Proceedings of IEEE SMC 2004, The Hague, The Netherlands, pp. 5368-5373, IEEE omnipress, 2004.

Conception and Realization of a Multi-Sensory Interactive Mobile Office Guide*

C. Martin, H.-J. Böhme and H.-M. Gross
Ilmenau Technical University, Department of Neuroinformatics
P.O.Box 100565, 98684 Ilmenau
E-mail: {christian.martin, hans-joachim.boehme, horst-michael.gross}@tu-ilmenau.de

Abstract – *This paper describes the conception and realization of a Multi-Sensory Interactive Mobile Office Guide. We designed a mobile robot based on a modified Pioneer-2 robot, which is able to welcome visitors in our department and guide them to the desired staff member. The main components of the system are a vision based Multi-Person-Tracker and the existing navigation toolkit CARMEN. Furthermore, the robot provides a bidirectional video conference system and a speech synthesis system. Experimental results show, that the implemented Multi-Person-Tracker is accurately able to track an unknown number of persons in real-time and guide them to the respective people, while keeping an eye on the interaction partner.*

Keywords: Multi-Person-Tracking, image processing, mobile robot conception, system integration.

1 Introduction

The goal of our work was to provide a mobile robot as a demonstrator for an interactive guidance-task in the office environment of our department. For this purpose, the system usually waits near the entrance door for approaching people. When a new potential user appears, the robot offers its services. Now the user needs to manually confirm the interaction by clicking on the touch-screen. After that, the user can interact with the robot in a graphical dialog. This dialog is commented by the robot by means of its speech synthesis system. For additional requests, the robot provides a video conference to the main office of our department. Finally, the user can select a staff member, and the robot guides the user to the respective office. On the way from the entrance door to the office, the robot does observe the user visually by means of its omnidirectional vision system, to make sure, that the user is really following it. At the end of the guidance tour, the robot drives back to the entrance door to welcome the next visitor.

The control architecture of the robot guide is based on a closed interaction cycle between the robot and the user (figure 1).

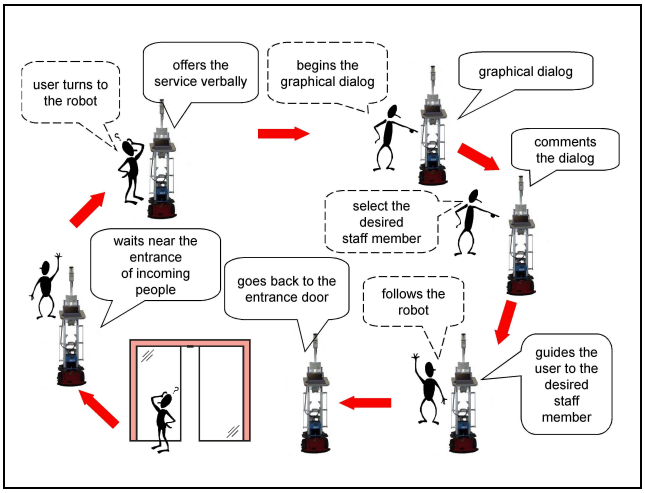

Figure 1: The used interaction cylce between the robot and the user: The robot welcomes people at the entrance door, gives help, provides a video conference and guides people to the desired staff members.

To realize such a system, a set of different robotics and computer vision methods had to be integrated: people detection, people tracking, collision avoidance, localization, navigation, speech synthesis, and video conferencing.

In this paper, we will focus on the omnivision-based probabilistic Multi-Person-Tracker and on the design of a suitable interaction cycle between the robot and the user.

This paper is organized as follows: In section 2 we discuss some related work. Our robot will be described in section 3. In sections 4 and 5 the different components of the robot are explained. In section 6 some experimental results are shown and section 7 gives an outlook on our future work.

2 Related work

In the past, a number of tour guide robots has already been used in public areas, e.g. the systems RHINO [1] and MINERVA [11] in museums. These systems didn't use vision systems for the human-robot interaction at

all. The people detection and people tracking was exclusivly done with laser range finders.

