A User-Interface Robot for Ambient Intelligent Environments

A.J.N. van Breemen[†]
albert.van.breemen@philips.com

K. Crucq[†]
ko.crucq@philips.com

B.J.A. Kröse[††]
krose@science.uva.nl

M. Nuttin[‡]
marnix.nuttin@mech.kuleuven.ac.be

J.M. Porta[††]
porta@science.uva.nl

E. Demeester[‡]
eric.demeester@mech.kuleuven.ac.be

[†]Philips Research (SwA)
Prof. Holstlaan 4
5656 AA Eindhoven
The Netherlands

[††]University of Amsterdam (IAS)
Kruislaan 403
1098 SJ Amsterdam
The Netherlands

[‡]K.U. Leuven (PMA)
Celestijnenlaan 300 B
B-3001 Heverlee
Belgium

Abstract

This paper reports on the development of a domestic user-interface robot that is able to have natural human interaction by speech and emotional feedback. Natural interaction with the user is achieved by means of a mechanical head able to express emotions. Additionally, our robot is aware not only of its position in the environment but also of the position and intentions of the users. The localization of the robot is achieved with an appearance-based localization method. To get information about the users, the robot is designed to operate in (and to cooperate with) an intelligent ambient that takes care of monitoring the users. The result is a service robot that departs from existing developments in the fields of interface and service robots that are mainly reactive and, thus, limited in functionality.

1 Introduction

In the last years an increasing effort is spent in research on service and entertainment robots which operate in natural environments and interact with humans. The Sony AIBO is an example of a robot which is meant to play with children: it has a perceptual system (vision, auditory, tactile), plays soccer, and can learn its own behavior [1]. NEC has developed "Papero", a *personal* robot which also is able to entertain the user but has more functionality: it serves as an interfacing with web-services and electronic equipment [17]. Even more functionality is present in various other service robots, such as robot-waiters [12], museum robots [22] or care-for-elderly robots [10], all examples of autonomous intelligent systems, operating in a real world.

Most of the existing service robots are only able to exhibit reactive behaviors and this clearly limits their utility. To extend service robot's functionality we have to make them aware of their own position in the environment as well as of the position and intentions of other agents (i.e., the users).

To make the robot aware of its position in the environment (and, thus, able to navigate on it) we propose to use an appearance-based robot localization technique [24].

On the other hand, to make the robot aware of the intentions of users we propose to integrate the robot in another recent development in the 'services for humans': the *Ambient Intelligence*. This refers to a new paradigm in information technology, in which people are served by a digital environment that is aware of their presence and context, and is responsive to their needs, habits, gestures and emotions. Ambient Intelligent environments incorporate the following elements: ubiquity (surrounded by many networked embedded systems), awareness (locate objects and people, recognize their intentions), intelligence (reason, learn from interactions), and natural interaction (speech and gesture recognition, speech and motion synthesis). But the question is whether the user is going to talk to his toaster or coffee machine... We think not.

As a part of the European project "Ambience" [13] we develop a domestic robot (see Figure 1). The robot must be some personification of the intelligent environment, and it must be able to show intelligent behavior, context awareness and natural interaction. The robot exploits the intelligent environment to get information about the user intentions, preferences, etc. In the other way around, the human user must be able to have a natural interaction with the digital world by means of the robot.

Very important for the natural interaction is a nice look of the robot, and the possibility to express some emotional state. Many other robots use a (touch)screen interface, sometimes with an animated face [12, 6]. We decided to use a 'real' face, consisting of dynamic mouth, eyes and eyebrows since this makes the interaction more attractive and also more natural.

The development of software modules for different tasks is carried out by multiple developers. Therefore, we have implemented a dedicated software tool to support the developers. Using this tool, different software modules of the robot application, running on different operating systems and computers, can be connected/disconnected interactively at runtime by means of a graphical user interface. With this approach, integrating the different software components is a matter of "configuration" rather than programming.

The objective of this paper is to introduce the different modules developed for our robot, the software tools used to integrate them, and the preliminary results we have obtained so far. In Section 2, we describe the emotional feedback mechanism, which is the most important aspect for creating a natural interaction between the robot and the user. Next, Section 3, we describe the mechanisms to pay attention to the user at any moment (i.e., user localization and speech understanding). After this, we introduce the localization (Section 4) and navigation (Section 5) abilities of the robot. Section 6 introduces the *Belief Desire Intention* framework that is used to coordinate the rest of modules. Then, Section 7 describes the software tools used to integrate all the presented modules and in Section 8 we present the results we have obtained up to now. Finally, Section 9, summarizes our work and extracts some conclusions out of it.

