Integrating an Entertainment Robot*

A. C. Domínguez Brito†, J. Cabrera Gámez, D. Hernández Sosa,
M. Castrillón Santana, J. Lorenzo Navarro, J. Isern González,
C. Guerra Artal, I. Pérez Pérez, A. Falcón Martel,
M. Hernández Tejera, J. Méndez Rodríguez.

Centro de Tecnología de los Sistemas y
de la Inteligencia Artificial (CeTSIA)
Universidad de Las Palmas de Gran Canaria - SPAIN

Abstract: *In this paper we will present Eldi, a mobile robot that has been in operation at the Elder Museum of Science and Technology at Las Palmas de Gran Canaria since December 1999. This is an ongoing project that was organized in three different stages of which only the first one has been accomplished. The initial phase, termed "The Player", the second stage, actually under development, has been called "The Cicerone" and in the final phase, termed "The Vagabond", in which Eldi will be allowed to move erratically across the Museum. This paper will focus on the accomplished first stage to succinctly describe the physical robot and the environment and demos developed. Finally we will summarize some important lessons learnt.*

Keywords: mobile robotics, entertainment robotics, agent architectures

1 Introduction

Last years have revealed Education and Entertainment as promising, though demanding, new application scenarios for robotics with a great scientific, economic and social potential [5]. The interest raised by products like Sony's Aibo or the attention deserved by the media to projects as Sage [4], Rhino [3], Kismet [1] and, more recently, Minerva [2] demonstrate the fascination of the general public for these new fashion robotic "pets".

*This work has been partially supported by the UE/DGES research project **1FD1997-1580-C02-02**, and by the research project **PI/1999/153** funded by the Autonomous Government of Canary Islands - Gobierno de Canarias.

†Contact address: Dpto. Informatica y Sistemas, Campus Univ. de Tafira, Las Palmas de Gran Canaria, 35017 Las Palmas, SPAIN. E-mail: acdbrito@dis.ulpgc.es. Fax: +34 928 458711.

In this paper we will present Eldi, a mobile robot that has been in operation at the Elder Museum of Science and Technology at Las Palmas de Gran Canaria since December 1999. This is an ongoing project that was organized in three different stages of which only the first one has been accomplished. The initial phase, termed “The Player”, was devoted to design and build the physical robot, obtain a scalable and extensible software control architecture and put all this into operation in a number of shows and demos that should be offered to visitors. The second stage, actually under development, has been called "The Cicerone" and aimed at adding better navigational capabilities in the robot such as it can give tours through some of the Museum’s halls. In a final phase, termed “The Vagabond”, Eldi will be allowed to move erratically across the Museum during its “spare” time (i.e. while not required to give a tour, attend a show or recharge batteries) and it will be possible for a visitor to demand its attention and services through a multimodal interface (gesture, voice and a touch-screen).

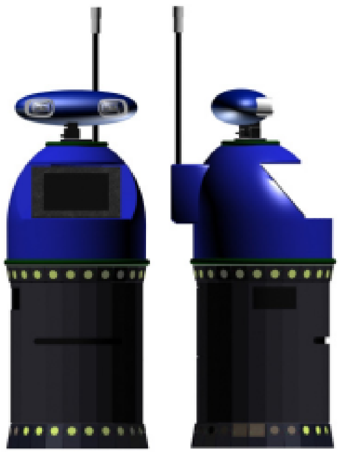

Figure 1. Front and side views of Eldi

2 ELDI system anatomy

2.1 Hardware description

As stated above, the first phase of the project, carried out during 11 months, was devoted to build the robot and accomplish a first level of capabilities. Physically, the robot’s body has two main components, see figure 1. The lower part integrates a commercial Nomadic’s XR4000 mobile platform that gives the robot its basic mobility and sensor capabilities, hosted by a processor under Linux operating system. On top of this platform, it integrates a “torso” that hosts a second processor – TopRobot – under Microsoft Windows NT, several radio communication systems that offer a 802.11 wireless link with off-board systems and transmission of color video and sound from the robot, a touch-screen, loudspeakers and a two degree of freedom head.

