Robox at Expo.02: A large-scale installation of personal robots

Roland Siegwart*, Kai O. Arras, Samir Bouabdallah, Daniel Burnier,
Gilles Froidevaux, Xavier Greppin, Björn Jensen, Antoine Lorotte,
Laetitia Mayor, Mathieu Meisser, Roland Philippsen, Ralph Piguet,
Guy Ramel, Gregoire Terrien, Nicola Tomatis

Autonomous Systems Lab, Swiss Federal Institute of Technology Lausanne (EPFL), CH-1015 Lausanne, Switzerland

Abstract

In this paper we present Robox, a mobile robot designed for operation in a mass exhibition and the experience we made with its installation at the Swiss National Exhibition Expo.02. Robox is a fully autonomous mobile platform with unique multi-modal interaction capabilities, a novel approach to global localization using multiple Gaussian hypotheses, and a powerful obstacle avoidance. Eleven Robox ran for 12 hours daily from May 15 to October 20, 2002, traveling more than 3315 km and interacting with 686,000 visitors.
© 2003 Elsevier Science B.V. All rights reserved.

Keywords: Personal and service robot; Global localization; Obstacle avoidance; Human–robot interaction; Long-term experience; Tour-guide robot

1. Introduction

In this paper we adopt an experimental view of the problem. After the problem specification of mass exhibitions, we briefly present issues regarding mechanical design, software architecture and safety. Focus is then given to the multi-modal interaction system SOUL and the enhanced navigation system able to handle the highly dynamic environment of the exhibition. The paper concludes by presenting results of the robot's performance and the visitor's experience.

During its operation of 159 days from May 15 to October 20, 2002, the 11 Robox autonomously guided visitors through the *Robotics Pavilion* at the *Swiss National Exhibition Expo.02*, up to 12 hours per day,

seven days a week (Fig. 1). Throughout this period, 686,000 people were in contact with a personal robot for the first time in their life. The scale of the installation and the fully autonomous navigation and interaction of the robots make this project a milestone in mobile robotics research. It served as an extraordinary research platform and enables better understanding of the needs and concerns of the public in respect with the future of personal robot technology.

2. Related work

Progress in the application of estimation and decision theory combined with recent advances in sensor and computer technology enable today reliable navigation and interaction in highly dynamic real world environments. A limited number of researchers have addressed the challenge of navigation in exhibition-like environments such as museums [7,13,19,24,26].

* Corresponding author. Tel.: +41-21-693-38-50;
fax: +41-21-693-78-07.
E-mail address: roland.siegwart@epfl.ch (R. Siegwart).
URL: http://asl.epfl.ch

Fig. 1. (a) The autonomous exhibition robot Robox. (b) Robox No. 6 with visitors in the pavilion at Expo.02.

Rhino [7] and Minerva [24] were both successfully deployed in a museum during 1 or 2 weeks, respectively. Their task was to give guided tours to the visitors. Their localization method relied on raw laser range and vision data. Using raw data in a highly dynamic environment makes it hard to decide which sensor reading shall be taken for localization and which one is to be discarded. The filter which was used in [7,24] relied on the heuristic that range readings from dynamic obstacles are typically shorter than an expected range from the map. Unfortunately, the robot position must be known in order for this to work. Also, both robots were not autonomous with respect to computation, i.e. localization was running on off-board hardware requiring a working radio link.

Unlike these short-term projects, the autonomous tour-guide robot Sage was in action permanently in a museum with a total of more than half a year of successful operation [19,26]. For localization, the environment was substantially modified by adding artificial landmarks (color patches). The approach performs well but limits the movements of the robot to a predefined set of unidirectional safe routes, which guarantee constant landmark visibility. Obstacle avoidance just stops the robot such that it does not deviate from these routes.

A multi-robot installation, which is operational since March 2000, is presented in [13]. As a permanent part of a museum, three self-contained mobile robots have the task to welcome visitors, offer them exhibition-related information and entertain them. Their navigation area is restricted and very well structured. Localization uses segment features and a heuristic scheme for matching and pose estimation.

Finally, the 72-robot installation at the World Fair Expo 2000 in Hannover, Germany, was the first application of mobile robots in a mass exhibition. The vehicles, however, were very low-tech. Localized and controlled from external infrastructure, they served as freely moving swarm entities forming a huge interactive art installation during the 6 months of Expo 2000 (there is no publication to the knowledge of the authors).