A robot system which is very similar to our target system is MOBSY [16]. It is an autonomous mobile system developed at the University of Erlangen-Nuremberg, Germany. MOBSY welcomes new visitors at the institute and serves as a robotic receptionist. It is equipped with a vision system for people detection and tracking (based on skin color), a speech synthesis and a speech recognition. However, MOBSY can only navigate in a very small area (3x8 meters) in front of the elevators, where new visitors normally appear.

3 Hardware of the robot

All our implementations and experiments were carried out on a modified Pioneer-2 robot (see figure 2).

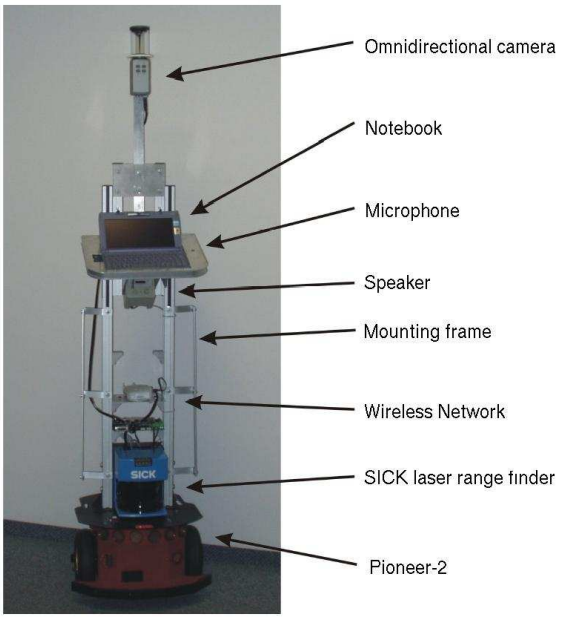

Figure 2: The modified Pioneer-2 platform, equipped with an omnidirectional digital camera, an additional notebook, a speaker, a microphone, and a WLAN adapter.

The robot is equipped with a Pentium-III 1.4 GHz PC, a WLAN adapter and a digital camera with an omnidirectional optics (see figure 4a). The camera (SONY DFW-VL500) with the omnidirectional optics (a hyperbolic mirror) is mounted at a height of about 170 cm.

4 Multi-Person-Tracker

A very important task for our robot is the localization and tracking of (potential) interaction partners. In most cases, there will be more than just one person in the robot's field of view: There can be other people in the background of the scene or people who are passing by. Many robots are only able to track one person at a time. If the person moves around and gets occluded by other people or obstacles, it is very hard to continue to track this person. Also it is undesired that the robot takes another person as a new interaction partner in such situations. But if the robot would be able to keep track of the different people in the scene, such situations could be handled much easier. Hence, it is not sufficient to track only one user.

Therefore, we need a tracking system for our robot, which is able to track more than just one person in a natural environment. Some known person detection algorithms purely rely on non-visual sensor systems such as laser range-finders [9]. With such systems it is possible to detect the presence of one or more persons in the robot surroundings, but they can't make any statement of whether this person is facing the robot as an expression of his will for interaction. In the future, it will be very important to design algorithms for low-cost robots. Therefore, we explicitly excluded the very expensive laser range-finder from the person tracker. In the long run, we mostly want to use a low-cost vision sensor to observe information from the environment.

Last but not least, the tracking system must work in real-time on a mobile robot with an Embedded PC.

4.1 Tracker architecture

Our tracking system can be divided in three main components (see Figure 3): First, an omnidirectional vision system to detect the primary features (skin color). Second a sampling algorithm (based on the CONDENSATION algorithm [4]) to track the positions of an unknown number of persons. And finally, a specific face detector to verify the hypotheses of the different user positions.

Our approach is an extension of a former tracker version described in [15]. The difference is, that our system is able to track more than one person simultaneously.

4.2 Feature extraction

To construct a low-cost system, for the Multi-Person-Tracker we only use the omnidirectional camera. An image taken from this camera can easily be transformed into a 360° panoramic image. Due to the shape of the used mirror, there will be a distortion in the y-direction in the panoramic image, which can be easily approximated and corrected by a quadratic parable.

