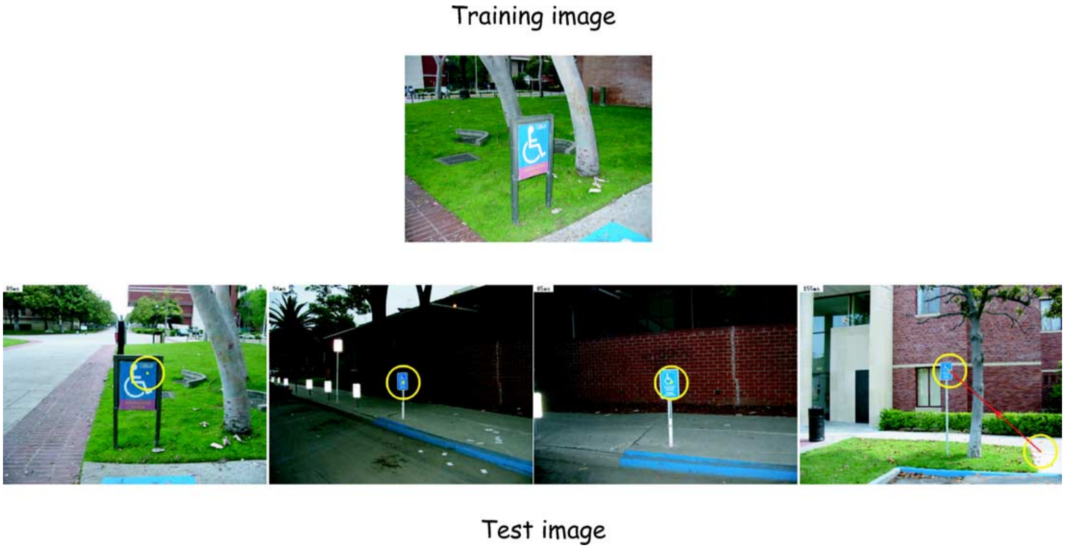


Fig. 13. One-shot learning: the model learned a specific instance of the handicap sign from the image shown in the center and used the learned instance to detect new handicap signs in different poses, sizes and backgrounds as shown in the other images.

Table 1
Statistics of target salience as computed by the biased model over that computed by the naive unbiased model

Operating mode	μ	σ	95% Confidence	Min	Max
Learned-instance	2.72	1.73	[0.91, 4.54]	0.91	5.01
General-object	2.67	1.79	[0.79, 4.54]	0.87	5.39
Exact-instance	3.47	2.45	[0.90, 6.03]	0.95	7.50

The first column states the target representation that was used for biasing (see Section 7 for details); the second column shows the mean improvement in target salience; the third column shows the standard deviation; the fourth column shows the 95% confidence interval; the fifth and sixth columns show the minimum and maximum improvements obtained.

Table 2
Statistics of target detection time as taken by the naive unbiased model over that taken by the biased model

Operating mode	μ	σ	95% Confidence	Min	Max
Learned-instance	2.24	1.27	[0.91, 3.58]	1.00	4.35
General-object	2.22	1.24	[0.92, 3.52]	1.00	4.26
Exact-instance	2.25	1.27	[0.92, 3.58]	1.00	4.35

The first column states the target representation that was used for biasing (see Section 7 for details); the second column shows the mean improvement in target detection time; the third column shows the standard deviation; the fourth column shows the 95% confidence interval; the fifth and sixth columns show the minimum and maximum improvements obtained.

relevant target. Our model handles such errors by recognizing the fixated entity and updating the state of the task graph in the symbolic WM to indicate that it has found the less relevant target, and it proceeds to detect the most relevant target by repeating the above steps. We tested multi-target detection and recognition on 28 new scenes containing fire hydrants and handicap signs. Since the influence of the gist on TRM is not implemented in our model yet, we placed the targets at random locations to eliminate the role of the gist in aiding the detection of the targets. Results showed that, on average, our model was 6.20 times faster than the naive unbiased model (95% confidence interval = [1.47, 10.94], min = 0.07, max = 28.86; Fig. 14). In these experiments, we thus tested the top-down biasing and recognition components involving visual WM and LTM modules, and the symbolic WM and LTM modules for creating and maintaining the task graph.

To further test the recognition module, we allowed the model to recognize the entity at the attended location by matching the visual features extracted at the fixation against those stored in the object hierarchy in visual LTM. Despite the simplicity of the model (it attempts to recognize fixations by looking at just one location in the object), it seems to be able to classify the target in the appropriate category of objects—as shown in Fig. 15, the contributors for false negatives and false positives share features with the target, i.e., they are similar to the target.

Next, we attempted to determine and learn the task-relevant locations in the scene. The visual WM, with the help of visual LTM, biased the attentional system for

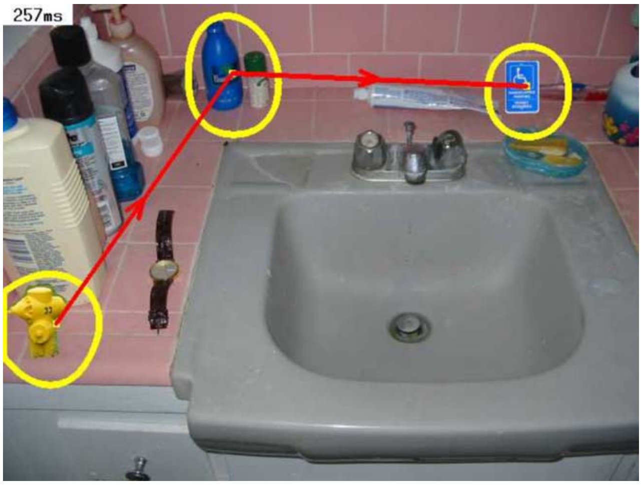


Fig. 14. Sequential detection of multiple targets: The model initialized the working memory with the targets to be found and their relevance (handicap sign, relevance = 1; fire hydrant, relevance = 0.5). It biased for the most relevant target (in this case, the handicap sign), made a false detection, recognized the fixation (fire hydrant), updated the state in its working memory (recorded that it found the fire hydrant), and proceeded to detect the remaining target by repeating the above steps.

