Gaze Control in a Multiple-Task Active-Vision System

Daniel Hernandez, Jorge Cabrera, Angel Naranjo, Antonio Dominguez, and Josep Isern

IUSIANI, ULPGC, Spain
dhernandez,jcabrera,adominguez@iusiani.ulpgc.es *

Abstract. Very little attention has been devoted to the problem of modular composition of vision capabilities in perception-action systems. While new algorithms and techniques have paved the way for important developments, the majority of vision systems are still designed and integrated in a very primitive way according to modern software engineering principles. This paper describes the architecture of an active vision system that has been conceived to ease the concurrent utilization of the system by several visual tasks. We describe in detail the functional architecture of the system and provide several solutions to the problem of sharing the visual attention when several visual tasks need to be interleaved. The system's design hides this complexity to client processes that can be designed as if they were exclusive users of the visual system. Some preliminary results on a real robotic platform are also provided.

1 Introduction

The control of the gaze in an active vision system is usually formulated as a problem of detection of significant points in the image. Under this perspective several aspects such as saliency, bottom-up vs. top-down control, computational modelling, etc, have been analyzed [1][2]. An alternative view considers the shared resource nature of the sensor, transforming the scenario into a management/coordination problem where the control of the gaze must be shared among a set of dynamic concurrent tasks.

In close relation with the aforementioned, but from a more engineering point of view, is the fact that researchers and engineers involved in the development of vision systems have been primarily concerned with the visual capabilities of the system in terms of performance, reliability, knowledge integration, etc. However very little attention has been devoted to the problem of modular composition of vision capabilities in perception-action systems. While new algorithms and techniques add new capabilities and pave the way for new and important challenges, the majority of vision systems are still designed and integrated in a very primitive way according to modern software engineering principles. There are few

* This work has been partially supported by Spanish Education Ministry and FEDER (project TIN2004-07087) and Canary Islands Government (project PI2003/160)

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

published results on how to control the visual attention of the system among several tasks that execute concurrently [3].

Posing a simple analogy to clarify these issues, when we execute programs that read/write files that are kept in the hard disk, we aren't normally aware of any contention problem and need not to care if any other process is accessing the same disk at the same time. This is managed by the underlying services, simplifying the writing of programs that can be more easily codified as if they have exclusive access to the device. Putting it on more general terms, we could consider this feature as an "enabling technology" as it eases the development of much more complex programs that build on top of this and other features.

How many of these ideas are currently applied when designing vision systems?. In our opinion, not many. Maybe because designing vision systems has been considered mainly as a research endeavor, much more concerned with other higher level questions, these low level issues tend to be simply ignored. If we translate the former example to the context of vision systems, some important drawbacks can be easily identified.

- Vision systems tend to be monolithic developments. If several visual tasks need to execute concurrently this need to be anticipated since the design stage. If some type of resource arbitration is necessary it is embedded in the code of the related tasks.
- Within this approach it is very difficult to add new visual capabilities that may compete against other tasks for the attention of the system.
- As far as contention situations are dealt with internally and treated by means of ad-hoc solutions, the development of such systems does not produce any reusable technology for coping with these problems.

As a selection of related work, the following three systems can be mentioned. Dickmanns and colleagues [4] have studied the problem of gaze control in the context of their MarVEye Project, where an active multi-camera head is used to drive a car in a highway. In that project, several areas of interest are promoted and ranked by different modules of the control architecture using a measure of information gain. The gaze controller tries to design the gaze trajectory that within 2 seconds can provide the highest gain of information to the system. Several areas of attention can be chained in a gaze trajectory, though in practice this number is never higher than two.