The robot is equipped with an active vision system that comprises a pair of Sony EviG21 motorized color cameras housed in the head, a Directed Perception pan-tilt that articulates the neck, and a PCI frame grabber. Basically, the processor installed in the mobile platform controls the motion, localization, obstacle avoidance and power resources of the whole system, it runs under the Linux operating system. The second "upper" system, TopRobot, that runs under Microsoft Windows NT 4.0, controls the whole robot and develops all interactions with users through a number of devices that include the vision system. Communications with off board systems are routed through the Linux system.

2.2 Control Systems

Three main subsystems control the robot. In the upper body a 350 MHz Pentium II takes care of vision, communication, interaction and high-level robot control. In the base, a micro-controller network manages the power and sensor systems (ultrasonars, infrared, bumpers) at low level, and a 233 MMX Pentium under Linux is in charge of platform sensor control, obstacle avoidance, localization, and low level motion control.

Several external machines complete the system – PC's connected using a local area network with two segments being Eldi the network gateway –: The **GameController**, a dual Pentium 350Mhz under Microsoft Windows NT, The **BoardController**, a Pentium 300 under Microsoft Windows 98 and, the **ConacPC**, a Pentium 100 under Microsoft Windows 95.

Global control is achieved by means of CAV, a software architecture that provides a substrate for combining different machines in an asynchronous manner, which can exchange signals with parameters. This architecture is described in more detail later.

2.3 Sensors

In its upper body, the robot incorporates an active color vision system – SONY EVI-G21 and Imaging PCI frame grabber – mounted on a pan-tilt unit by Directed Perception for color detection and tracking – faces, robot games pieces –, and a 14" SVGA color touch-screen by ELO Touchsystems for direct interaction with visitors – information, screen games –. A laser beacon is also included as part of the location system – CONAC –.

In its lower body – an XR4000 from Nomadic Technologies –, there are microswitches for door opening detection and temperature probes for motor overheating control. The robot base has two rings with 24 sensor modules with ultrasonic sensors for long range obstacle detection, infrared sensors for short range obstacle detection and bumpers for contact detection, besides there are additional contact sensors on doors.

External sensors include a pair of color vision cameras mounted on the ceiling of the robot area to help players and robot localization, using a PCI Imaging frame grabber, a wireless microphone – TOA – for voice recognition, and laser detectors for location system.

2.4 Degrees of freedom

The robot head is mounted on a neck – PTU-Directed Perception – with 2 degrees of freedom – pan, tilt –. The robot eyes are constituted by two motorized cameras that contribute with 2 mechanical degrees of freedom – pan, tilt – and 2 optical degrees of freedom – zoom, focus –.

An holonomic system allows for the movement of the base with 4 wheels driven by 2 motors each one – wheel rotation and translation –.

2.5 Power Systems

A microcontroller based system is in charge of power distribution. The mobile platform has a main battery set with four 33 Amp·h batteries. The platform contains also an auxiliary battery set with four 18 Amp·h batteries in order to supply some extra power for those additional devices added to the mobile platform: The upper PC – TopRobot –, a speaker, the CONAC emitter and the video transmitter. Two DC-DC conversors and a devices power control board supply upper body systems from the battery sets.

2.6 Communication Systems

All the machines compose a two segments local network connected by means of Wireless network interface – Lucent Technologies Wireless IEEE 802.11 interface in 2.4 GHz DS – that uses the lower body as gateway. Internal robot communication systems include a 100 Mb/sec Fast Ethernet linking the robot's main processors – upper and lower body – and an Arcnet network for transmission of information between microcontrollers and the platform main processor. External systems are connected using a classic ethernet.

Audiovisual data are transmitted from the robot by means of a video-audio transmitter – Eagle 2.4 GHz PAL –.