the task-relevant target. Initially, the model had no prior knowledge of the scene, hence the TRM was uniform (baseline 1.0 everywhere), and the model attended to scene locations based on their visual salience. For each incoming visual scene, the TRM was updated as follows: at each fixation, the recognition module, with the help of visual LTM, recognized the entity at the attended scene location. The symbolic WM, with the aid of symbolic LTM, determined the task-relevance of the recognized entity. It marked the corresponding location in the TRM with the estimated relevance. To learn the contents of the TRM across all the incoming scenes, we computed the average TRM in an online and incremental manner (for this purpose, we maintained the sum of TRMs and the number of TRMs or scenes seen so far). As shown below, we designed a task in a dynamic environment to test the learning and working of the TRM. The other modules that were also involved in the test include the top-down biasing and recognition modules, the working memory and the long-term memory modules.

For a driving task, we allowed the model to bias for cars and attend to the salient scene locations and

Target	% False positives	Top Contributor	% False negatives	Contributor1	Contributor2	Contributor3
⬢	64.53	⬢	20.00	⬢ (20.00)		
⬢	30.55	⬢	60.00	⬢ (50.00)	▪ (10.00)	
⚪	0.00	⬜	100.00	⬢ (70.00)	⬢ (30.00)	
=	0.00	/\	10.00	○ (10.00)		
/\	0.00	=	0.00			
O	18.18	0	10.00	0 (10.00)		
0	0.00	O	10.00	O (10.00)		
○	3.97	⚲	100.00	⚲ (30.00)	⬢ (20.00)	▯ (10.00)
⚲	0.00	○	70.00	○ (40.00)	▯ (30.00)	
□	6.71	▯	70.00	○ (20.00)	▯ (10.00)	▮ (10.00)
▪	8.86	⬜	30.00	···· (30.00)		
🟩	0.00	🟥	80.00	🟥 (80.00)		
🟥	0.00	🟩	0.00			
····	7.77	▬	90.00	▬ (70.00)	▪ (20.00)	

Fig. 15. Statistics for the hierarchical recognition of arbitrary fixations, for a sample of objects from our database. As an initial implementation, we considered a simple object hierarchy with just three main levels (level 1: all objects, level 2: instances and level 3: views) and at level 0 was a dummy root that was a general class combining all the objects. The first column is the target object; the second column shows the percentage of false positives (number of distractors that were falsely recognized as the target, over the total number of distractors); the third column shows the distractor that accounted for the false positives; the fourth column shows the percentage of false negatives (number of targets that were not recognized as the target, over the total number of targets); the fifth, sixth and seventh column show the top 3 contributors to false negatives. Despite the simplicity of the model (it attempts to recognize fixations by looking at just one location in the object), it seems to be able to classify the target in the appropriate category of objects—as shown in this figure, the contributors for false negatives and false positives share features with the target, i.e., they are similar to the target.

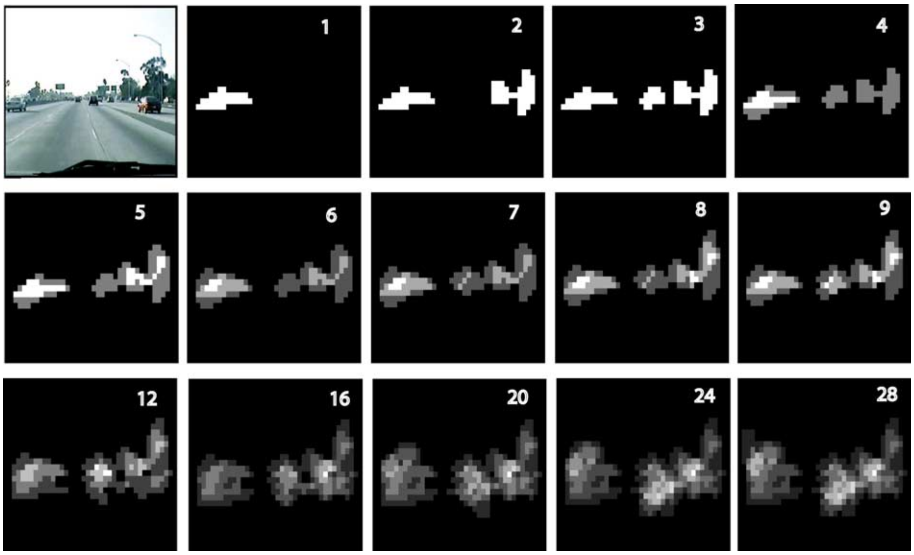


Fig. 16. Learning the TRM. The model learned the TRM for a driving task by attending, estimating the relevance of attended scene locations and updating the TRM. The development of the TRM across 28 fixations is shown here (brighter shades of grey indicate locations more relevant than baseline). Note that the TRM does not change significantly after a while and is learned to a reasonable precision within the first 5–10 fixations.

recognize them as belonging to the car or the sky category. Initially, the TRM was unbiased due to the lack of any knowledge of the scene. As the model attended and recognized locations as belonging to the car category, the relevance of these locations was updated in the TRM. The development of the TRM over a number of fixations is shown in Fig. 16.

