EVALUATION OF ATTENTIONAL CONTROL IN ACTIVE VISION SYSTEMS USING A 3D SIMULATION FRAMEWORK

Gerriet Backer[1], **Bärbel Mertsching**
AG IMA, Dept. of Computer Science
University of Hamburg, Vogt-Kölln-Str. 30
22527 Hamburg
Germany
<surname>@informatik.uni-hamburg.de http://ima-www.informatik.uni-hamburg.de

ABSTRACT

In active vision systems, attentional control is used to determine the relevant parts of a scene and to direct perception towards these parts. To test and evaluate active vision systems, we have implemented a 3D simulation framework capable of simulating a broad scope of environments from simple block worlds to complex photorealistic scenes. The simulator allows full control of all aspects of the simulation, acting and moving inside virtual environments. In this paper, we demonstrate its use for evaluating our attentional control system. The attention model is based on a novel two-stage selection mechanism and especially focuses on the dynamic and three-dimensional aspects of its environment.

Keywords: active vision, virtual reality, attentional control, 3D simulation.

1 INTRODUCTION

By selecting relevant parts of the available input data, attention serves different purposes in active vision systems. A simple reduction of the data to be computed is the main goal to be achieved. Often, attention is used to serialize complex operations, like object recognition, so they may be applied only to one object after another. Another important function is the removal of distracting information belonging to neighboring objects in order to facilitate the perception of a single target. Due to the impressive performance of the human visual system, modelling of attentional control is heavily influenced by natural visual attention research. While we will not discuss this aspect here, it was relevant in the design of our model and an extended discussion can be found in [Backe01].

In order to be used in active vision systems, a model of attentional control has to cope with dynamic environments, moving objects and occlusions. We will focus on the dynamic aspect of assigning attention towards moving objects in the following discussion. Another important problem is the representation of the world inside the attentional system. Our model uses a three-dimensional representation of saliency to appropriately reflect the properties of the environment.

Because evaluation by subjective judification of example performance seems nonsatisfying, we chose to employ a simulation environment capable of providing us with a range of environments of variable qualities from simple block worlds up to complex photorealistic environments. As no existing simulator framework met our demands, we decided to use the simulation framework Orbital 3D[2], currently being developed at the IMA lab [Bunge01]. This framework allows us to use virtual sensors and actuators in environments of scalable quality and complexity with controllable parameters for systematic experimenting.

The rest of the paper is organized as follows. Section 2 gives a short overview of models of visual

[1]Support of the Deutsche Forschungsgemeinschaft is appreciated.

[2]We thank Andreas Baudry and Michael Bungenstock for providing us with helpful hints, discussions, and implementational assistance on the simulator framework Orbital 3D.

attention and of approaches to simulating environments for active vision systems. A description of our system architecture is given in section 3, while the simulation framework is introduced in section 4. Experimental results (section 5) and a conclusion complete the paper.

2 RELATED WORK

2.1 Models of attentional control

Many aspects present in current models of attentional control can already be found in the groundbreaking work of Koch and Ullman [Koch85], which was influenced by research on human visual attention. It consisted of a parallel computation of feature maps indicating the presence of a particular feature at each location. These maps were integrated into a master map of attention, describing the saliency of each location, and for which a WTA-process determined the location of the focus of attention (FOA). After attention was deployed to this location, it was stored in a so-called inhibition map which inhibited the master map of attention which allowed the focus to be moved to another place. The group at Caltech is still improving the orginal Koch and Ullman model by examining methods of integrating different features [Itti01] and including object recognition methods [Miau01]. The group around Eklundh developed models of visual attention for active vision systems using depth information [Maki96] and integrating depth and motion to form a target mask [Maki00]. Kopecz [Kopec96] used dynamic neural fields to integrate selection and tracking for a visual attention system.

2.2 Simulation environments

While in other domains using a simulator for providing virtual 3D environments is a common task, not much work has been devoted to the design and use of simulators for active vision systems and mobile robots. Most systems used in the context of robotic applications [Act01, Miche96, Konol97, Balch00] only employ a two-dimensional map of the scene, which obviously does not meet our requirements. In [Matsu99] the view of a mobile robot is simulated using specialized dedicated hardware not available to us. The quality of the generated views cannot be scaled up for more complex lighting models. The simulator introduced in [Lu00] provides some relevant features, but is also not usable in our environment.