2 Emotional Feedback Module

Nonverbal emotional feedback by means of facial expressions and head/body motion plays a very important role in human interaction. Small motions of the head, the mouth, the eyes, or the eyebrows are often sufficient to indicate the state of the interaction process e.g. not interested, enjoying, understanding, etc. We decided to use a real physical embodiment in the form of a 3D mechanical head. With others [8, 18], we strongly believe that the human user needs a natural physical entity along with the embedded intelligence of the interface. Additionally, the recent commercial success of toy robots (AIBO, I-Cybie, Poo-Chie) clearly indicates the attractiveness of such an embodiment.

Figure 1: The robot Lino.

Furthermore, a robot head with a nice, cute appearance and emotional feedback can be configured in such a way that the human user enjoys the interaction (fun, excitement) and will more easily accept possible misunderstandings: The human user remains feeling his superiority.

The physical embodiment of the robot is based on a Pioneer2 DXE platform [19] (see Figure 1). On top of the platform we have built the mechanical head (ø25 cm) with an attractive, well-groomed appearance. The head is equipped with mobile and controllable eyebrows, eyelids, eyeballs, and mouth shapes. It is supported by a neck with three degrees of freedom allowing for three basic movements: (1) turning left/right, (2) turning up/down, (3) approaching/withdrawing. The position of the various parts is controlled by 17 standard RC-servomotors. During the development, the aspect of a head with a nice, well-groomed appearance was emphasized as we consider this of importance for the evaluation of user tests. Other features have been largely determined by compromising factors as time, costs, and availability, like e.g. the artistic design, the size and the shape.

To coordinate the degrees of freedom of the robot's face to express emotions we use the OCC model [16]. This framework defines a simple interface to select the kind of emotion to express that is then automatically translated into motor movements.

3 User Awareness Module

We have implemented and tested one of the tasks of the robot called "Turn-to-speaker" behavior. This behavior basically determines the direction of a speaker in a 3D space and turns the head and body toward the speaker. The 3D speaker location estimation is determined by means of three mutually perpendicular microphone pairs. These microphones are mounted inside the head and are separated by a distance of 25 cm. Each microphone pair uses a stereo USB audio digitizer for the signal acquisition. We analyze the recorded signals to determine the difference in the time of flight of speech that arrives. The basic problem in this measurement is to get rid of the numerous acoustic reflections in the recorded speech signals. With an adaptive Filtered-Sum Beamformer and an optimization algorithm [3, 4] it is possible to determine the contribution of these reflections and to largely compensate for them in the recorded signals.

The location of the speaker is indicated in the local robot coordinate system by two angles, φ (horizontal plane), θ (tilt). The angle φ is used to turn the robot platform and θ is used to turn the head up to the speaker with the loudest voice. There is a problem when there are many speakers at the same level. In this case the system generates inconsistent values and we pause the turning of the robot. The system does not respond to random acoustic noise. It detects human voices by looking for harmonics (the pitch) in the recorded signal. With this technique we also want to explore "Follow-me" behavior: e.g. advancing small distances in the direction of someone speaking at regular intervals.

The identification of the speaker will be done in cooperation with the intelligent ambient. This environment uses the network of sensors distributed all around the house to be aware of the position of each user at any time. Thus, when the position of a given speaker is determined it is easy to query the intelligent ambient to determine the identity of this speaker.

Beside the sound localization, the robot is also equipped with speech recognition. This ability is used to accept verbal commands given by the user that can be either directly executed to the robot (simple navigation tasks, etc) or passed to the intelligent ambient for its completion. Finally, the robot is also able to generate speech (to interact with the user and to inform him/her of relevant events: completion of given robot/environment task, etc).