3 Functional description: CAV in Eldi

A major breakthrough accomplished during this first phase has been the software architecture and associated methodology used to control the robot and off-board systems. The system that controls Eldi and the rest of the installation has been conceived as a set of agents that interact by means of discrete events. Eldi has been built using an extended version of CAV [7], a tool that enormously eases the definition and implementation of distributed systems modeled as discrete event systems. It was devised initially to facilitate the development process and reduce the integration effort involved in the design and implementation of active vision systems [9]. It permits to model a set of interacting control modules as a set of parallel/concurrent, asynchronous and weakly coupled agents interacting by means of events or signals, where in turn, each agent is modeled as a Port Automaton [10] [8]. CAV makes all agents, both on-board and off-board, share the same

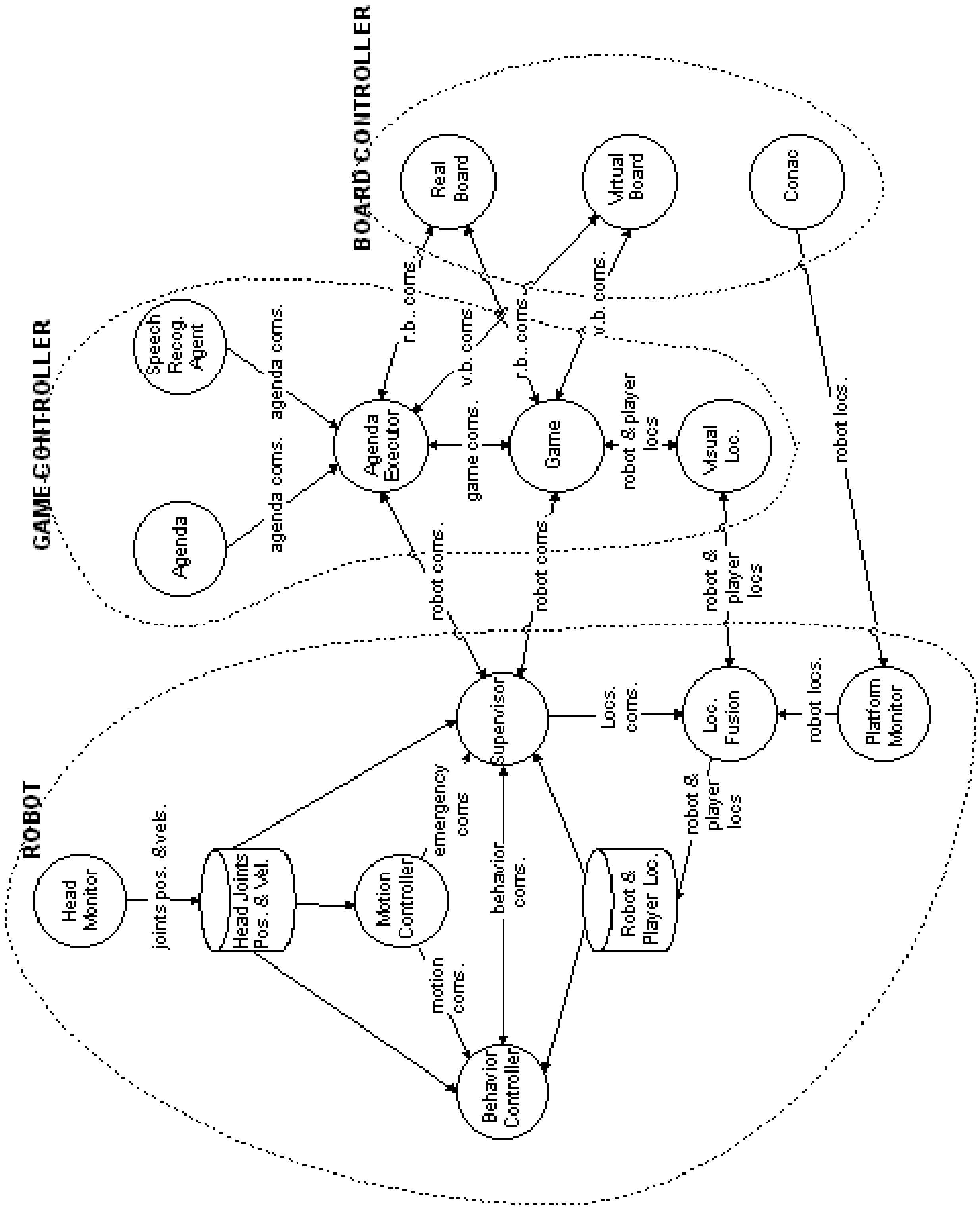

Figure 2. Software architecture under CAV

control and communication scheme, endowing each agent with an uniform external interface through with events and signals are issued and/or received, being the agent whether local or remote, and confers each one an internal uniform automaton structure as well. All this proved to be crucial during the integration of the whole system and sets the basis for system extension and modification along future phases of the project.

In figure 2 are depicted the main CAV agents involved in the current phase of project Eldi, there are other secondary agents, that are their subsidiaries, which are not shown. In the diagram, circles are agents, the cylinders are memory storages, arrows are data flows among agents, bidirectional arrows implies a protocol of commands request and response. Functionally, as displayed by means of dashed contours on the diagram, there are three different computer systems:

The ROBOT: That physically includes the mobile platform and TopRobot, the upper part. Responsible for controlling both low level platform and head movements, it is in charge of video transmission to external machines, and the extra power supply units. Some brief notes about the main agents involved by this entity are:

- **Supervisor:** Entrance door for commands from agenda and player agents. Any of these commands would be properly translated to other agents actions.
- **Head Monitor:** In charge of retrieving head motors state.
- **Motion Controller:** Controls the proper state of head motor commands and checks if a command has finished.
- **Platform Monitor:** Responsible of collecting data from platform sensors – ultrasonars, infrared, bumpers, motor temperature sensors and odometry –, providing and estimation of the robot localization.
- **Behavior Controller:** According to those commands received by the Supervisor and the data about robot localization and head motors positions, it is able to perform an action previously defined as a behavior. The behavior structure provides tools for integrating loops, conditions, timers, using variables and even calling another behaviors. In this way just including new behaviors could provide new possibilities for the system. Current sample behaviors are: look to the right, go to the 3-4 cell, track someone, dance, solve the puzzle, etc.

The GAME CONTROLLER: It is responsible for:

- **Speech Recognition Agent:** Attending user commands through speech recognition – using IBM ViaVoice –.
- **Agenda:** The robot uses an agenda for performing the shows, using the system clock,a task can be defined to be executed at a certain time or every x minutes. A typical agenda routine would be to take care of synchronization by means of the discrete signal events among different agents which runs completely asynchronously.

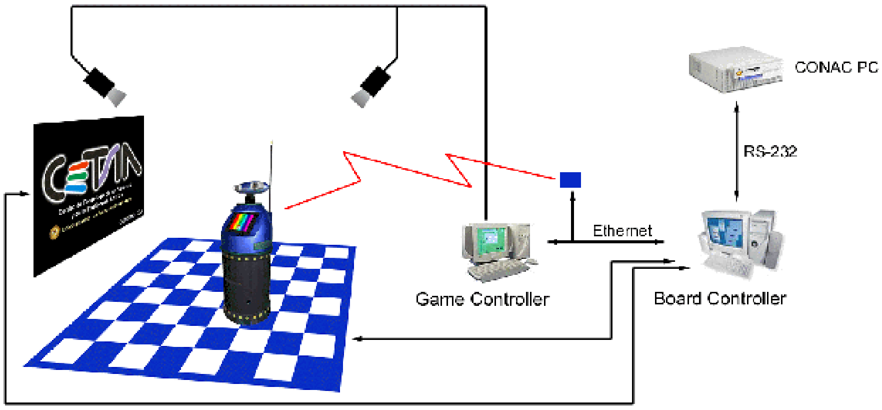

Figure 3. Complete system schema. The robot moves on the chess-like board – the real board –, a video board presents synthetic feedback of the game – virtual board –, and the external machines and cameras allow controlling the game.

Agenda Executor: Executes agenda commands.