Humanoid robots need to rely in their visual capabilities to perceive the world. Even for simple navigation tasks, a number of tasks (obstacle avoidance, localization, ...) need to operate concurrently to provide the robot with minimal navigation capabilities. Several groups have explored the problem of gaze arbitration in this scenario, both in simulation and with real robots. Seara et al. [5] [6] have experimented with a biped robot that used a combination of two tasks to visually avoid obstacles and localize itself. The decision of where to look next was solved in two stages. Firstly, each task selects its next preferred focus of attention as that providing the largest reduction of incertitude in the robot localization, or in the location of obstacles. In a second stage, a multiagent decision schema, along with a winner-selection society model, was used to finally

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

decide which task was granted the control of gaze. Of the several society models that were evaluated, the best results, as judged by the authors, were obtained by a society that tries to minimize the total "collective unhappiness". Here the concept of unhappiness is derived of loosing the opportunity of reducing incertitude (in self or obstacle localization).

Sprague et al. [7][3] have designed a simulation environment where a biped robot must walk a lane while it picks up litter and avoids obstacles, using vision as the only sensor. These capabilities are implemented as visual behaviors using a reinforcement learning method for discovering the optimal gaze control policy for each task. The lateral position of the robot within the lane and the position of obstacles and litter are modelled by Kalman filters. Every 300 msec, the gaze is given to the task that provides the largest gain in uncertainty reduction.

Similar in spirit to this related research, the motivation of our work is consequently two-fold: contribute to build a vision system more consistent from an engineering point of view, and to take a first step towards systems where the vision becomes integrated in an action context with higher semantic and cognitive level (an "intelligent" way of looking).

In the next sections we will present the proposed architecture, with its design objectives and main components. Some experimental results obtained on a real robot, along with conclusions and intended future development will be described in the last two sections.

2 MTVS

Motivated by the previously stated description of the problem we have designed and implemented MTVS (Multi-Tasking Vision System), a proposal of architecture for active-vision systems in multi-tasking environments. MTVS has been designed to deal with the scheduling of concurrent visual tasks in such a way that resource arbitration is hidden to the user.

2.1 Objectives

More in detail, MTVS pursues the following objectives:

- The assignment of the gaze control to a task is based on a simple scheduler model, so that the behavior can be easily interpreted by an external observer.
- The client tasks are integrated in the system individually with no coordination requirements.
- The set of tasks managed (the activity) can change dynamically.
- The clients should not assume any a priori response time guarantee, though the system can offer high-priority attention modes.
- The system offers services based on a reduced set of visual primitives, in pre-categorical terms.

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

2.2 Internal architecture

The figure 1 shows an example with two clients and the basic elements making up the vision system architecture: a system server, a task scheduler and the data acquisition subsystem. Basically, the clients connect to the system server to ask for visual services with a given configuration (client A active). In response, the system server launches both a task-thread, to deal with internal scheduling issues, and a devoted server-thread that will be in charge of the interaction with the external client. The scheduler analyzes the tasks demands under a given scheduling policy and selects one to receive the gaze control. In combination, a second covert scheduler checks for compatibility between tasks to share images among them (FOA's overlapping). The data acquisition subsystem processes the different sensor data streams (images, head pose and robot pose) to generate as accurately as possible time stamps and pose labels for the served images.

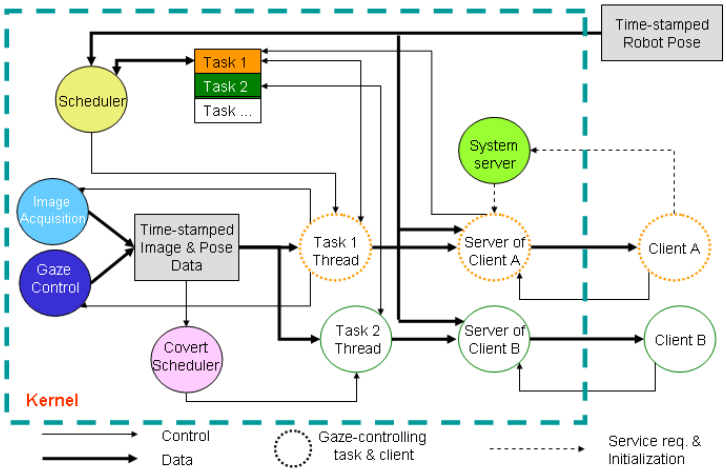

Fig. 1. Control Architecture: example with two clients

2.3 Visual Services

Clients can connect to the vision system and use it through a number of pre-categorical low-level services. The MTVS services are built around basic visual capabilities or primitives that have also been explored by other authors [8]:

WATCH: Capture N images of a 3D point with a given camera configuration.
SCAN: Take N images while the head is moving along a trajectory.