On the same scenes as used for the driving task, we attempted to learn the scene locations that belonged to the sky category. We repeated exactly the same steps as above and obtained the TRM as shown in Fig. 17. Thus, we explored how different locations in the same scene become relevant as the task changes.

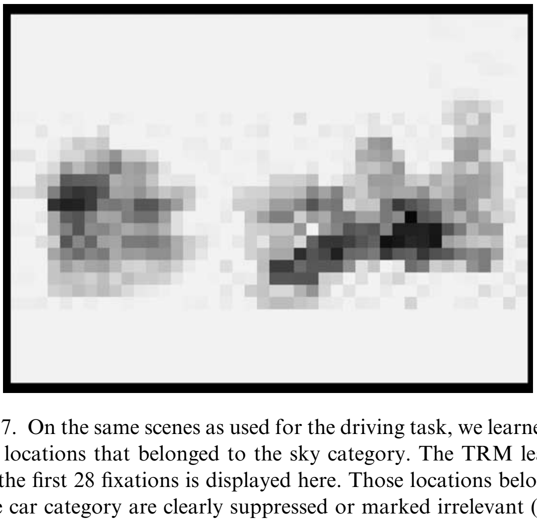


Fig. 17. On the same scenes as used for the driving task, we learned the scene locations that belonged to the sky category. The TRM learned after the first 28 fixations is displayed here. Those locations belonging to the car category are clearly suppressed or marked irrelevant (dark) compared to baseline (white). It may appear misleading that the road is marked as relevant. Since the road was non-salient, it did not attract any attention and hence was not marked as irrelevant and remained at baseline.

8. Discussion

In this paper, we have designed and partially implemented an overall biologically plausible architecture to model how different factors such as bottom-up cues, knowledge of the task and target influence the guidance of attention. We have tested our model on a variety of tasks including search tasks in static scenes and a driving task in dynamic scenes. The results show that our model can determine the task-relevant targets from a given task definition; detect the targets amidst clutter and diverse backgrounds; reproduce basic human visual search behavior; recognize many targets and classify them into their corresponding categories with few errors; learn the task-relevant locations online in an incremental manner, and use the learned target features as well as likely target locations to bias the attentional system to guide attention towards the target. In the rest of this section, we discuss our main contributions in this paper, namely target representation, target detection, recognition, and memorization, followed by a brief discussion on scene representation. Finally, we present the limitations of our model, along with future directions.

8.1. Target representation

Our model represents the target by center-surround features at different spatial scales, where the coarse scales include background information and the finer scales contain target information. Traditional approaches attempt to segment the target from the background in order to avoid the confusion between the target and the background. In simple cases where the object appears in similar sizes but in different backgrounds, our model achieves the equivalent of segmentation by determining the scales that reliably represent the target. If the background is inconsistent or changing, its variability is reflected in the high variance in response to the features at coarse spatial scales. Consequently, those features are considered unreliable and are not promoted during biasing for the target. In other cases where the background is consistent, the co-occurrence of the target and its background is captured by the low variance in response to the features at the coarse scales. Thus, our target representation provides a convenient way to include contextual information.

8.2. Target detection

In our model, feature maps are computed in parallel, non-linear interactions occur in all of them (Itti & Koch, 2001b), and they are weighted in a top-down manner before being summed into the salience map. The target is made salient by adjusting the weights of the low-level feature maps so as to promote the target's relevant features and suppress its irrelevant features. Thus, our model provides a computational implementation of a Guided Search mechanism (Wolfe, 1994), and it learns the appropriate feature weights directly from training images containing the targets. Consequently, our model predicts that all scene locations whose features are a superset of the target's features or share it also become salient, e.g., a red ellipse also becomes salient if we are searching for a red circle. This prediction of the model could be verified with psychophysics experiments. In addition to top-down factors that influence target salience, bottom-up factors such as spatial non-linear interactions modulate the target salience based on the salience of the neighboring distractors (Duncan & Humphreys, 1989; Moraglia, 1989; Nothdurft, 1992). The winner of the spatial competition depends on the positions and relative salience of the target and distractors. Hence, biasing is likely, but not guaranteed to make the target most salient. In important cases like conjunction searches (Figs. 11 and 12), we have shown how biasing is fairly ineffective in our model, in agreement with human data. This reinforces the plausibility of the biasing approach proposed in our model, especially compared to template-matching models (Rao et al., 2002) which seem difficult to reconcile with empirical data (as they do not yield pop-out in feature search cases when the target features are unknown, as mentioned in Section 2, but yield pop-out in both feature and conjunction searches alike when target features are known, as shown in Section 7).

8.3. Target recognition

The object recognition model proposed here is simple and shares its resources intimately with the attentional system by re-using in target representation the pre-attentive features computed for guiding attention. Hierarchical matching from general representations like object categories to specific representations like the object, instance or view allows us to terminate the search at the appropriate level of representation, depending on our task requirements (e.g., distinguishing between a white and red object may not require processing down to the level of instances such as white car, or white horse). Further, by pruning the subtrees (in the object hierarchy) that do not match, we can accelerate the search for the best match. Currently, our model attempts to recognize an object by matching any one location's visual features against all learned representations, hence, there are false recognitions and limitations on the complexity of objects that can be recognized. Though it cannot recognize complex objects, this could possibly be achieved by decomposing the complex object into a spatial configuration of simpler objects (parts) (Wiskott et al., 1997), that could each be recognized using our proposed schema. A higher-level mechanism can then check for the spatial relations between the parts to recognize the whole. However, in this paper, our aim is to explore how the pre-attentive features used to guide attention may be re-used for object representation and recognition. Since we represent the target as a feature vector, we do not explicitly handle complex or composite objects in the current model. Yet, our results indicate that the model could recognize some complex objects such as geometrical shapes including rectangles, cubes and striped bars to a reasonable extent (see Fig. 15).