4 Robot Localization Module

It is well known that the estimation of the robot position from odometry is subjected to large errors

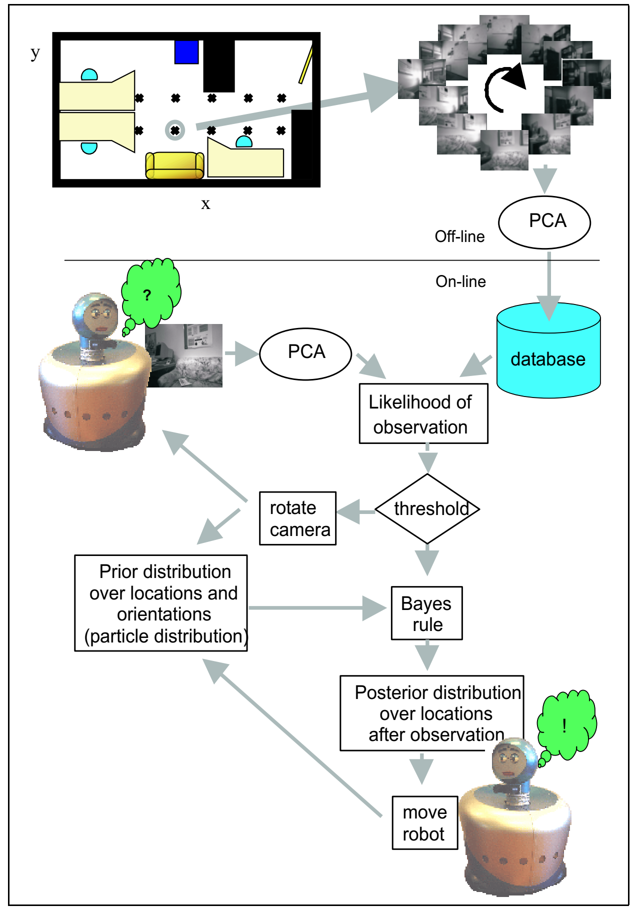
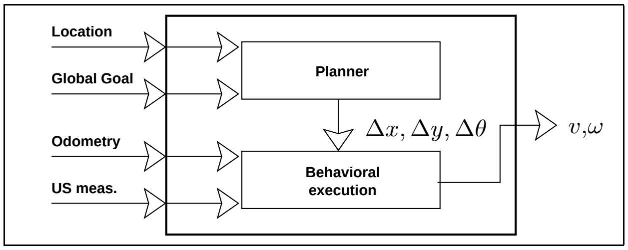


Figure 2: Sketch of the appearance-based localization process.

in the long term. Consequently, additional sources of information must be used to determine the robot's position. In our case, we advocate for the use of vision.

Many of the existing approaches for robot localization using vision techniques are based on the landmark detection paradigm [21]: images are scanned for the presence of special patterns and the position of these patterns is used to determine the position of the robot. Of course, errors both on landmark detection (occlusions, confusions, etc.) and on their position estimation decrease the localization accuracy. As an alternative, the appearance-based localization approach [15] is appealing for its simplicity. In this approach, the robot position is determined by a direct comparison between the image observed at a given moment and a set of images observed before. Note that, since the entire images are compared, we avoid the errors associated with the landmark detection procedures.

The appearance-based localization (see Figure 2)



departs from a training set including images shot at known positions. The dimensionality of these images is reduced using a standard PCA technique, producing an appearance-based map of the environment. When the robot is moving, the collected images are compressed using PCA and the resulting set of feature detectors is compared with those stored in the appearance-based map. The position of the map points closer to the current observation are used within a Markov process [14] to improve the estimation of the position of the robot. We have obtained good results combining this approach with the use of a particle filter to estimate the probability distribution on the robot's position [24].

In our previous work, we used an omnidirectional camera. The advantage of such a camera is that the provided image is sensitive to both orientation and positions changes. The disadvantage is that the presence of an object changes the whole image. Thus, to increase the robustness, within the "Ambience" project, we use a stereo camera with a more limited field of view. The head can be moved to point the camera in different directions. Therefore, in the presence of confusing images (images that do not match with any of the images in the appearance-based map) we can readily obtain a new image just moving this head. The idea is to select the head movement that is likely to provide the less ambiguous view. In this way, we will minimize the number of head movements necessaries to determine the robot's position [20].

An additional advantage of using a stereo vision system is that it can provide depth maps that are less sensitive to change in illumination than usual intensity images.

5 Navigation Module

The objective of the navigation module is to drive the robot to the desired positions (expressed as abstract placements such as *kitchen*, *bedroom*, etc) avoiding collision with obstacles.