3 ATTENTIONAL CONTROL

Our model differs from the ones described in section 2 in a number of notable ways. First, we abandon the all-to-one selection scheme. Instead we introduce a first selection stage responsible for selecting a small number of salient items (all-to-some). These items are selected for different computations, among which are tracking and collecting feature, location and motion information. The second selection stage, providing a classical single focus of attention, operates on the result of the first stage (some-to-one). The contents of the focus of attention are subject to high-level-computation like object recognition.

The decision for two distinct selection stages is based on the need to track all salient objects simultanously to achieve an effective inhibition of moving targets. When serializing high-level operations, tracking is also necessary for binding extracted information to an object instead of an outdated location. The tracking mechanism in our model is integrated with the selection mechanism and based on dynamic neural fields. The representation of saliency in three dimensions is another difference from other models using two-dimensional maps for saliency values.

3.1 Local saliency computation

As in other visual attention models, our model used various features for computing local measures of saliency as the basis for the deployment of attention. These features should correspond to relevant aspects of the scene, require little knowledge of the environment, and be as diverse as possible to achieve robust performance while at the same time reusing computations in order to increase time efficiency. For special implementations of the system with a known environment or a given task, special adapted feature computations should be added like face detectors or motion templates.

Symmetry Motivated by experiments on spontaneous human fixations [Kaufm69], as well as the symmetric structure of many artificial objects, we have developed a symmetry computation based on the results of gabor-filtering the image. For each image location and different radii, the symmetry is computed as the maximum across different radii of the sum of gabor filter responses perpendicular to a circle around the location with the specified radius.

Eccentricity Complementary to the edge-based symmetry-feature, eccentricity is an area-based feature based on a gray-level segmentation of the image. The segmentation is based on sobel-filtering the image followed by a region-growing process, after which dilation and merging is applied. The eccentricity and orientation is then determined for each segment. The local saliency corresponds to the eccentricity of the segment.

Color contrast Due to the meaningfulness of color in human-made environments as well as in nature, we implemented a color segmentation in the psychophysical MTM color space as the base of our color contrast feature. The contrast of each segment to its neighbors is computed and weighted by the borderlength. The result is taken as the measure of saliency.

Depth from stereo The spatial arrangement of objects is important for attention as well. We compute the depth of objects using stereo information gathered from two cameras. Gabor-filtering applied to both images, from which only the vertical or near-vertical orientations are used. After searching for multiple correspondences, a self-organization process is applied to improve the quality of the initial disparity data. Additionally, we compute confidence values for the computed disparities. For details of the process used, the reader is referred to [Liede98]. The saliency for the depth values is proportional to the disparity thus closer objects achieve higher saliency values because the interaction with the viewer is more immediate.

Saliency representation The saliency computations for the different features now have to be integrated. We have to take into account that we intend to use spatial information from all three dimensions, so we choose a three-dimensional map as the representation with a coarse resolution along the third dimension. After superimposing the values of the feature maps, we distribute them along the third dimension according to the data derived during the computation of the depth feature. Using the confidence of the depth data, we distribute the saliency values using a normal distribution along the third dimension, with the disparity value as the mean and the standard deviation inversely proportional to the confidence.

Figure 1 depicts the model of local saliency computation together with the first selection stage.

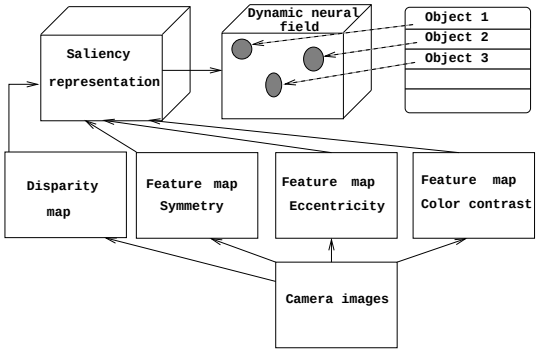

Figure 1: First selection stage and local saliency computation.