8.4. Memorization

On the one hand, we have symbolic knowledge that deals with high-level concepts and objects. On the other hand, there are low-level neural maps of the scene that encode salience or other image attributes at each pixel or image location. To bridge the gap between these extreme representations, we have proposed a two-dimensional topographic map called task-relevance map (TRM) that encodes task-relevance of the scene entities. To memorize the target, an area in the TRM corresponding to the locations and approximate size and shape of the target (Walther, Itti, Reisenhuber, & Poggio, 2002) is highlighted with the target's relevance, and visual features are stored in visual working memory along with links to symbolic knowledge. The TRM has several potential uses as explained below. It helps to prime a particular scene location by increasing its relevance in the TRM, thus supporting spatial top-down attentional modulation. The TRM also helps in object detection in the following manner. Non-attentional scene representations such as gist and layout have been shown to play an important role in object detection (Biederman et al., 1982; Chun & Jiang, 1998; De Graef et al., 1990; Henderson & Hollingworth, 1999; Palmer, 1975; Rensink, 2000; Torralba, 2003). Our model suggests an easy way to incrementally learn the relation between gist and the constituent scene objects. We suggest that the TRM may be used to learn object properties such as locations where an object is likely to occur and its approximate size. The relation between gist and object properties may be learned by maintaining a loop between the gist and the TRM (via working memory). During the feedforward loop, the quick and imprecise gist may be used to retrieve the appropriate, previously learned TRM and use it as an initial guide to drive the focus of attention. Subsequently, by the slow and precise processes of attending and updating, the TRM can be refined and learned online in an incremental manner within the first few fixations, and be used to drive further fixations. Finally, the feedback loop may use the TRM to reinforce, confirm or even update the gist. It may also be used to store the currently learned TRM.

8.5. Scene representation

Knowledge of gist, visual features and location of the object may be important for scene understanding and representation, but they are not sufficient. Consider the following example of a scene with a man, a laptop and a cake. In order to understand the scene, we need to know how the entities are bound or related to each other. If the man and the laptop are bound by the 'work' action, then we can conclude that the man is working. Else if the man and the cake are bound by the 'eat' action, we can conclude that the man is eating the cake. To represent such relationships in our model, the symbolic working memory (WM) maintains relations among entities by seeking the help of the symbolic long-term memory. However, we do not make any claims on the biological feasibility of our current implementation. It is not clear to us as to how these relations may be represented in our brain and how the entities may be bound together into composite structures.

Our model presents the following hypothesis on how a scene may be represented. To bind the symbolic attributes of the attended object with its visual features and its location, our model suggests the creation and maintenance of a link between the object in the symbolic WM, its visual features in the visual WM and the corresponding location in the TRM. This constitutes our explicit representation of an object file (Kahneman & Treisman, 1984). These links can be very useful in recall, e.g., an object at a particular location may be recalled by activating that location in the TRM, that in turn activates the link and the associated object. Similarly, where we saw a given object may be recalled by activating the object in the working memory that in turn activates the link to the corresponding location.

The following discussion, though not directly tied to the reported model, is an interesting detour that explores the role of the links (that bind visual and symbolic properties of the stimuli) in scene representation. We propose to use the above links for scene representation by extending Rensink's triadic architecture (Rensink, 2000) as follows. He proposes a coherence field where a spatio-temporal structure is created at the focus of attention and is lost when the focus of attention shifts. Rensink suggests that the low level visual stages such as proto-objects are volatile and are bound only at the focus of attention. We extend that hypothesis and suggest that while the low-level visual stages may be volatile, high-level visual stages such as the WM (and, further, LTM) may not be volatile and may store the recently attended relevant objects, their locations and their visual features, even though they may not be the current focus of attention (Hollingworth, 2004; Hollingworth & Henderson, 2002; Hollingworth et al., 2001). This is consistent with studies showing that visual representation at high-level visual stages may be impoverished and less precise than their low level counterparts (Irwin, 1991; Phillips, 1974), but they can be maintained for longer durations under backward pattern masking (Phillips, 1974) and across saccades (Irwin, 1992b). Hence, in our representation, the links between the TRM and objects in short term memory or working memory do not die when the focus of attention shifts. But several studies have shown that there exist strict limitations (∼4) on how many object files may coexist at any given time (Irwin, 1992a; Irwin & Zelinsky, 2002; Luck & Vogel, 1997; Pashler, 1988; Sperling, 1960). This implies that there must be some competition among the links so that the strong links may survive and the weaker ones may die (see (Schneider, 1999) for an activation level based competition). We suggest that the strength of the link depends on the relevance of the associated object, perhaps directly proportional. Hence, links are not established for irrelevant objects and, consequently, their visual features or locations are forgotten. Older links suffer interference from newer links and gradually weaken and die. A new link also suffers interference from existing links and may die if its relevance *is* not high. Thus, links to irrelevant objects/locations or those seen in the remote past may die or disappear whereas links to the relevant objects/locations seen recently may be strong and consequently, we remember the associated details.