To accomplish this objective, the architecture in Figure 5 is used in the navigation module. There are two internal modules: the planner and the behavioral execution module.

The planner generates subgoals from the robot's current location to the global goal position using a map. In previous work, a computationally efficient planner was developed based on the Wave Front Distance Field (WFDF) algorithm; see [23] for more details. This planner finds the shortest path to the goal and calculates subgoals on this path. The straight path between two consecutive subgoals is obstacle free. In a final step, subgoals that are close to each other are merged. This avoids that subgoals are too close together, which is not desirable in behavioral execution. The efficiency of the algorithm allows to re-plan approximately every second, coping with the robot getting off course in front of obstacles. Figure 3 shows the subgoals in the HomeLab (the domestic test environment at Philips Research Eindhoven, The Netherlands), from the robot's current location (L) to a certain goal position (G). The planner outputs the desired change in position in order to reach the first calculated subgoal. By doing so, this ultimately leads the robot to his final goal. The information provided by the localization module (see Section 4) is used determine the position of the robot and, thus, to keep track of the change in position already achieved.

The second component of the navigation architecture, the behavioral execution module, receives as input (from the planer) the desired relative displacement for the robot and determines the linear and angular speeds (v and ω, respectively) necessaries to perform it. Then, these speeds can be readily transformed to wheel motor commands. The behavioral execution is implemented using a behavior-based control approach. We refer to [2] for an introduction to behavior-based robots. Obstacles which are not in the map, both static and dynamic, possibly show up in front of the robot while moving. To avoid bumping into them, an avoid-obstacle algorithm is implemented. Ultrasonic sensors are used to detect these obstacles.

The cooperation of the fast planner module and the behavioral execution one leads the robot to his goals.

6 High Level Reasoning Module

In order for the robot to realize high level goals it must be capable of reasoning about the information it has about its world. A flexible reasoning mechanism that is dedicate to operate in such a practical problem domain as the domestic user environment is essential for a proper functioning of the robot. We've chosen to use the Belief, Desires and Intention (BDI) architecture that is well known in the field of agent and multi-agent systems. The roots of the BDI architecture can be traced back to the work of Bratman *et*

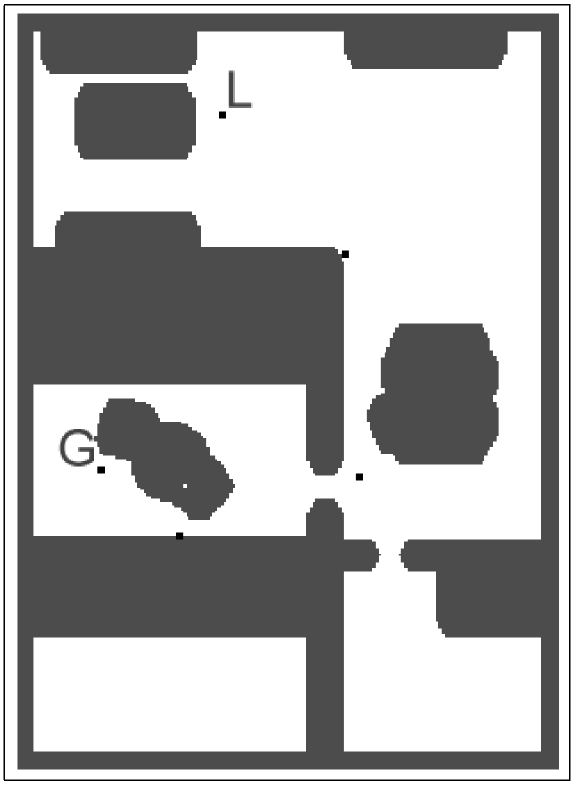


Figure 3: The planner uses a map to generate subgoals (the small dots in the figure) from the current location L to the goal position G.

al. [5] who discuss the issue that an architecture for resource-bounded agents (such as our robot) should incorporated means-end reasoning, weighting of competing alternatives and an interaction mechanisms for these two processes. In particular, the robot should aim at constraining the amount of practical reasoning.