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

SEARCH: Detect a model pre-categorically in a given image area.
TRACK: Track a model pre-categorically.
NOTIFY: Inform the client about movement, color or other changes.

Except for WATCH, the rest of primitives can be executed discontinuously, allowing for the implementation of interruptible visual tasks.

The clients also regulate their activity in the system by means of the messages they interchange with their devoted server. Currently, the following messages have been defined for a task: creation, suspension, reconfiguration (modify parameters, change priority, commute primitive on success) and annihilation.

2.4 The scheduler

Several scheduling policies have been implemented and studied inside MTVS. This analysis has considered two main groups of schedulers: time-based and urgency based schedulers.

Time-based schedulers Three types of time-based schedulers have been studied: Round-Robin (RR), Earliest Deadline First (EDF) and EDF with priorities (EDFP). The prioritized RR algorithm revealed rapidly as useless in a dynamic and contextual action schema. First, it makes no sense to assign similar time slices to different tasks, and second, the time assigned used for saccadic movements, specially when a slow neck is involved, becomes wasted.

The EDF algorithm yielded a slightly better performance than RR, but was difficult to generalize as visual tasks are not suitable for being modelled as periodic tasks. The best results of this group were obtained by the EDFP algorithm combining critical tasks (strict deadline) with non-critical tasks. Each time a task is considered for execution and not selected its priority is incremented by a certain quantity [9].

Urgency-based schedulers The concept of urgency is well correlated with a criteria of loss minimization, as a consequence of the task not receiving the control of the gaze within a time window. This measure can also be put into relation with uncertainty in many visual tasks.

Two schedulers have been studied in this group: lottery [7] and max-urgency. The lottery scheduler is based in a randomized scheme where the probability of a task being selected to obtain the gaze control is directly proportional to its urgency. Every tasks has the possibility of gaining the control of the gaze, but the random unpredictability can sometimes produce undesirable effects.

The max-urgency scheduler substitutes the weighted voting by a direct selection of the task with higher urgency value. This scheme has produced acceptable results provided that the urgency of a task is reduced significantly after gaining the control of the gaze (similar to an inhibition of return mechanism).

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

3 Implementation

The Vision System can be operated in a number of configurations. In its most simple configuration the system may comprise a simple pan/tilt system and a camera, or a motorized camera, or it can integrate both systems as illustrated in the experiments described later. In this last configuration, a relatively slow pan/tilt system, acting as the neck of the system, carries a motorized camera (SONY EVIG21) equipped with a fast pan/tilt system, that can be considered as the eye of the system. This mechanical system in turn can be used in isolation or it can be installed on a mobile robot.

The vision system runs under the Linux OS as a multithreaded process. Clients run as independent processes. They may share the same host, in which case the communication primitives are implemented using shared memory, or on different hosts connected by a local network. The interface to the image acquisition hardware is based on the Video4Linux2 services so that a large number of cameras and/or digitizers can be supported. For the image processing required by the primitives of the system or at the clients, the OpenCV library is used.

4 Experiments

A set of experiments were carried out to analyze the behavior of MTVS on a real robotic application. The basic experimental setup consists of two ActivMedia Pioneer robots, one with the basic configuration and the other mounting an active vision system formed by a Directed Perception PTU (neck) and a motorized Sony EVI-G21 camera (eye).

Two main tasks were combined along the different experiments: target following and obstacle avoidance. The target following task commands the active vision robot (pursuer) to detect and follow a special square target mounted on other robot (leader), trying to keep a predefined constant distance between them. The obstacle avoidance task looks for colored cylinders on the floor, estimating, as exactly as possible their 2D position. Kalman filtering is used to model both target and obstacles positions.