8.6. Limitations of our current model

Our current implementation of the model has a number of limitations. For example, the model cannot yet make directed attentional shifts. Including directed attentional shifts into our model would require that spatial relations also be included in our ontology (e.g., look upwards if searching for a face, but found a foot) and would allow for more sophisticated top-down attentional control. Knowledge of such spatial relationships will also help us prune the search space by filtering out most irrelevant scene elements (e.g., while looking for John, if we see Mary's face, we can also mark Mary's hands, legs, etc. as irrelevant provided we know the spatial relationships). Several models already mentioned provide an excellent starting point for this extension of our model (Rybak et al., 1998). Our model also does not support instantiation such as "John is an instance of a man" where each instance is unique. The model currently uses absolute scales as a signature for an object. This is undesirable for real vision where the scales change with changes in viewing distance, and pose. This issue can be addressed by using a scale invariant object representation where all scales are considered relative to the dominant scale. Currently, the object hierarchy stored in the visual long-term memory is partially hand-coded, i.e., we have to manually group a set of images as belonging to the same object, but given an image and a location cue, our model can automatically extract the views. To make the object hierarchy fully learnable, we could allow the model to fixate arbitrarily and if the fixated entity is new, it could learn the features and automatically classify the new entity into some object category and update the hierarchy (but note that such incremental unsupervised building of object categories is a particularly difficult problem). The knowledge base in symbolic long-term memory is also currently hand-coded. For the purpose of testing our model, we considered human-related objects, actions, body parts and their relationships. Extensive research in knowledge representation has led to several ontologies for various contexts including the animal kingdom and behavior, weather, ceramics, congress-related events, managing an enterprise, and many more (Ontologies, 2003); our ontology may be extended by importing these.

8.7. Conclusion

In this paper, we have proposed and partially implemented a computational model for the task-specific guidance of attention in real-world scenes. Our main contributions in this paper are: First, providing a biologically plausible architecture for object detection, by top-down biasing the bottom-up attentional system for the object's pre-attentive features so as to make the object more salient; second, object recognition by re-using the pre-attentive features for object representation and matching hierarchically against stored representations; and, third, memorization of relevant scene locations in visual working memory by learning their locations and approximate sizes in a topographic two-dimensional task-relevance map. We have also proposed a non-biological computational scheme to estimate the task-relevance of scene entities using an ontology containing entities and their relationships. Thus, given a task specification, our model determines the task-relevant entities, biases for the current most task-relevant entity, recognizes the fixated entity, memorizes the task-relevance of the fixated entity, updates its working memory and repeats the process until the task is complete. The promising results of our model suggest that the model may provide a reasonable approximation to many of the brain processes involved in complex task-driven visual behaviors. As part of our future work, we are planning to further confirm the above by comparing our model's performance against human eye tracking data (Itti, 2004).

Acknowledgments

This project was supported by the National Science Foundation, National Eye Institute, National Imagery and Mapping Agency, Zumberge Innovation Fund, and Charles Lee Powell Foundation. The opinions expressed here are those of the authors and may not necessarily represent the views of the supporting agencies.

Appendix A

Here, we show the derivation of the class conditional density, $P(O=o|\,Y)$ of super-class Y that is formed by combining several equally likely and mutually exclusive object classes X_i (refer to Section 5.1).

$$
\begin{aligned}
P(O=o \mid Y) =& P\left(O=o \mid \bigcup_i X_i\right) \quad \text{(using Eq.(2))}\\
=& P\left(O=o, \bigcup_i X_i\right) \Big/ P\left(\bigcup_i X_i\right)\\
& \text{(using Bayes rule)}\\
=& P\left(\bigcup_i X_i \mid O=o\right) P(O=o) \Big/ P\left(\bigcup_i X_i\right)\\
& \text{(using Bayes rule)}\\
=& \sum\nolimits_i P(X_i \mid O=o) P(O=o) / \sum\nolimits_i P(X_i)\\
& \text{(since } X_i \text{ are mutually exclusive)}\\
=& \sum\nolimits_i P(X_i, O=o) / \sum\nolimits_i P(X_i)\\
& \text{(using Bayes rule)}\\
=& \sum\nolimits_i P(O=o \mid X_i) P(X_i) / \sum\nolimits_i P(X_i)\\
& \text{(using Bayes rule)}\\
=& \sum\nolimits_i P(O=o \mid X_i) w_i \qquad (10)\\
\textit{where } w_i =& P(X_i) / \sum\nolimits_j P(X_j)\\
=& 1/n \quad \text{(since } X_i \text{ are equally likely)}
\end{aligned}
$$

The mean of $O|Y$ is derived as follows:

$$\begin{aligned}
E[O \mid Y] =& \int_o oP(O=o \mid Y)\,\mathrm{d}o \qquad (11)\\
=& \int_o o\Big(\sum\nolimits_i P(O=o \mid X_i)w_i\Big)\,\mathrm{d}o\\
& \text{(using Eq. (10))}\\
=& \sum\nolimits_i w_i\Big(\int_o oP(O=o \mid X_i)\,\mathrm{d}o\Big)\\
=& \sum\nolimits_i w_i E[O \mid X_i]\\
& \text{(substituting } Y \text{ by } X_i \text{ in Eq. (11))}\\
\mu =& \sum\nolimits_i w_i\mu_i
\end{aligned}$$