A BDI architecture contains three sets of information

- A set of Beliefs which contains information about the robot's environment and internal state. In our case this will be information about users and their preferences, the presence of objects in the robot's environment and information about the robot's own actions and internal state.
- A set of Desires which contains information about the objectives or goals of the robot. In our application this will be typical goals such as serving users or the desire to go to a particular location.
- A set of Intentions which contains information about the actions the robot will execute in order to realize its desires.

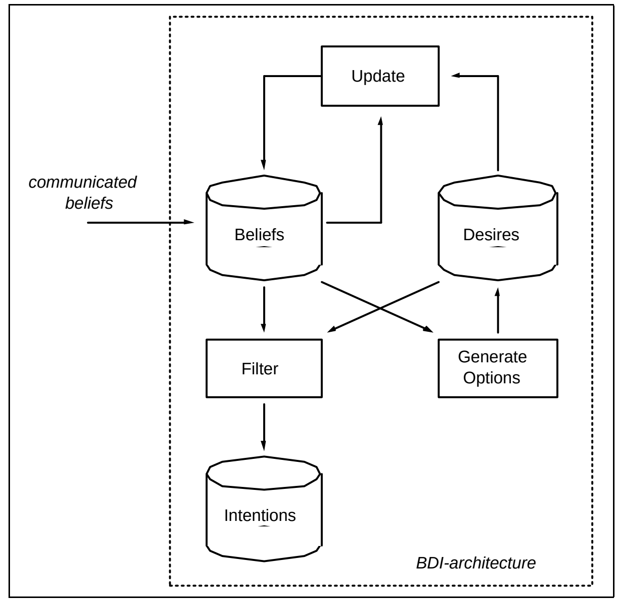


Figure 4: BDI architecture.

Figure 4 shows a diagram of the BDI architecture. In this diagram we find three databases representing the three sets of information (Beliefs, Desires, Intentions).

Besides the Beliefs, Desires and Intention sets, a BDI architecture also contains three functions which operates on these sets. These functions are [25]:

- The Generate Options function carries out means-end reasoning and thereby generating new Desires (goals). While doing so, it must maintain consistency between the Beliefs, Desires and Intentions. For instance, if the robot has a belief that a particular user is in the kitchen, then it must not create a desire to search for that user in the kitchen.
- The Filter function is responsible for three things. First, it should drop intentions that are not achievable. Secondly, it should retain intentions that could not be achieved. And thirdly, it should adopt new intentions.
- The Update function is responsible for updating the set of beliefs with new information.

The BDI architecture is implemented as a high level reasoning module that takes care of the high level reasoning and planning of the robot. This module receives information from other modules and is also able to control the state of these modules. For instance, the high level reasoning module can start the navigation module in order to go to a particular location, but also it could stop the navigation module when the robot is standing still doing some other task.

7 Implementation

An efficient implementation and integration of all the different functional software components requires a dedicated software framework. We have developed a module-based software framework, called the *Dynamic Module Library* that services this purpose. The basic software construct is that of a *module* that has input and output ports, which can be connected to each other to exchange data. The framework meets the following requirements

Runtime flexibility The possibility to change algorithms of modules at runtime, to extend the robot application with new modules at run-time, and to probe ingoing and outgoing data of modules at runtime.

Runtime robustness Stopping (or crashing) one or more modules of a running robot application should not result into an overall stopping (or crashing) of the robot application.

Runtime configurability The possibility to define at runtime the configuration of modules (that is, the connections between the modules) that make up the robot application.

Distribution The possibility to distribute the robot application over several host computers in order to enlarge the computational resources.

Portability Support for the most popular programming languages (C, C++, Java) and operating systems (Windows, Linux).

Modules are implemented separately from each other and are executed as individual processes in some operating system (both MS Windows or Linux are currently being supported). By using a registry, modules can discover and lookup each other at run-time. Once a module has found an other module, it can make connections to the ports of that other module. It is also possible to externally connect ports of modules. By means of the graphical user interface we can start/stop modules, connect/disconnect ports as well as probing ports. This way, a robot application consisting of modules can be configured at run-time which greatly enhances the interactivity of the development of the robot application.

In order to synchronize the tasks of all the modules, each module implements a particular task model. The model consists of five states in which a module can occur, namely `IDLE`, `RUNNING`, `PAUSED`, `SUCCEEDED` and `FAILED`. Each module can read as well as control the state of other modules. For instance, a Reasoning module can send an `execute` message to a Relocalization module to start this one up, and a `pause` message to a Navigation module to pause its task.