In our work, we use the skin color as primary feature. For the skin color detection the dichromatic r-g-color space

$$r_i = \frac{R_i}{R_i + G_i + B_i}, g_i = \frac{G_i}{R_i + G_i + B_i} \qquad (1)$$

is used, which is widely independent from variations in luminance. The color detection can be calculated very fast because it consists only of a lookup operation, but on the other hand, it is highly dependent on illumination color and variations in hue. Therefore, a white reference ring is placed in-between the camera and the objective

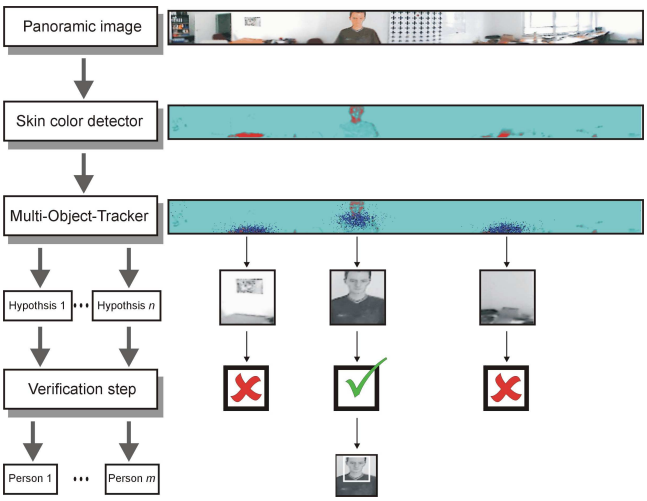

Figure 3: The three main components of our Multi-Person-Tracker: a primary feature detector (skin color), the Multi-Object-Tracker and the verification system (face detector).

(see Figure 4) to perform a white balance step for an automatic color calibration. Details of this correction approach can be found in [3] and [15].

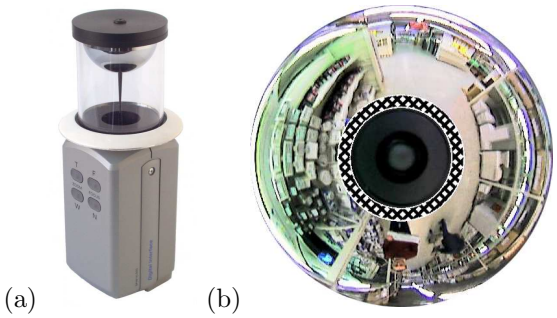

Figure 4: (a) SONY DFW-VL500 camera with a hyperbolic mirror and a white reference in-between camera and objective. (b) An image taken with this camera: the white reference is located near the center region of the image (marked with a checkered pattern).

4.3 A sampling algorithm for multi-object tracking

The used Multi-Object-Tracker is based on the work of Tao et al. [10], which is an extension of the CONDENSATION algorithm [4]. In order to be able to represent multiple objects, Tao et al. enhanced the basic representation to a so called *object configuration*, which represents all objects in an image. Such a configuration can be expressed by a set of parameters:

$$s_t = \{x_{t,1}, x_{t,2}, ..., x_{t,m}\} \in X^m \tag{2}$$

where $x_{t,j}$ stands for the state of an object j at time t and m is the number of objects.

The goal is to compute the posterior probability of the configuration parameters by means of the following Bayes filtering:

$$P(s_t|\mathbf{Z}_t) = P(z_t|s_t) \int P(s_t|s_{t-1}, u_{t-1}) \cdot P(s_{t-1}|\mathbf{Z}_{t-1}) ds_{t-1} \tag{3}$$

where $\mathbf{Z}_t = \{z_t, u_{t-1}, z_{t-1}, ..., u_1, z_1, u_0, z_0\}$ is the set of made observations z_t and given controls u_t (the robot movements).

The term $P(z_t|s_t)$ is called *configuration likelihood* and describes how good the observation z_t can be explained by the configuration s_t. The other important term is $P(s_t|s_{t-1}, u_{t-1})$, which is called *configuration dynamics* and describes how a configuration s_{t-1} is changed into s_t in consideration of the control u_{t-1}. In the next two subsections both terms will be described in more detail.