By definition of variance,

$$\begin{aligned}
\sigma_i^2 =& E[(O \mid X_i - E[O \mid X_i])^2]\\
=& E[(O \mid X_t)^2] - (E[O \mid X_i])^2\\
\sigma_i^2 =& E[(O \mid X_i)^2] - \mu_i^2 \qquad (12)\\
\sigma^2 =& E[(O \mid Y)^2] - \mu^2 \quad \text{(similarly)} \qquad (13)\\
E[(O \mid Y)^2] =& \int_o o^2P(O=o \mid Y)\,\mathrm{d}o\\
& \text{(by definition of expectation)}\\
=& \int_o o^2\Big(\sum\nolimits_i P(O=o \mid X_i)w_i\Big)\,\mathrm{d}o\\
& \text{(using Eq. (10))}\\
=& \sum\nolimits_i w_i\Big(\int_o o^2P(O=o \mid X_i)\,\mathrm{d}o\Big)\\
=& \sum\nolimits_i w_iE[(O \mid X_i)^2]\\
& \text{(by definition of expectation)}\\
=& \sum\nolimits_i w_i(\sigma_i^2+\mu_i^2) \quad \text{(using Eq. (12))} \qquad (14)\\
\sigma^2 =& \sum\nolimits_i w_i(\sigma_i^2+\mu_i^2) - \mu^2\\
& \text{(using Eqs. (13) and (14))} \qquad (15)
\end{aligned}$$

References

Arman, F., & Aggarwal, J. K. (1993). Model-based object recognition in dense-range images—a review. *ACM Computing Surveys (CSUR), 25*(1), 5–43.

Bauer, B., Jolicoeur, P., & Cowan, W. B. (1996). Visual search for colour targets that are or are not linearly-separable from distractors. *Vision Research, 36*(10), 1439–1465.

Beck, J., Prazdny, K., & Rosenfeld, A. (1983). *A theory of textural segmentation*. New York: Academic Press.

Biederman, I. (1987). Recognition-by-components: A theory of human image understanding. *Psychological Review, 94*, 115–147.

Biederman, I., Mezzanotte, R. J., & Rabinowitz, J. C. (1982). Scene perception: Detecting and judging objects undergrouping relational violations. *Cognitive Psychology, 14*, 143–177.

Bilsky, A. B., & Wolfe, J. M. (1994). Part-whole information is useful in visual search for size × size but not orientation × orientation conjunctions. *Perception and Psychophysics, 57*(6), 749–760.

Blaser, E., Sperling, G., & Lu, Z. L. (1999). Measuring the amplification of attention. *Proceedings of the National Academy of Sciences of the United States of America, 96*(20), 11681–11686.

Brachman, R. J., & Levesque, H. J. (1985). Morgan Kaufmann Publishers.

Buracas, G. T., Albright, T. D., & Sejnowski, T. J. (1996). Varieties of attention: A model of visual search. *Institute of Neural Computation Proceedings of the 3rd Joint Symposium on Neural Computation, 6*, 11–25.

Burt, P. J., & Adelson, E. H. (1983). The laplacian pyramid as a compact image code. *IEEE Transactions on Communications, 31*, 532–540.

Chun, M. M., & Jiang, Y. (1998). Contextual cueing: Implicit learning and memory of visual context guides spatial attention. *Cognitive Psychology, 36*, 28–71.

Colby, C. L., & Goldberg, M. E. (1999). Space and attention in parietal cortex. *Annual Review of Neuroscience, 22*, 319–349.

Courtney, S. M., Ungerleider, L. G., Keil, K., & Haxby, J. V. (1996). Object and spatial visual working memory activate separate neural systems in human cortex. *Cerebral Cortex, 6*(1), 39–49.

De Graef, P., Christiaens, D., & d'Ydewalle, G. (1990). Perceptual effects of scene context on object identification. *Psychological Research, 52*, 317–329.

DeValois, R. L., Albrecht, D. G., & Thorell, L. G. (1982). Spatial-frequency selectivity of cells in macaque visual cortex. *Vision Research, 22*, 545–559.

Driver, J., McLeod, P., & Dienes, Z. (1992). Motion coherence and conjunction search: Implications for guided search theory. *Perception and Psychophysics, 51*(1), 79–85.

Duncan, J., & Humphreys, G. W. (1989). Visual search and stimulus similarity. *Psychological Review, 96*, 433–458.

Engel, S., Zhang, X., & Wandell, B. (1997). Colour tuning in human visual cortex measured with functional magnetic resonance imaging [see comments]. *Nature, 388*(6637), 68–71.

Enns, J. T. (1986). Seeing textons in context. *Perception and Psychophysics, 39*, 143–147.

Gottlieb, J. P., Kusunoki, M., & Goldberg, M. E. (1998). The representation of visual salience in monkey parietal cortex. *Nature, 391*, 481–484.

Haenny, P. E., & Schiller, P. H. (1988). State dependent activity in monkey visual cortex. Single cell activity in VI and V4 on visual tasks. *Experimental Brain Research, 69*, 245–259.

Henderson, J. M., & Hollingworth, A. (1999). High level scene perception. *Annual Review of Psychology, 50*, 243–271.

Herzog, G., & Wazinski, P. (1994). Visual translator: Linking perceptions and natural language descriptions. *Artificial Intelligence Review, 8*(2-3), 175–187.

Hollingworth, A. (2004). Constructing visual representations of natural scenes: The roles of short- and long-term visual memory. *Journal of Experimental Psychology: Human Perception and Performance, 30*, 519–537.

Hollingworth, A., & Henderson, J. M. (2002). Accurate visual memory for previously attended objects in natural scenes. *Journal of Experimental Psychology: Human Perception and Performance, 28*, 113–136.