3.2 First selection stage

The task of the first selection stage is to integrate the saliency measure spatially and temporally and to extract a small number of objects with high saliency. As was mentioned earlier, to bridge the gap between the selection of locations and objects in a time-varying environment, we have to track the selected areas of high saliency. It is this first selection stage that is responsible for transferring the representation from the subsymbolic to the symbolic level to allow easier representation and manipulation at the subsequent levels.

Neural fields for tracking and selection In order to make the model as simple as possible, we decided to integrate the selection and tracking aspects of the first selection stage into a system of dynamic neural fields (DNF) [Amari77]. The properties of neural fields, especially conditioned maximum selection, hysteresis, integration of information over time and tracking, have been used for different purposes, one of which was modelling attention [Kopec96].

Dynamic neural fields are recurrent networks of neurons, whose connections are of a local excitatory and global inhibitory kind. Different parametrizations of the fields lead to different behaviours. Most notably, one has to distinguish between a global inhibition type (stable states show at most one cluster of activity) and local inhibition type (at sufficient distance, more than one cluster of activity is allowed). In contrast to [Kopec96] we are interested in selecting and tracking multiple objects, so we chose a local inhibition type. The dynamics of neural fields can be described by:

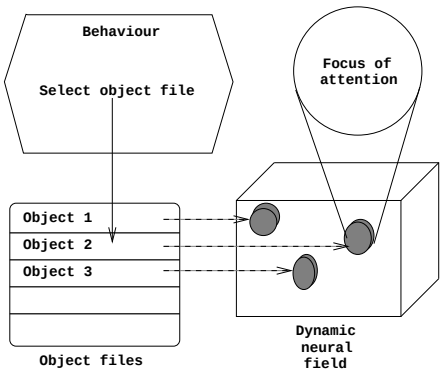

Figure 2: Second selection stage.

$$\tau \frac{d}{dt} u(x,t) = \\ -u(x,t) + h + \int w(x-x') S[u(x,t)] d^2x' + i(x,t)\,.$$

, where u denotes the activity of the neuron at location x and time t, τ is a time constant, S a sigmoid function, h the (negative) resting value, w gives the weights and i the input into the neural field. Using the saliency representation as the input for the neural fields, after a few cycles of updating the fields, we get a small number of activity clusters at the most salient locations. These are the items selected by the first selection stage.

Object files for collecting information For each item selected by the neural fields, we create a so-called object file where information about this object is collected. It contains the timestamped locations of the object as well as information about the presence of the different features at this location. For each new input frame the information is updated and the correspondence of the activity clusters to the object files is reestablished. Every time high-level computations are carried out for an item their results are also stored in the object file. This collection of object files describes the world model of our system. We distinguish between active object files, those who link to an existing activity cluster, and passive object files without such a link.

3.3 Second selection stage

The second selection stage takes the result of the first stage as input and selects one of the items for the single focus of attention. This stage is influenced by top-down information on the current system-state, i. e. the goal to achieve. We have implemented different behaviours for this stage. In this paper we will focus on the "Explore"-behaviour, intended to collect information about an unknown environment. We have described additional behaviours in [Backe01]. Due to the simple representation, it is easy to adapt the system to more specialized behaviours as well. Fig. 2 shows an overview of this second selection stage.

The "Explore"-behaviour selects the most salient item from the active object files as the first FOA. Attention is directed towards this object so that high-level computations can be applied. The high-level computations themselves are not part of attentional control. Its selection time is stored in the object file. The next selection occurs, when the high-level computation for this object is finished. From this point on, the next-to-be-focussed item is selected according to the following priorities: as long as unrecognized active object files exist, one of these will be selected. If, however, all active object files contain identity information, the one with the least recently computed information is selected for an update of the identity. In case the selection is not unique, the amount of saliency is used to define the priority among the items. Inhibition of return in this model is implicit in the behaviour and bound to the object file and therefore to the (dynamic) object instead of a (static) location. The resulting behaviour can be depicted as a scanpath.