8 Experiments and Results

As far emotion generation is concerned, Figure 5 show some pictures of the head with different facial expressions. Although, an official evaluation is not yet done, the reaction of numerous people for which we demonstrated the robot clearly indicates that the robot appearance is appreciated and that the six basic emotions [11] of happiness, sadness, disgust, surprise, fear and anger are easily recognized.

The various moving parts of the robot's head are designed for low inertia as an adequate speed of response is considered as very important. The response speed of the lips is sufficient for realizing lip synchronous speech. Table 1 gives some figures of performance.

Demonstrations of the "Turn-to-speaker" behavior for various people have indicated that this behavior is highly appreciated: as soon as someone starts to speak the robot automatically turns to that person and smiles, no matter what is said. This clearly favors the natural interaction between the robot and the user.

9 Conclusions

This paper reported the results we have obtained during the on-going development of our domestic user-interface robot. To realize emotional feedback we have built a mechanical 3D head which is controlled by 17 standard RC-servo-motors. We showed pictures of this head expressing three of six basic emotional facial expressions. The robot is able to determine the position of the user localizing the origin of any person speaking near him. Additionally, the robot can gather information from the ambient intelligence in which it is assumed to operate and, in the other way around, it can redirect user commands to this environment.

The robot can localize himself in the environment using stereo images and the so-called appearance-based approach. This approach is appealing for its simplicity and due to the stereo vision less sensitive to change in illumination. On the basis of a proper localization, navigation is performed by using two modules: a planner and a behavioral execution module. The planner module calculates subgoal positions for

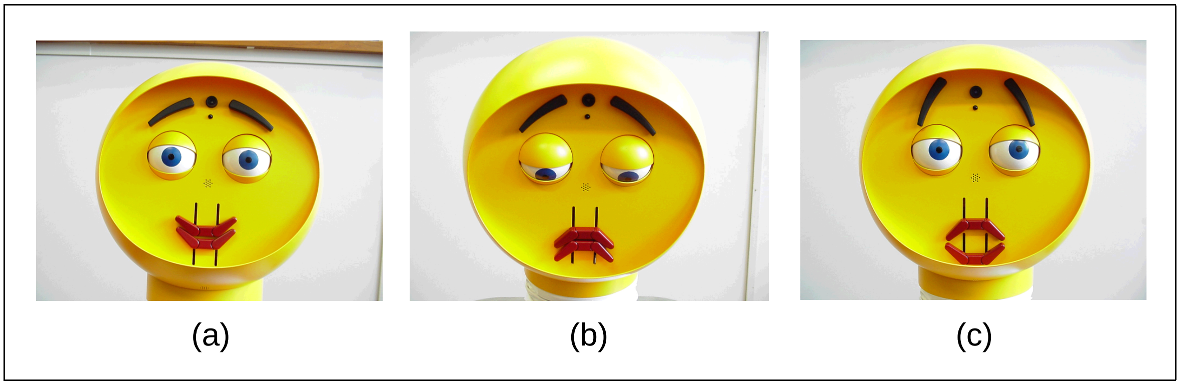


Figure 5: 3D mechanical head expressing emotions: a) happy, b) sad, c) surprised.

Component	**Action**	**Needed time [sec]**
Eyeball	Turn left/right/up/down	< 0.5
Eyelid	Open/close	< 0.1
Eyebrow	Up/Down	< 0.1
Mouth shape	Change of shape	< 0.2
Neck	Head turning left/right	< 1
Neck	Head up/down	< 1
Neck	Head approaching/withdrawing	< 1

Table 1: Reaction response of the different face components.

the behavioral execution module in order to prevent getting stuck by obstacles. The Wave Front Distance Field algorithm is used by the planner to calculate the subgoals.

All the modules of our robot are controlled and coordinated in a flexible way using a *Belief Desire Intention* architecture.

Finally, we presented our software development framework called the Dynamic Module Library. This framework is a state-of-the-art software tool to implement distributed robot applications. An application is runtime configurable by means of a graphical console: the robot application software modules can be probed, started, stopped, removed, added, and connected to each other on-line.

Our project represents a link between two *service to humans* paradigms: service robots and ambient intelligence. Hopefully, more fruitful cooperations would emerge between these two field in the next years.