Game: Controlling the evolution of board games.

Visual Localization: Robot and player visual localization using ceiling cameras for board games.

The BOARD CONTROLLER: In charge of controlling the game board (the Real Board), the video board (the Virtual Board) and the CONAC systems.

Real Board: In charge of providing light effects to the floor, music, etc.

Virtual Board: Responsible for commanding graphical output to the screen located in the hall, providing a metaphor for the live game.

Conac: Hosts an agent collecting information about robot localization from the CONAC system. As the platform odometry fails these extra information is crucial for the right coordination of robot movements.

An agenda or voice command will produce the execution by means of the agenda executor agent of different commands in the Real Board, the Virtual Board and the Robot Supervisor. Each agent would be in charge of controlling their own devices.

4 Daily activity

Currently, the daily activity of the robot cycles between a show that is carried out over a back lighted 8x8 glass board. Over that board (the Real Board) Eldi develops several shows and plays different games as "Treasure search", figure 4, or interacts with a visitor to solve an instance of an 8-puzzle, figure 5, using vision and commanding each new movement using voice. Additionally it performs a choreography combining music, video and game board light effects. Furthermore, there is a resting period while Eldi recharges batteries and offers the public the opportunity either to play different games as mastermind, chess or four-in-a-line, or to learn more about robotics using the multimedia information system and the tactile screen on its "torso". Some of the available games has been programmed to offer the opponent explanations of the actions taken by the robot during the course of the game or, to give comments that reflect the judgement of the robot about player's actions.

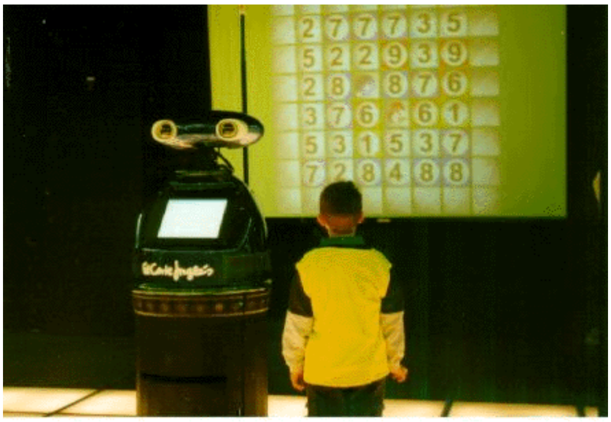

Figure 4. Frame of Treasure Search game

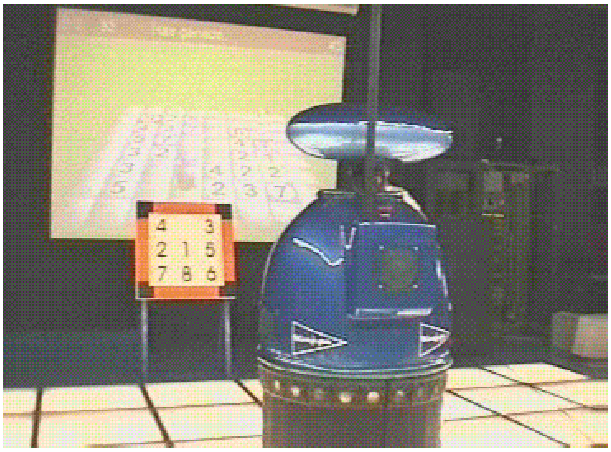

Figure 5. eldi solving the puzzle

5 Conclusions.

First we have to emphasize that Eldi is an ongoing project which is still facing its early stages and that has been acting at the Elder Museum of Science and Technology at Las Palmas de Gran Canaria since December 1999. With Eldi the objectives prefixed initially in this first stage, *The Player*, the integration of an entertainment robot under a reliable and extensible agent-based control scheme in a real human environment, has been attained. Reliability is another important aspect which has been carried out in this first phase. After installation and initial verification of the system by its developers, it has been and is maintained, or at least, operated on a daily basis by personnel not specialized in these systems. If the system were hard to maintain or did not assist the maintainer in discovering the cause of a failure, the system could not have survived the first problems. Reliability need to be addressed not only during a startup checkouts but specially during operation. This demands a software control architecture that must guarantee the correct operation of the different parts of the system both hardware and software. These goals have been partially addressed within this project associating watchdog timers with communication links and light weight verification daemons with device controllers.