Gaze control Thus far, the system only shows covert attention - attention by internal selection and assignment of computational resources. In the context of active vision systems, our interest is in overt attention as well - attention by sensor manipulation, especially movement of cameras, in order to fixate a selected target. While only covert attention was described before, our model covers overt attention as well. The selected behaviour is responsible for triggering a gaze shift toward the focus of attention. The neural field activity and the positions stored in the object files are adjusted to match the modified view. The neural field activation is moved in accordance to the gaze shift and areas entering the view are initialized with the resting value of the field. Object files relating to a position outside the current view are deactivated. They will be checked for a match whenever a new object file is created.

4 SIMULATION FRAMEWORK

4.1 Design of Orbital 3D

Evaluation of attentional control is not the primary purpose of our simulation framework. It is intended to generally simulate different kinds of sensors and actuators, e.g. cameras, pan-tilt-units, or mobile robots, in a given environment. Research and teaching on active vision systems and mobile robots is carried out using the simulator. This resulted in the need for a component-oriented framework with plugins for the sensors and actuators, allowing easy configuration and modification of the simulated environment and the devices used inside these environments.

To allow for maximal portability, the simulation framework Orbital 3D was implemented in Java and was tested on Linux, Windows and Solaris platforms. In Orbital 3D the simulated world is described by configuration files in XML; graphical objects can be imported from POVRay and Java 3D-files. Sensor models can be added as plugins to provide a wide range of lighting models. It is up to the user to choose the appropriate tradeoff between quality and computational complexity (times range from 0.3 seconds on a OpenGL-supported platform and Java-3D up to 10 seconds for a high quality POVRay image on a standard PC for pictures of 512 by 512 pixels). Communication with the simulator is done via HTTP so that the simulator and the program using it can run on different machines. A C++-library translates the requests and commands from a calling application into HTTP-requests. This way, we were able to build interfaces most similar to those of our existing technical systems and easily switch between these systems and the simulator with only slight modification of the external application.

An overview of the architecture is depicted in fig. 3. For a more detailed description and a different test scenario see [Bunge01].

4.2 Using Orbital 3D for evaluation of attentional control

In order to evaluate our model of attentional control, we use Orbital 3D with environments of different complexities. To easily identify the effects of environment properties, such as color, depth or object size, we use a simple kind of block world with a small number of different objects. The objects can be described by a small number of parameters.

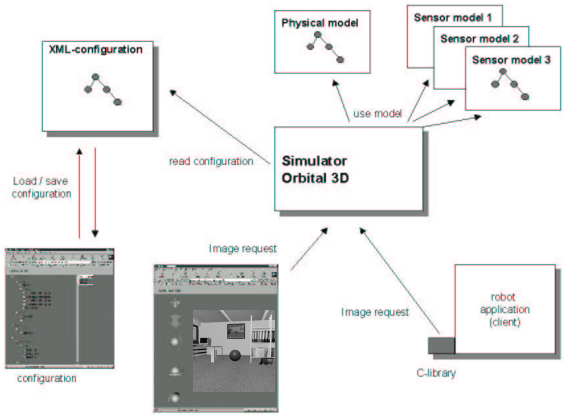

Figure 3: Overview of the simulation framework.

By modifying these parameters we can identify the relationship between properties of the environment and the system performance. Our special interest lies in examing the saliency of the objects and their resulting position in the scanpaths.

The scalability of the simulator gives us the additional benefit of being able to use more complex, photorealistic environments to explore the behaviour of our attentional control system in realistic environments. Here, we use a simulation of our laboratory. Later we intend to compare the results to those achieved by real-world systems like our stereo camera-head or our Pioneer II mobile robot equipped with a stereo camera-head.

5 RESULTS

5.1 Feature computation

Fig. 4 shows a simple block world scene and the results of the feature computation. The three objects are easily recognizable in the different feature maps, the elongated object received the highest activation for eccentricity, the ball in front the highest activation for symmetry and depth and the colored cube for color contrast.

For a closer examination of the feature computation we used the simulator to vary some parameters of the scene while the parameters of the feature computation process were held constant. We find the expected relation between saliency and the following modifications of the properties of the block world: the eccentricity of the elongated object, the color contrast of the cube and the depth of the ball (see fig. 5).