4.1 One-task-only experiments

As a reference for the maximum expected performance for each task some experiments where designed involving only one task.

Experiment 1: Follow Robot only In this experiment, the leader robot is commanded to move forward at a constant speed of 200 mm/sec, while the pursuer must try to keep a constant separation of 2 meters. Several tests have been conducted along the main corridor of our lab following a 15 meters straight line path. The pursuer was able to stabilize the reference distance with a maximum error around 150 mm.

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

Experiment 2: Obstacle avoidance only Now the active vision robot is commanded to explore the environment looking for objects (yellow cylinders), trying to reduce their position uncertainty below a predefined threshold. The robot moves straight-line inside a corridor formed by 8 cylinders equally distributed in a zigzag pattern along the path. The figure 2 illustrates the robot path and the different detections for each localized object, including their first (larger) and minimum uncertainty ellipses. The results show how the robot was able to localize all the objects with minimum uncertainty ellipses ranging from 100 to 200 mm in diameter.

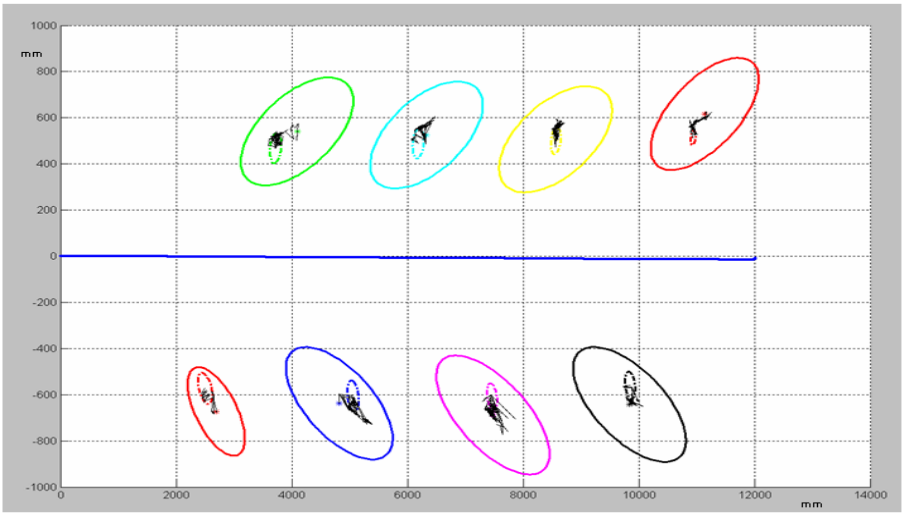

Fig. 2. Obstacle avoidance-Only experiment

4.2 Multiple-task experiments

The multiple-task experiments consider an scenario in which each task computes its desired camera configuration and urgency and asks the MTVS scheduler to obtain the gaze control. The scheduler uses this information to select where to look next and how to distribute images. The obstacle avoidance task is extended to classify special configurations of objects as "doors" (two objects aligned perpendicularly to robot initial orientation with a pre-defined separation).

The urgency of the following task is computed as a function of the distance error, the robot velocity and the time. This urgency increases as the distance between the robots differs from the reference, the velocity is high and the elapsed time since the last image was received becomes larger.

The urgency of the obstacle avoidance task is computed separately for three possible focus of attention: front (the urgency increases when the robot moves to-

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

wards visually unexplored areas), worst estimated object (the urgency increases as the position of a previously detected object is not confirmed with new images), and closest door (the urgency increases with narrow doors).

The first two simple multiple-task experiments try to illustrate the sharing images capability of MTVS. In each case one task has higher priority, so in a context of exclusive camera sensor access the secondary task shows very low performance. Allowing the sharing of images (comparing FOA's), however, the overall performance can be improved. In experiment 5 a more complex scenario including doors is analyzed.