4.3.1 Configuration likelihood

The likelihood $P(z_t|s_t)$ is a very complicated distribution. In our work, we use the same approximation as in [10], which uses a so called *configuration decomposition*. Tao et al. designed an energy function, which gives higher values to more desired configurations. This energy function consists of three factors. The first factor describes how the objects in the configuration s_t fit with the observation z_t and is called *object-level likelihood*. To compute this factor, we compute the likelihood $L(z_t, x_{t,i})$, which measures how well the image data z_t supports the presence of the object $x_{t,i}$. This can be easily done with the skin color detector.

The second factor is how much of the observations has been explained by the configuration (*configuration coverage*). The last factor is called *configuration compactness*. It is more desired to explain the observations by a minimum number of objects. This way, this factor gives higher values to configurations with fewer objects.

To compute the last two factors, in [10] motion blobs are used. In our work we use skin color blobs, which can be easily detected as described in section 4.2.

4.3.2 Configuration dynamics

The configuration dynamics $P(s_t|s_{t-1}, u_{t-1})$ can be decomposed into object-level and configuration-level dynamics.

First, the object-level dynamic $P(x'_{t,i}|x_{t-1,i}, u_{t-1})$ is applied to predict the behavior of each object. The result is a set $s'_t = \{x'_{t-1,1}, x'_{t-1,2}, ..., x'_{t-1,m}\}$. The behavior of the objects can be modeled with a simple state transition matrix and a Gaussian noise. This step can be compared with the motion model of the CONDENSATION algorithm. In our case, we use the known movement of the robot as the state transition matrix. The movement of the people is modeled by the Gaussian noise around the current position.

In a second step, the configuration-level dynamic is applied. This dynamics models the appearance and dis-

appearance of single objects. Therefore, an addition probability and a deletion probability is used. In our work, it is sufficient to use a small constant value (e.g. 0.01) for both probabilities. We also tried to use a function of image coordinates (x, y) as proposed in [10]. But it turned out experimentally, that in the case of an 360° panoramic image this does not lead to a better performance.

4.3.3 Complexity considerations

It is obvious, that the *configuration space* X^m is exponential in the number of objects m. To guarantee the same sample density with growing m, an exponential number of samples is necessary. This causes an exponentially computation complexity as discussed in [5]. For small m however, this algorithm can still be computed in real-time on a standard PC. In our work, we use $m = 5$, which is entirely sufficient for the visual tracking of multiple persons, i.e., our typical real-world requirements.

Another solution to this problem would be the *hierarchical sampling algorithm* presented in [10].

4.4 The face detector

The output of the Multi-Person-Tracker provides a set of hypotheses of positions of the sourrounding people or people-like objects based on skin color features. In a natural environment, however, there are numerous skin colored objects, which are not humans. Especially objects with a wooden surface are very similar to skin color in the most color spaces. Therefore, it is necessary to verify all hypotheses, to make sure that only humans are contacted and tracked.

In our implementation, we use the face detector by Viola and Jones described in [13, 14]. This face detector does work very fast, while, at the same time, it has high detection rates and very low false positives rates [15].

Due to the limited resolution and quality of the 360° panoramic image taken from the omnidirectional camera, we only can extract potential face images with a size of approximately 25x25 pixels with a "medium" quality. In some cases it is hard for the face detector, to detect the faces certainly. Therefore, we integrated a simple heuristic rule to keep track of the faces: At the first time during the interaction, the person must be verified by the face detector. In the following time steps, it is allowed that the face detector temporarily fails and we only use the skin color to keep track of the person. For each detected person i, we are using a certainty measure c_t^i:

$$c_t^i = \begin{cases} 1.0 & \text{if person is successfully verified by means of the face detector} \\ \gamma \cdot c_{t-1}^i & \text{otherwise} \end{cases} \quad (4)$$

with $0 < \gamma < 1.0$. If c_t^i drops below a certain threshold (e.g. 0.2) the tracker will loose the person.