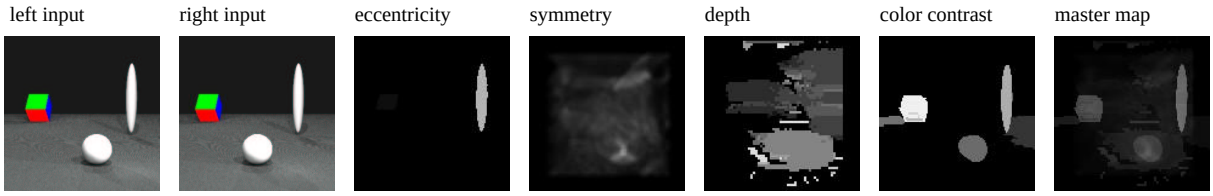

Figure 4: Feature computations for a simple block world scene. Lighter values denote higher saliency.

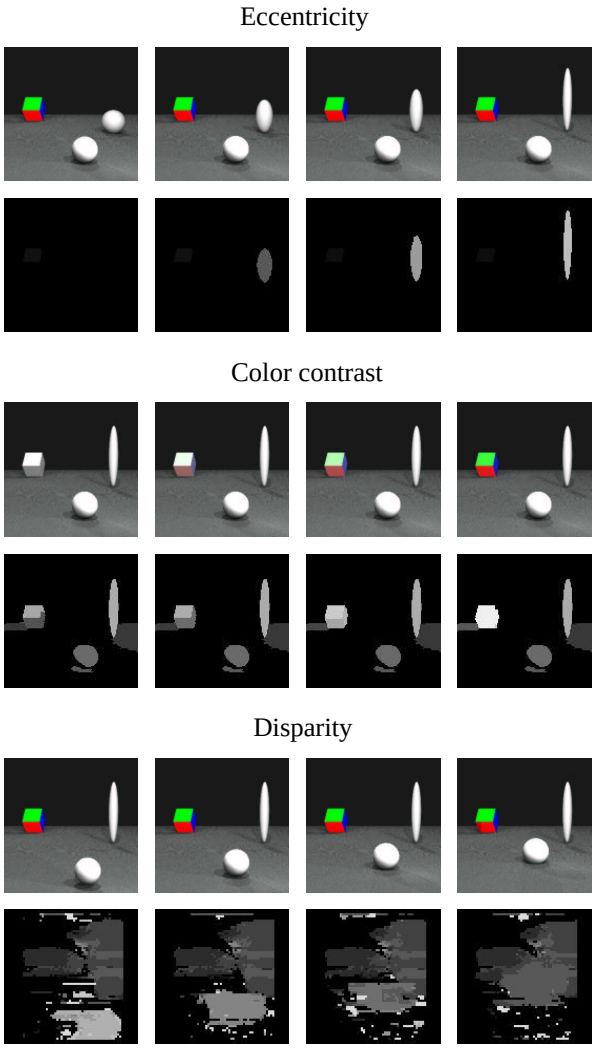

Figure 5: Feature computations for different parametrizations of the simple block world scene. Lighter values denote higher saliency. Only the modified feature is examined.

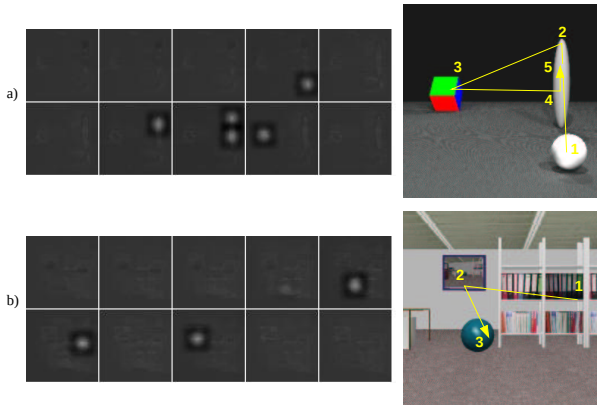

Figure 6: Neural field activity (left row) and scanpaths for a) a simple block world and b) a laboratory scene using the Orbital 3D simulator. The depth slices of the neural fields are arranged according to rising depth in reading order. Lighter values denote higher activity.