3. Problem statement

The Swiss National Exhibition takes place around every 40 years. The recent edition, Expo.02, took place from May 15 to October 20, 2002. It was a major national happening with 37 exhibitions and a rich event program. The exhibition *Robotics* was intended to show the increasing closeness between man and robot technology (Fig. 2). The central visitor experience of Robotics was the interaction with 11 autonomous, freely navigating mobile robots (Fig. 1) on a surface of about $315 \, \text{m}^2$. Their main task was giving guided tours but included also a robot taking pictures of visitors. The exhibition was scheduled for 500 persons per hour. For this task, the main specifications can be summarized as follows:

- Navigation in unmodified, highly populated environment with visitors and other freely navigating robots.
- Bi-directional multi-modal interaction using easy-to-use, intuitive yet robot-typical interaction modalities. Speech output in four languages: French, German, Italian and English.
- Safety for visitors and robots at all time.
- Reliable operation during up to 12 hours per day, 7 days per week, during 5 months.
- Minimal manual intervention and supervision.
- Adaptive multi-robot coordination scenarios in function of the number of visitors and their interests.
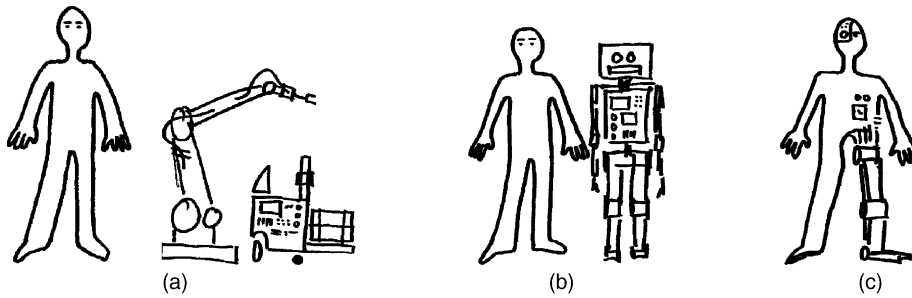- Control of visitor flow by the robots.

Fig. 2. Definition of robotics through the proximity and level of interaction between humans and robots: (a) industrial robotics, (b) personal robots and (c) cyborgs.

- Development of 11 robots within tight budgets and schedules.

After an outline in Section 4 on how these points have been addressed in the design of the robots, the remainder of this paper focuses on the most important competencies of Robox, which are the multi-modal interaction and navigation system. The paper concludes with results from the operational experience and interpretations of the robots' performance and the visitors' feedback.

4. The mobile robot Robox

The personal robot Robox is the result of an interdisciplinary project team involving engineers, exhibition makers, and industrial designers. The development of two prototypes started in January 2001 while the dedicated software development started in April 2002, only 13 months before the exhibition start. This short development time combined with a very tight development budget of around $ 400,000 defined the challenging pre-conditions of the project. Based on the Autonomous System Lab's experience in robot design, system integration and long-term experimentation [4,5], we concluded that building a robot from scratch [25] was the best way to fulfill the requirements regarding functionality and design.

4.1. Basic functionalities

The basic functionalities of the exhibition robot Robox are shown in Fig. 3. Its hardware consists of three main components: *mobility*, *interactivity* and *safety* (Fig. 4). The base of the robot contains the
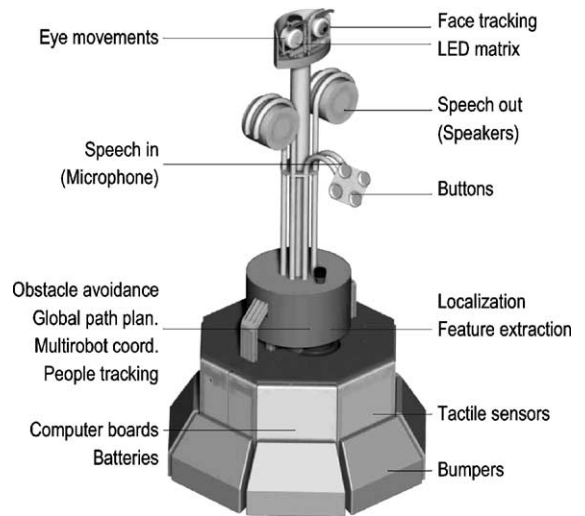