Experiment 3: Obstacle avoidance and robot following competing for the gaze (following priority) In this experiment, the control of the gaze is only granted to the avoidance task when both the leader speed and the distance error are low. Typically, the following task performance is not affected significantly, but the avoidance task degrades yielding few objects localization with poor precision. As an example, the upper plot of the figure 3 presents the results of a non sharing run where only half the potential objects (all right sided due to the position of the closest obstacle) have been detected with large uncertainty ellipses. As the lower plot of the figure shows, the sharing of images permits a much better behavior of the obstacle avoidance task.

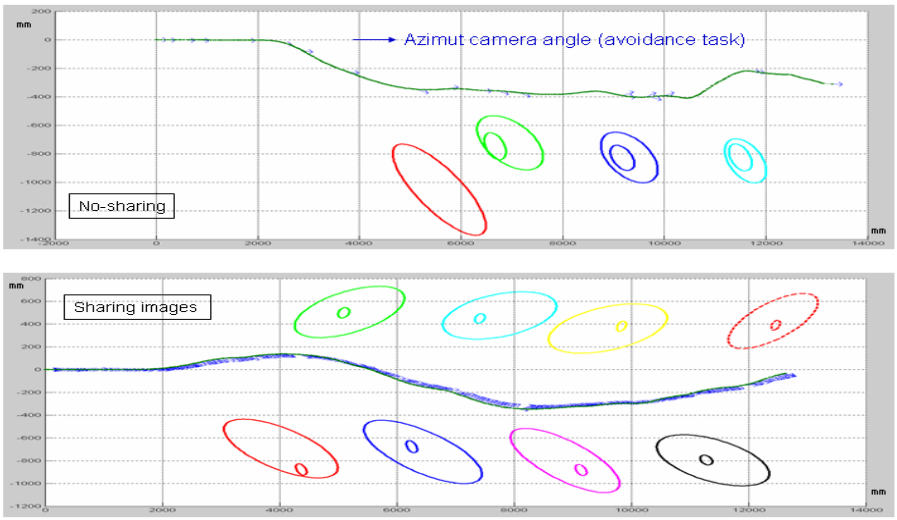

Fig. 3. Follow (priority) and avoidance experiment

Experiment 4: Obstacle avoidance and robot following competing for the gaze (avoidance priority) In this experiment the control of the gaze is

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

only granted to the following task when the closest objects have been localized with precision. When sharing is not allowed, the avoidance task keeps an acceptable performance while the following task fails as the leader robot goes out of visual field with no reaction. When sharing is permitted, the following task behavior improves, but only initially. Soon, the camera must be pointed laterally to reduce the localization error of the objects, and the captured images are no longer valid for the following task, that degrades rapidly.

Experiment 5: Localize doors and robot following competing for the gaze (narrow and wide doors) The configuration of objects used for this experiment consist of a set of four "doors": two narrow type (600 mm width) and two wide type (1500 mm width). All doors are located straight line in front of the robot, the first one (wide) three meters ahead and the rest every 1.5 meters, alternating narrow and wide types. The leader robot is commanded to move at constant speed crossing the doors centered.

The figure 4 illustrates how the camera is pointed to both sides when crossing narrow doors. As a consequence of this behavior, the pursuer robot slows down when approaching a narrow door until the door extremes position have been estimated with the required precision (compare final error ellipses for narrow and wide doors). After traversing the door, the robot accelerates to recover the desired following distance from the leader.