5.2 Scanpaths for static input

To examine the behaviour of the complete system, we used a photorealistic scene representing our laboratory. The neural field activity after convergence for the different static inputs is depicted in figure 6. Note the selection of the most salient objects and their localization in three spatial dimensions. Using this neural field activity, a scanpath is generated without sensor movement. The length of the scanpath is determined by the parameters of the dynamic neural fields and the characteristics of the input scene. For the first input, five distinct activity clusters evolved, while for the second input there were three activity clusters. Each of them is visited one time before the scan cycle is repeated. The sequence is determined using the accumulated activity in the neural field.

In figure 7 we used the same input modification as for the block world and examined the consequences for the scanpath. We expected to find that objects with a reduced saliency would be scanned

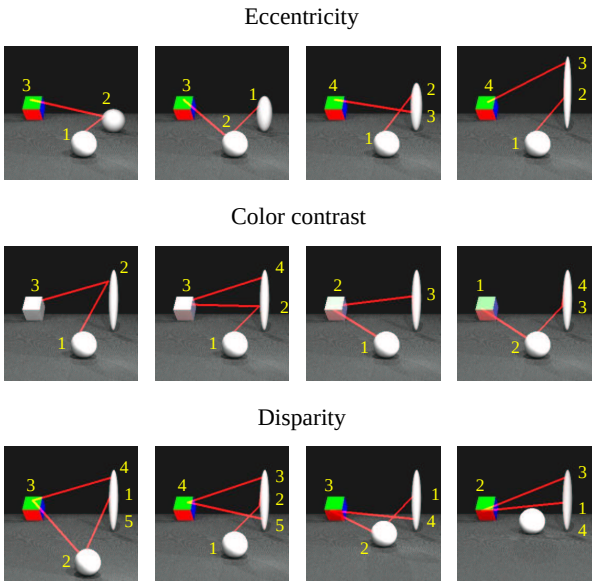

Figure 7: Scanpaths for modifications of a simple block world.

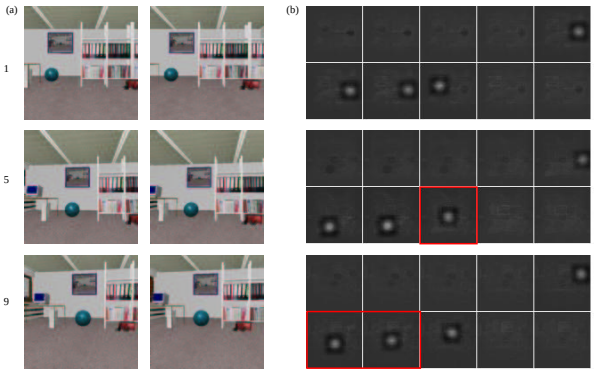

Figure 8: System behaviour for a number of selected frames. From left to right, the left and right input frames (activities are related to the left input frames) and neural field activities are shown. Every four frames, a gaze shift is executed. The depth slices of the neural field containing the selected activity cluster are highlighted.

later. Although this is the case most times, there are some notable exceptions. In the first row, the eccentricity of the right object is increased, resulting in its earlier selection in the second picture. In the following picture, the size of the object increases, so that multiple object files are created for this single object. These object files each have a reduced saliency and, accordingly, are scanned later. They would be scanned earlier, if an integration of saliency for the complete object were taken into account. The results from these experiments pointed us towards the need of integrating object files belonging to the same object, which we will implement in the future. Raising the color contrast of the left object in the second row shows the expected results.

5.3 Overall system behaviour for dynamic input

Using a more complex dynamic input scene with moving objects, the behaviour of our attentional control , including sensor movement, is shown in figure 8. The activity clusters closely follow the moving objects selected by the dynamic neural fields. In this simulation, we assume that the high-level computation takes four frames, after which a saccade is performed to the next object file position chosen by the "Explore"-behaviour for a closer examination of the object. After the initial gaze direction in frame 1, frame 5 shows a fixation of the picture at the wall and the last frame a fixation of the ball rolling through the room.