Modern robotic systems are typically too complex to be developed and operated using conventional programming techniques. Overall system complexity can be reduced by decomposing it into smaller units or modules with well-defined abstraction levels and interfaces between them. The Eldi project, in its first phase, has constituted a challenging opportunity to put into practice some ideas about how to model a robotic system as a network of concurrent/parallel/distributed and cooperative agents, each one sharing a uniform internal structure and an uniform external interface [7], using a software framework previously tested successfully in the field of active vision systems [9].

In our opinion, these facts emphasize the importance of a suitable control architecture and associated design implementation that must hold the extensibility and demands of easy integration of these systems,in particular, and of mobile robotics, in general.

Perhaps the most important lesson learnt from the Eldi project is that a museum robot must be conceived as a living being. Shortly after the initial goals had been accomplished, the museum staff has required the development team to add new capabilities to the robot to allow for new or better shows or activities. Indeed, this situation must be considered in the light that these "pieces" normally capture a great deal of attention from the visitors and it is not unusual that they end up being considered as the "flagship" of the exhibition. A logical consequence is that staff demands to constantly update the shows or to add new capabilities to the robot, to renew the interest of the public and to attract new visits. Surprisingly, we have observed, as other authors [6], that it is the emotional and expressive abilities of the robot what captures much of the people's attention and not its navigational or obstacle avoidance capabilities. Most people do not realize, and do not mind, if the robot is avoiding obstacles or not. People enjoy frequently catching robot attention (cameras) and seeing themselves on the videoboard or on Eldi's screen. In our opinion, a clear indication, of the type of expectations these type of robots poses on the general public.

References

[1] The kismet project. URL: http://www.ai.mit.edu/projects/humanoid-robotics-group/kismet/kismet.html.

[2] The minerva project. URL: http://www.cs.cmu.edu/∼minerva/.

[3] The rhino project. URL: http://www.informatik.uni-bonn.de/∼rhino/.

[4] The sage project. URL: http://www.cs.cmu.edu/∼illah/SAGE/index.html.

[5] Toschi doi citation. excerpt from Computists' Weekly, AI Brief section in Intelligent Systems, December 1999.

[6] W. Burgard, A.B. Cremers, D. Fox, D. Haehnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. Technical Report CMU-CS-98-139, Carnegie Mellon University, Pittsburgh, PA. Available through URL: http://www.cs.cmu.edu/∼thrun/papers/thrun.tourguide_tr.pdf, 1998.

[7] A.C. Domínguez-Brito, F. Mario Hernández-Tejera, and J. Cabrera-Gámez. A control architecture for active vision systems. Frontiers in Artificial Intelligence and Applications: Pattern Recognition and Applications, M.I. Torres and A. Sanfeliu (eds.), pp. 144-153, IOS Press, Amsterdam., 2000.

[8] Antonio C. Domínguez-Brito, Magnus Andersson, and Henrik I. Christensen. A software architecture for programming robotic systems based on the discrete event system paradigm. Technical Report CVAP 244, Centre for Autonomous Systems, KTH - Royal Institute of Technology), S-100 44 Stockholm, Sweden, September 2000.

[9] M. Hernández, J. Cabrera, M. Castrillón, A. Domínguez, C. Guerra, D. Hernández, and J. Isern. Deseo: An active vision system for detection, tracking and recognition. in Computer Vision Systems, H. I. Christensen (Ed.), Vol. 1542 of Lecture Notes in Computer Science, Springer-Verlag, 1999.

[10] M. Steenstrup, M. A. Arbib, and E. G. Manes. Port automata and the algebra of concurrent processes. *Journal of Computer and System Sciences*, 27:29–50, 1983.