References

[1] Aibo, http://www.aibo.com, 2002.

[2] R. C. Arkin, “Behaviour Based Robotics”, *MIT Press*, 1997.

[3] H.J.W. Belt and C.P. Janse, *Audio Processing Arrangement with Multiple Sources*, Patent application PHN 16638 EP-P, 1998.

[4] H.J.W. Belt and C.P. Janse, *Time delay Estimation from Impulse Responses*, Patent application PHN 017163.

[5] M.E. Bratman, D.J. Israel & M.E. Pollack, Plans and Resource-Bounded Practical Reasoning, *Computational Intelligence*, 4(4):349-355, 1988.

[6] A. Bruce, I. Nourbakhsh and R. Simmons, “The Role of Expressiveness and Attention in Human-Robot Interaction” in *Proceedings of the 2002 IEEE International Conference on Robotics and Automation Washington, DC, May 2002*, pp. 4138-4142, 2002.

[7] W. Burgard, D. Fox, and S. Thrun, “Active Mobile Robot Localization” in *Proceedings of the 15th International Joint Conference on Artificial Intelligence*, 1997.

[8] J. Cassell, “Embodied Conversational Agents: Representation and Intelligence in User Interface”, *AI magazine*.

[9] A. J. Davison, “Mobile Robot Navigation Using Active Vision” *PhD Dissertation*, Robotics Research Group, University of Oxford, 1999.

[10] A. J. Davison, M. Montemerlo, J. Pineau, N. Roy, S. Thrun and V. Verma, “Experiences with a Mobile Robotic Guide for the Elderly,” in *Proceedings of the AAAI National Conference on Artificial Intelligence* 2002.

[11] P. Ekman, *Darwin and facial expression : a century of research in review*, New York: Academic Press, 1973.

[12] P. Elinas, J. Hoey, D. Lahey, J. D. Montgomery, D. Murray, S.S. James and J. Little, “Waiting with José, a vision-based mobile robot” in *Proceedings of the 2002 IEEE International Conference on Robotics and Automation Washington, DC, May 2002*, pp. 3698-3705, 2002.

[13] ITEA Ambience project, http://www.extra. research.philips.com/euprojects/ambience/.

[14] B.J.A. Kröse, N. Vlassis, R. Bunschoten and Y. Motomura, “A probabilistic model for appearance-based robot localization”. *Image and Vision Computing*, 19(6):381-391, April 2001.

[15] B.J.A. Kröse, N. Vlassis and R. Bunschoten, “Omnidirectional vision for Appearance-based Robot Localization”, in *Lecture Notes in Computer Science*, pages 39-50. Springer, 2002.

[16] A. Ortony, A. Clore and G. Collins, *The Cognitive Structure of Emotions*. Cambridge University press, Cambridge, England, 1988.

[17] Papero, http: // www.incx.nec.co.jp/ robot/ PaPeRo/ english/ p_index.html. 2002.

[18] R. Picard, *Private Communication*, 2001.

[19] Pioneer2 Dxe, http://www.activrobots.com/ ROBOTS/p2dx.html

[20] J.M. Porta, B. Terwijn and B. Kröse, “Efficient Entropy-Based Action Selection for Appearance-Based Robot Localization”. In *Proc. IEEE Int. Conf. on Robotics and Automation*, Taipei, May 2003.

[21] S. Thrun, W. Burgard and D. Fox, “A probabilistic approach to concurrent mapping and localization for mobile robots” in *Machine Learning*, 31:29-53, 1998.

[22] S. Thrun, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hahnel, C.R. Rosenberg, N. Roy, J. Schulte and D. Schulz, “MINERVA: A Tour-Guide Robot that Learns”, {*KI*} - *Kunstliche Intelligenz*, pp. 14-26, 1999.

[23] J. Vandorpe, “Navigation Techniques for the mobile robot LiAS” *PhD Dissertation, PMA, Katholieke Universiteit Leuven*, 1997.

[24] N. Vlassis, B. Terwijn and B. Kröse, “Auxiliary particle filter robot localization from high-dimensional sensor observations”. In *Proc. IEEE Int. Conf. on Robotics and Automation*, Washington D.C., May 2002.

[25] M. Wooldridge, *Multi-Agent System*, John Wiley & Sons, 2002.