5 Other components of the robot

Besides the Multi-Person-Tracker and the interaction cycle, there are more components necessary to realize the Mobile Office Guide:

- Navigation: Till now, the robot navigates through its environment based on the input of a SICK laser range finder. In our work, we use the Carnegie Mellon Navigation (CARMEN) toolkit [6] for collision avoidance, localization and navigation.
- Video conference: To provide a video conference system on the robot, we have choosen the standardized ITU-T H.323 teleconferencing protocol. More precisely, we use an open source implementation of this standard (called OpenH323 [8]).
- Speech synthesis: For the speech synthesis we are using Txt2pho [12]. This is a German text to speech front end for the MBROLA [2] synthesizer.

All these components are integrated in one application. For the information exchange between all the different components, a blackboard architecture [7] was implemented.

6 Results

We have successfully tested all subsystems (collision avoidance, localization, navigation, people detection, people tracking and video conferencing) in a real office environment, i.e. in the hallway of our department (40 meters long with 15 offices).

Especially the vision-based Multi-Person-Tracker was extensively tested.

6.1 Results of the vision-based tracking

In order to analyse the capabilties of the presented Multi-Person-Tracker, the robot was placed in the center of a hallway in our office environment. During the experimentes, a number of people appeared in the robot's field of view. The people stayed there a few seconds and than disappeared.

Figure 5 shows a typical course of the experiment. For the experiment, a subset of 360° panoramic images and the corresponding feature images are shown. A white rectangle in a feature image represents a detected hypothesis of a person's position and a black rectangle stands for a face successfully verified by the Viola and Jones face detector. At the bottom of the frame sequence, a diagramm shows the number of detected people over time. The light-grey curve illustrates the output of the Multi-Person-Tracker and the black one shows the real number of people, which was obtained by manually counting.

The appearance of the different persons was detected correctly. It is visible, that the tracker needs more time to detect the correct number of people as the number of people in the field of view increases. This can be

explained by the growing complexity of the problem. There is also a little artifact during the disappearance of the second person (frame 115). This was probably caused through the restructuring of the particles in the Multi-Person-Tracker.

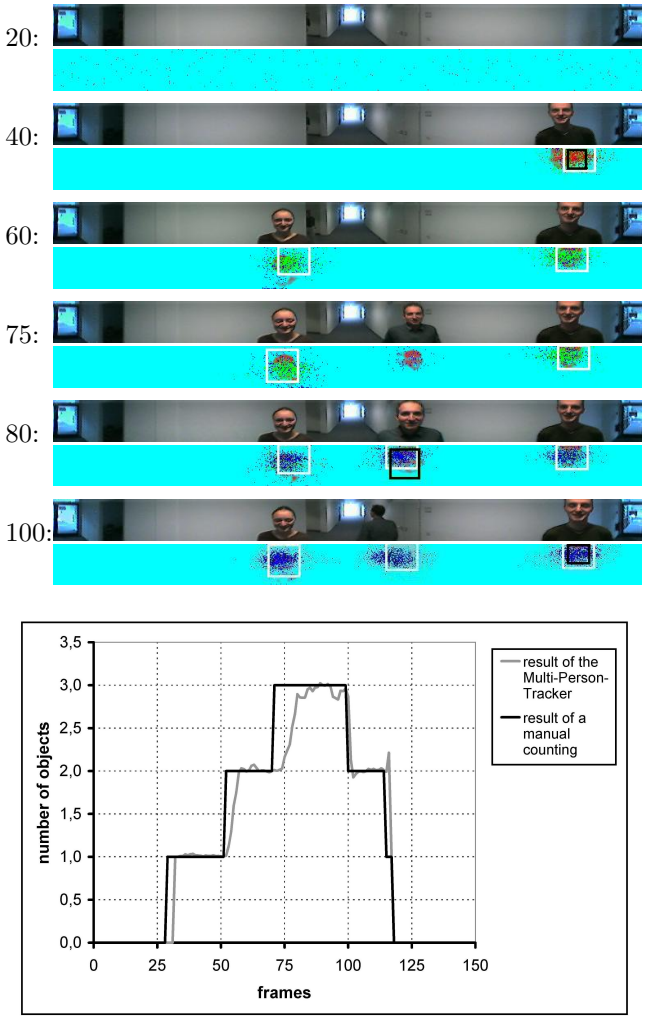

Figure 5: Top: Sequence of panoramic images and the corresponding feature images with hypotheses (white rectangles) and verified faces (black rectangles). Bottom: Time course of the number of tracked persons in comparison to the actual number of persons.