6 CONCLUSIONS AND OUTLOOK

In this paper, we presented a novel model of visual attention, which employs a dual-stage selection mechanism. The model appropriately reflects the more advanced properties of its environment like multiple moving objects, depth and occlusion. The decision for a two-stage selection architecture allowed us to apply computations of different complexities either to multiple salient objects or to the single focus of attention. The dynamics of the neural field model efficiently integrated selection and tracking of multiple items, which is not implemented in standard models of visual attention. For the selected items, we used a symbolic representation which would allow different behaviours to be applied. These behaviours describe the selection of a classical focus of attention for high-level computations like object recognition and the control of gaze.

The simulator Orbital 3D allowed us to carry out different experiments in order to evaluate this model. While most of the experiments proved the strengths of the model, some pointed us towards necessary additions and modifications.

To sum up, the system presented represents an efficient module for narrowing the stream of visual data down to the relevant subset to be used in an

active vision system. The simulator demonstrated the usefulness and apropriateness of our approach. In the future the system will be extended to make use of additional more complex behaviours as well as other features like motion. These extensions will be evaluated in additional experiments and scenarios using Orbital 3D. We hope to be able to test concurrent approaches to attentional control using this simulation framework.

REFERENCES

[Act01] Activemedia robotics, basic suite. http://www.amigobot.com/, 2001.

[Amari77] S.-I. Amari. Dynamics of pattern formation in lateral inhibition type neural field. *Biological Cybernetic*, 27:77–87, 1977.

[Backe01] Gerriet Backer, Bärbel Mertsching, and Maik Bollmann. Data- and model-driven gaze control for an active-vision system. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(12):1415–1429, 2001.

[Balch00] T. Balch. Cmu's multirobotlab, teambots simulator. http://www.teambots.org/, 2000.

[Bunge01] M. Bungenstock, A. Baudry, J. Bitterling, and B. Mertsching. Development of a simulation framework for mobile robots. In *Proceedings of the EUROIMAGE ICAV3D 2001*, pages 89–92, 2001.

[Itti01] L. Itti and C. Koch. Feature combination strategies for saliency-based visual attention systems. *Journal of Electronic Imaging*, 10(1), 2001.

[Kaufm69] L. Kaufman and W. Richards. Spontaneous fixation tendencies for visual forms. *Perception and Psychophysics*, 5(2):85–88, 1969.

[Koch85] C. Koch and S. Ullman. Shifts in selective visual attention: Towards the underlying neural circuitry. *Human Neurobiology*, 4:219–227, 1985.

[Konol97] K. Konolige. *Saphira Software Manual Version 6.1.* ActiveMedia, http://www.ai.sri.com/ konolige/saphira/, 1997.

[Kopec96] K. Kopecz. Neural field dynamics provide robust control of attentional resources. In B. Mertsching, editor, *Aktives Sehen in technischen und natürlichen Systemen*, pages 137–144, 1996.

[Liede98] T. Lieder, B. Mertsching, and S. Schmalz. Using depth information for invariant object recognition. In S. Posch and H. Ritter, editors, *Dynamische Perzeption*, pages 9–16. St. Augustin (Infix), 1998.

[Lu00] J.Z. Lu and M. Xie. Simulation of vision-guided vehicle. In *Sixth International Conference on Control, Automation, Robotics and Vision, Singapore*, 2000.

[Maki96] A. Maki, P. Nordlund, and J.-O. Eklundh. A computational model of depth-based attention. In *Proc. 13th Int. Conf. on Pattern Recognition*, volume 4, pages 734–738, 1996.

[Maki00] A. Maki, P. Nordlund, and J.-O. Eklundh. Attentional scene segmentation: Integrating depth and motion. *Computer Vision and Image Understanding*, 78:351–373, 2000.

[Matsu99] Y. Matsumoto, T. Miyazaki, M. Inaba, and H. Inoue. View simulation system : A mobile robot simulator using vr technology. In *Proceedings of 1999 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'99)*, pages 936–941, Kyongju, Korea, 1999.

[Miau01] F. Miau and L. Itti. A neural model combining attentional orienting to object recognition: Preliminary explorations on the interplay between where and what, in press. In *Proceedings IEEE Engineering in Medicine and Biology Society (EMBS)*, 2001.

[Miche96] O. Michel. *Khephera Simulator 2.0 - User Manual*, 1996.