Hollingworth, A., Williams, C. C., & Henderson, J. M. (2001). To see and remember: Visually specific information is retained in memory from previously attended objects in natural scenes. *Psychonomic Bulletin and Review, 8*, 761–768.

Huk, A. C., & Heeger, D. J. (2000). Task-related modulation of visual cortex. *Journal of Neurophysiology, 83*, 3525–3536.

Irwin, D. E. (1991). Information integration across saccadic eye movements. *Cognitive Psychology, 23*, 420–456.

Irwin, D. E. (1992a). Memory for position and identity across eye movements. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 18*, 307–317.

Irwin, D. E. (1992b). Visual memory within and across fixations. In K. Rayner (Ed.), *Eye movements and visual cognition: Scene perception and reading*. New York: Springer-Verlag.

Irwin, D. E., & Andrews, R. (1996). Integration and accumulation of information across saccadic eye movements. In *Attention and performance XVI: Information integration in perception and communication* (pp. 125–155). Cambridge, MA: MIT Press.

Irwin, D. E., & Zelinsky, G. J. (2002). Eye movements and scene perception: Memory for things observed. *Perception and Psychophysics, 64*, 882–895.

Itti, L. (2004). Automatic foveation for video compression using a neurobiological model of visual attention. *IEEE Transactions on Image Processing, 13*(10).

Itti, L., & Koch, C. (2000). A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research, 40*(10-12), 1489–1506.

Itti, L., & Koch, C. (2001a). Computational modeling of visual attention. *Nature Reviews Neuroscience, 2*(3), 194–203.

Itti, L., & Koch, C. (2001b). Feature combination strategies for saliency-based visual attention systems. *Journal of Electronic Imaging, 10*(1), 161–169.

Itti, L., Koch, C., & Niebur, E. (1998). A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 20*(11), 1254–1259.

Julesz, B. (1971). *Foundations of cyclopean perception*. Chicago, Illinois: University of Chicago Press.

Julesz, B., & Bergen, J. R. (1983). Textons, the fundamental elements in preattentive vision and perception of textures. *The Bell System Technical Journal, 62*(6), 1619–1645.

Kahneman, D., & Treisman, A. (1984). Changing views of attention and automaticity. In D. Parasuraman, R. Davies, & J. Beatty (Eds.), *Varieties of attention* (pp. 29–61). New York, NY: Academic.

Kanwisher, N. (1987). Repetition blindness: Type recognition without token individuation. *Cognition, 27*, 117–143.

Kenner, N., & Wolfe, J. M. (2003). An exact picture of your target guides visual search better than any other representation [abstract]. *Journal of Vision, 3*(9), 230a.

Koch, C., & Ullman, S. (1985). Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology, 4*(4), 219–227.

Kustov, A. A., & Robinson, D. L. (1996). Shared neural control of attentional shifts and eye movements. *Nature, 384*, 74–77.

Lades, M., Vorbrüggen, J. C., Buhmann, J., Lange, J., von der Malsburg, C., Würtz, R. P., & Konen, W. (1993). Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers, 42*, 300–311.

Leventhal, A. G. (1991). *The neural basis of visual function. Vision and visual dysfunction* (Vol. 4). Boca Raton, FL: CRC Press.

Luck, S. J., & Vogel, E. K. (1997). The capacity of visual working memory for features and conjunctions. *Nature, 390*, 279–281.

Luschow, A., & Nothdurft, H. C. (1993). Pop-out of orientation but no pop-out of motion at isoluminance. *Vision Research, 33*(1), 91–104.

Moraglia, G. (1989). Display organization and the detection of horizontal line segments. *Perception and Psychophysics, 45*, 265–272.

Moran, J., & Desimone, R. (1985). Selective attention gates visual processing in the extrastriate cortex. *Science, 229*(4715), 782–784.

Motter, B. C. (1993). Focal attention produces spatially selective processing in visual cortical areas VI, V2, and V4 in the presence of competing stimuli. *Journal of Neurophysiology, 70*(3), 909–919.

Motter, B. C. (1994a). Neural correlates of attentive selection for color or luminance in extrastriate area V4. *Journal of Neuroscience, 14*(4), 2178–2189.

Motter, B. C. (1994b). Neural correlates of feature selective memory and pop-out in extrastriate area V4. *Journal of Neuroscience, 14*(4), 2190–2199.

Nagy, A. L., & Sanchez, R. R. (1990). Critical color differences determined with a visual search task. *Journal of the Optical Society of America A, 7*(7), 1209–1217.

Nagy, A. L., & Sanchez, R. R. (1992). Chromaticity and luminance as coding dimensions in visual search. *Human Factors, 34*(5), 601–614.

Nakayama, K., & Silverman, G. H. (1986). Serial and parallel processing of visual feature conjunctions. *Nature, 320*, 264–265.

Norton, D., & Stark, L. (1971). Scanpaths in saccadic eyemovements during pattern perception. *Science*, 308–311.

Nothdurft, H. C. (1992). Feature analysis and the role of similarity in preattentive vision. *Perception and Psychophysics, 52*(4), 355–375.

Oliva, A., & Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision, 42*(3), 145–175.

Ontologies (2003). Available http://www.cs.utexas.edu/users/mfkb/related.html, http://saussure.irmkant.rm.cnr.it/onto/link.html.

O'Regan, J. K. (1992). Solving the "Real" Mysteries of Visual Perception: The World as an Outside Memory. *Canadian Journal of Psychology, 46*, 461–488.