In another run of the experiment it turned out, that our configuration has problems with very tall persons. One person was so large, that the face was only partially visible in the panoramic image (primarily the forehead was not visible). Therefore, the structure-based face detector was not able to verify the person. Since this problem is caused through the mounting frame of the robot, it is not a problem of our Multi-Person-Tracker approach in general.

Despite the mentioned problems, this results show that the implemented Multi-Person-Tracker is able to track a changing number of people in real-time on a robot with an Embedded PC.

6.2 Tracking during the guidance tour

Besides the tracking of the potential users when the robot is not moving, it is important to keep track of the current user during the guidance tour. Figure 6 shows a typical result of one tracking experiment during a guidance tour.

In turned out, that the face of the user becomes very small in the omnidirectional images, when the distance between user and robot grows over 1.5 meter. The faces are so small, that our tracker in unable to continously track the user certainly. Therefore, we use a certainty measure (equation 4) to keep track of the person. If the certainty measurement drops below a threshold, the robot stops and asks the user to come closer.

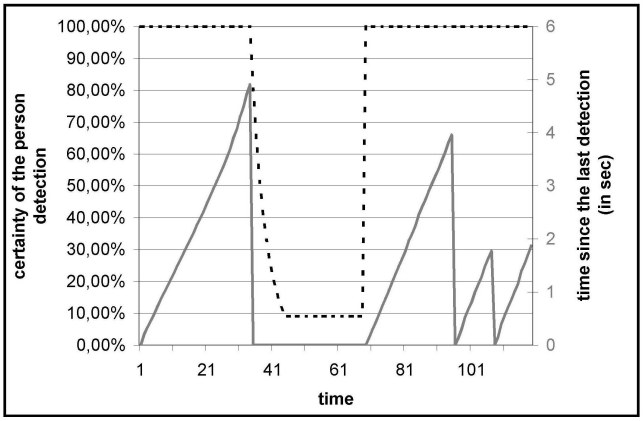

Figure 6: Certainty of the person tracker during a guidance tour. The solid line shows the time difference since the last successful person detection and the dashed line shows the resulting certainty.

6.3 Interaction between user and robot

Another important experiment was the testing of the designed interaction cycle between the user and the robot. Figure 7 shows a typical example of an interaction process between an user and the robot. It is visible, that not all system components are active over the whole time. To activate or deactive single components, each module is able to send appropriated events to other modules. This events are illustrated by the arrows in figure 7.

7 Outlook

At the moment, we are using the robot guide only temporarily in the hallway and the entrance area of our office environment. Due to some technical issues (especially the power supply) we're not yet able to use the robot guide permanently.

In the near future, we will test other optical systems (omnidirectional mirrors with other shapes or fish-eye

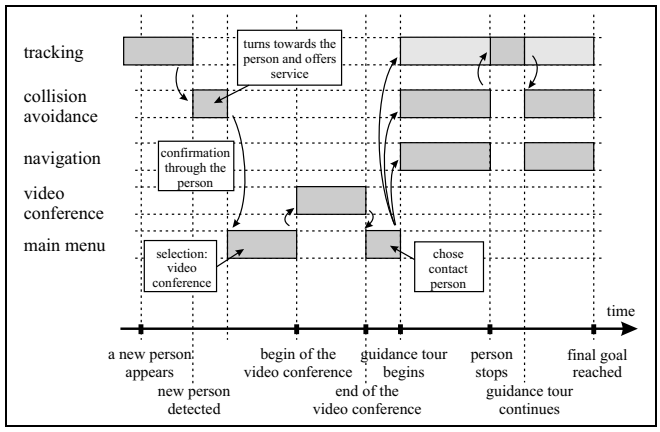

Figure 7: A typical example of an interaction process between an user and the robot.

lenses). Furthermore, the robot needs to be equipped with an animated face, to be able to express a set of basic emotions. Another important task for the future will be the integration of a speech recognition system to make to communication between the robot and the user much more easier.