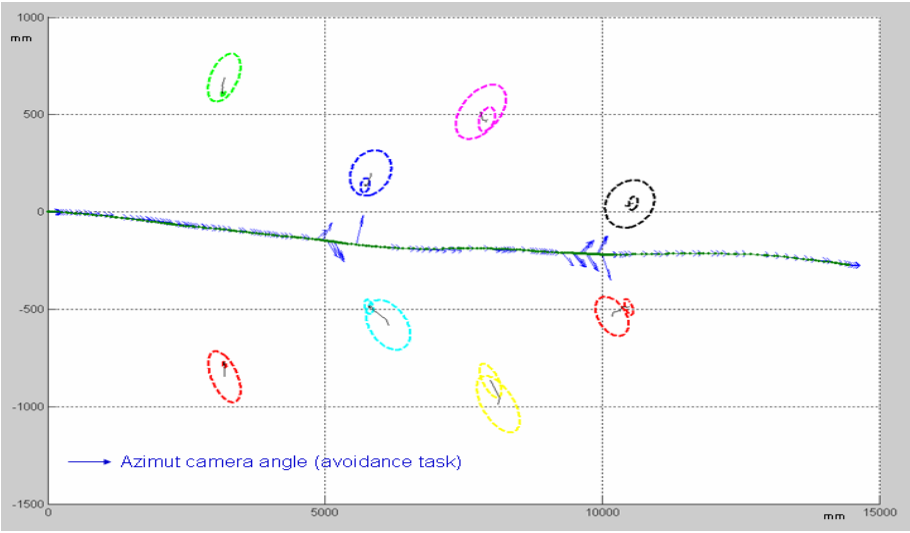

Fig. 4. Narrow and wide doors experiment

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de

5 Conclusions and Future developments

In this paper we propose an open architecture for the integration of concurrent visual tasks. The clients requests are articulated on the basis of a reduced set of services or visual primitives. All the low level control/coordination aspects are hidden to the clients simplifying the programming and allowing for an open and dynamic composition of visual activity from much simpler visual capabilities.

Regarding the gaze control assignation problem, several schedulers have been implemented. The best results are obtained by a contextual scheme governed by urgencies, taking the interaction of the agent with its environment as organization principle instead of temporal frequencies. Usually, a correspondence between urgency and uncertainty about a relevant task element can be established.

The system described in this paper is just a prototype, mainly a proof of concept, and it can be improved in many aspects. The following are just a few of them. We plan to improve the adaptability of the system to different active vision heads (hardware abstraction). A first step is to consider the extension of the system to be applied over a binocular system, where new problems like eye coordination, vergence and accommodation must be tackled. Another issue is the need of an acceptance test for new service requests to avoid overloading the system. Besides, the introduction of homeostasis mechanisms could help to make the system more robust; and the inclusion of bottom-up directed attention (independent movement, e. g.) would allow for new saliency-based primitives.

References

1. Arkin, R., ed.: Behavior-Based Robotics. MIT Press (1998)
2. Itti, L.: Models of bottom-up attention and saliency. In Itti, L., Rees, G., Tsotsos, J.K., eds.: Neurobiology of Attention. Elsevier Academic Press (2005)
3. Sprague, N., Ballard, D., Robinson, A.: Modeling attention with embodied visual behaviors. ACM Transactions on Applied Perception (2005)
4. Pellkoffer, M., Ltzeler, M., Dickmanns, E.: Interaction of perception and gaze control in autonomous vehicles. In: SPIE: Intelligent Robots and Computer Vision XX: Algorithms, Techniques and Active Vision, Newton, USA (2001)
5. F. Seara, J., Lorch, O., Schmidt, G.: Gaze Control for Goal-Oriented Humanoid Walking. In: Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids), Tokio, Japan (November 2001) 187–195
6. F. Seara, J., Strobl, K.H., Martin, E., Schmidt, G.: Task-oriented and Situation-dependent Gaze Control for Vision Guided Autonomous Walking. In: Proceedings of the IEEE/RAS International Conference on Humanoid Robots (Humanoids), Munich and Karlsruhe, Germany (October 2003)
7. Sprague, N., Ballard, D.: Eye movements for reward maximization. In: Advances in Neural Information Processing Systems (Vol. 16). MIT-Press (2003)
8. Christensen, H., Granum, E.: Control of perception. In Crowley, J., Christensen, H., eds.: Vision as Process. Springer-Verlag (1995)
9. Kushleyeva, Y., Salvucci, D.D., Lee, F.J.: Deciding when to switch tasks in time-critical multitasking. Cognitive Systems Research (2005)

Proceedings of the 5th International Conference on Computer Vision Systems (ICVS 2007)
Published in 2007 by Applied Computer Science Group, Bielefeld University, Germany, ISBN 978-3-00-020933-8
This document and other contributions archived and available at: http://biecoll.ub.uni-bielefeld.de