Palmer, S. E. (1975). The effect of contextual scenes on the identification of objects. *Memory and Cognition, 3*, 519–526.

Pashler, H. (1988). Familiarity and the detection of change in visual displays. *Perception and Psychophysics, 44*, 369–378.

Phillips, W. A. (1974). On the distinction between sensory storage and short-term visual memory. *Perception and Psychophysics, 16*, 283–290.

Rao, R. P., Zelinsky, G., Hayhoe, M., & Ballard, D. H. (2002). Eye movements in iconic visual search. *Vision Research, 42*(11), 1447–1463.

Rensink, R. A. (2000). The dynamic representation of scenes. *Visual Cognition, 7*, 17–42.

Rensink, R. A. (2002). Change Detection. *Annual Review of Psychology, 53*, 245–277.

Rensink, R. A., O'Regan, J. K, & Clark, J. J. (1997). To see or not to see: The need for attention to perceive changes in scenes. *Psychological Science, 8*, 368–373.

Riesenhuber, M., & Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience, 2*(11), 1019–1025.

Riesenhuber, M., & Poggio, T. (2000). Models of object recognition. *Nature Neuroscience*, 1199–1204.

Rybak, I. A., Gusakova, V. I., Golovan, A. V., Podladchikova, L. N., & Shevtsova, N. A. (1998). A model of attention-guided visual perception and recognition. *Vision Research, 38*, 2387–2400.

Saenz, M., Buracas, G. T., & Boynton, G. M. (2002). Global effects of feature-based attention in human visual cortex. *Nature Neuroscience, 5*(7), 631–632.

Schneider, W. X. (1999). Visual-spatial working memory, attention, and scene representation: A neuro-cognitive theory. *Psychological Research, 62*, 220–236.

Sperling, G. (1960). The information available in visual presentations. *Psychological Monographs, 74*, 1–29.

Thompson, K. G., & Schall, J. D. (2000). Antecedents and correlates of visual detection and awareness in macaque prefrontal cortex. *Vision Research, 40*(10–12), 1523–1538.

Thorpe, S. J., Fize, D., & Marlot, C. (1996). Speed of processing in the human visual system. *Nature, 381*, 520–522.

Tootell, R. B., Silverman, M. S., Hamilton, S. L., De Valois, R. L., & Switkes, E. (1988). Functional anatomy of macaque striate cortex. III. Color. *Journal of Neuroscience, 8*(5), 1569–1593.

Torralba, A. (2002). Contextual modulation of target saliency. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems (Vol. 14)*. Cambridge, MA: MIT Press.

Torralba, A. (2003). Contextual priming for object detection. *International Journal of Computer Vision, 53*(2), 153–167.

Treisman, A., & Gelade, G. (1980). A feature integration theory of attention. *Cognitive Psychology, 12*, 97–136.

Treisman, A., & Gormican, S. (1988). Feature analysis in early vision: Evidence from search asymmetries. *Psychological Review, 95*(1), 15–48.

Treue, S., & Maunsell, J. H. (1996). Attentional modulation of visual motion processing in cortical areas MT and MST. *Nature, 382*(6591), 539–541.

Triesch, J., Ballard, D. H., Hayhoe, M. M., & Sullivan, B. T. (2003). What you see is what you need. *Journal of Vision, 3*(1), 86–94.

Triesman, A., & Souther, J. (1986). Illusory Words: The roles of attention and top-down constraints in conjoining letters to form words. *Journal of Experimental Psychology: Human Perception and Performance, 14*, 107–141.

Walther, D., Itti, L., Reisenhuber, M., Poggio, T., & Koch., C. (2002). Attentional selection for object recognition—a gentle way. In *Proc., 2nd workshop on biologically motivated computer vision BMCV2002* (pp. 472–479).

Watanabe, K. (2003). Differential effect of distractor timing on localizing versus identifying visual changes. *Cognition, 88*(2), 243–257.

Weber, M., Welling, M., & Perona, P. (2000). unsupervised learning of models for recognition. In *Proc. 6th Europ. Conf. Comp. Vis., ECCV2000, Dublin, Ireland* (June).

Weichselgartner, E., & Sperling, G. (1987). Dynamics of automatic and controlled visual attention. *Science, 238*(4828), 778–780.

Wilson, F. A., O Scalaidhe, S. P., & Goldman-Rakic, P. S. (1993). Dissociation of object and spatial processing domains in primate prefrontal cortex. *Science, 260*, 1955–1958.

Wiskott, L., Fellous, J.-M., Krüger, N., & von der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. In G. Sommer, K. Daniilidis, & J. Pauli (Eds.), *Proc. 7th Intern. Conf. on Computer Analysis of Images and Patterns, CAIP'97* (pp. 456–463). Kiel, Heidelberg: Springer-Verlag.

Wolfe, J. M. (1994). Guided search 2.0: A revised model of visual search. *Psyonomic Bulletin and Review, 1*(2), 202–238.

Wolfe, J. M., Priedman-Hill, S. R., Stewart, M. I., & O'Connell, K. M. (1992). The role of categorization in visual search for orientation. *Journal of Experimental Psychology: Human Perception and Performance, 18*(1), 34–49.

Wurtz, R. H., Goldberg, M. E., & Robinson, D. L. (1980). Behavioral modulation of visual responses in the monkey: Stimulus selection for attention and movement. *Progress in Psychobiology and Physiological Psychology, 9*, 43–83.

Yarbus, A. (1967). *Eye Movements and Vision*. New York: Plenum Press.