8 Acknowledgement

This work was founded by Thuringian Ministry of Science, Research and Arts (grant number: B509-03007).

References

[1] Burgard, W., Fox, D., Hähnel, D., G. Lakemeyer, G., Schulz, D., Steiner, W., Thrun, S., and Cremers, A.B. Real Robots for the Real World – The RHINO Museum Tour-Guide Project. In *Proc. of the AAAI 1998 Spring Symposium on Integrating Robotics Research*, Stanford, CA, 1998.

[2] Dutoit, T., Pagel, V., Pierret, N., Bataille, F., and van der Vreken, O. The MBROLA Project: Towards a Set of High-Quality Speech Synthesizers Free of Use for NonCommercial Purposes. In *Proceedings of the International Conference on Speech and Language Processing (ICSLP)*, volume 3, pages 1393–1396, Philadelphia, 1996.

[3] Gross, H.-M., Koenig, A., Schroeter, Ch., and Boehme, H.-J. Omnivision-based Probalistic Self-localization for a Mobile Shopping Assistant Continued. In *Proc. IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS)*, pages 1505–1511, Las Vergas, USA, 2003.

[4] Isard, M. and Blake, A. Contour tracking by stochastic propagation of conditional density. In *Proc. of the European Conference of Computer Vision (ECCV)*, pages 343–356, 1996.

[5] MacCormick, J. and Blake, A. A probabilistic exclusion principle for tracking multiple objects. In *Proc. Int. Conf. Computer Vision (ICCV)*, pages 572–578, 1999.

[6] Montemerlo, M., Roy, N., and Thrun, S. Perspectives on Standardization in Mobile Robot Programming: The Carnegie Mellon Navigation (CARMEN) Toolkit. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, volume 3, pages 2436–2441, 2003.

[7] Nii, H. P. Blackboard Systems. *The Handbook of Articifical Intelligence 4*, pages 1–82, 1989.

[8] OpenH323 Project Homepage. WWW: `http://www.openh323.org/`.

[9] Schulz, D., Burgard, W., Fox, D., and Cremers, A. Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *Proc. of the IEEE International Conference on Robotics & Automation (ICRA)*, 2001.

[10] Tao, H., Sawhney, H.S., and Kumar, R. A sampling algorithm for tracking multiple objects. In *Workshop on Vision Algorithms*, pages 53–68, 1999.

[11] Thrun, S., Bennewitz, M., Burgard, W., Cremers, A.B., Dellaert, F., Fox, D., Hähnel, D., Rosenberg, C.R., Roy, N., Schulte, J., and Schulz, D. MINERVA: A Second-Generation Museum Tour-Guide Robot. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1999–2005, 1999.

[12] Txt2pho - German TTS front end for the MBROLA synthesizer. WWW: `http://www.ikp.uni-bonn.de/dt/forsch/phonetik/hadifix/HADIFIXforMBROLA.html`.

[13] Viola, P. and Jones, M. Fast and robust classification using asymmetric adaboost and a detector cascade. In *NIPS 2001*, pages 1311–1318, 2001.

[14] Viola, P. and Jones, M. Robust real-time object detection. *Proc. of IEEE Workshop on Statistical and Computational Theories of Vision*, 2001.

[15] Wilhelm, T., Böhme, H.-J., and Gross, H.-M. Looking closer. In *Proc. of the 1st European Conference on Mobile Robots*, pages 65–70, Radziejowice, 2003.

[16] Zobel, M., Denzler, J., Heigl, B., Nöth, E., Paulus, D., Schmidt, J., and Stemmer, G. MOBSY: Integration of Vision and Dialogue in Service Robots. In *Computer Vision Systems, Proceedings Second International Workshop (ICVS)*, pages 50–62, Vancouver, Canada, 2001.