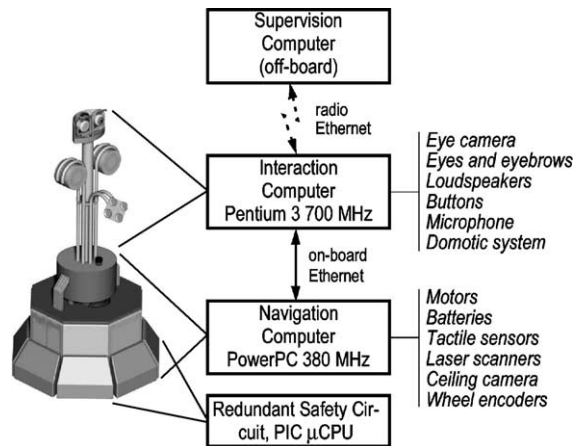Fig. 3. Functionalities of the exhibition robot Robox.

Fig. 4. Hardware architecture.

Compact-PCI controller, the two laser range finders (Sick LMS 200), the drive motors, the redundant safety circuit (see Fig. 4), the batteries, and the bumpers that contain tactile sensors. It is motorized by two differentially actuated wheels on the middle axis using 1:50 harmonic drives and a castor on the front and rear, one of them on a spring suspension. This gives Robox a good maneuverability and stability in spite of its height of 1.65 m. The battery pack gives the robot an autonomy of around 12 hours and requires 12 hours for recharging.

The upper part of the robot incorporates the interaction modalities (Fig. 3). The face includes two eyes with independently actuated pan-tilt units and mechanically coupled eyebrows. The right eye is equipped with a Firewire color camera for people tracking. The left eye contains a LED matrix to display symbols and icons. A high quality speech synthesizer enables the robot to speak in the four required languages. The central input device enabling a bi-directional communication with the visitors are four buttons which allow selecting the language, responding to questions the robot asks, and other types of actions. Two of the 11 Robox are additionally equipped with a directional microphone matrix for speech recognition. Although speaker-independent speech recognition in four languages in an exhibition-like context is not possible today, on-site experiments demonstrated the feasibility of recognizing simple words like *yes* and *no* in four languages.

4.2. Hardware architecture

The navigation software is considered safety-critical and is running on the hard real-time operating system XO/2 [5] installed on a PowerPC 750 (G3) at 380 MHz. The PowerPC interfaces the two laser scanners, a camera looking to the ceiling (via a Bt848-based frame grabber), the tactile sensors, the safety circuit and the drive motors/encoders (Fig. 4). It communicates with the interaction PC through the on-board Ethernet.

The interaction software, which is not considered safety-critical, is running on a Pentium III Compact-PCI board at 700 MHz under Windows 2000. This allows using standard software tools with a wide availability of drivers and libraries for vision, speech synthesis and speech recognition. The PC has

access to the Firewire eye camera, the controller for eye pan-tilts and eyebrows, the input buttons, the LED matrix, the microphone, and the two loudspeakers. Via a domotic system, the PC can furthermore interact with elements in the environment: switch on and off light spots, trigger flashes, start demonstrations or control other exhibits via infrared.

The robot is linked via radio Ethernet with an off-board PC that serves as a supervision and logging tool only. It displays important state variables of the robot at any time and stores relevant log information on a daily basis. However, Robox does not rely on this connection because it is always running from its on-board computers in fully autonomous mode.

4.3. Software architecture

As mentioned, the robot controller contains an Intel Pentium and a Motorola PowerPC board. The software was first designed on an abstract level. In a second step, functional units were assigned to one of the two computers accounting for three different criteria: hardware relation, safety and availability. For hardware related objects the choice was usually obvious (e.g. the Sick LMS 200 to the PowerPC). Safety and time-critical objects were assigned to the PowerPC due to the functional and temporal guarantees provided by the XO/2 operating system. Objects requiring commercial, off-the-shelf components have been implemented on the Windows-based machine because of their wider availability (e.g., speech out, Firewire eye camera).

The resulting software architecture is shown in Fig. 5. Watchdogs are implemented for the speed controller, the obstacle avoidance, the bumpers, and the laser scanner driver. Failure of one of these tasks is detected by the security controller, which attempts to restart the crashed task or, if the restart fails, stops the vehicle and sends an e-mail to the operator. The security controller furthermore takes care of the hardware watchdog, which is used by the PIC processor. A missing watchdog signal within two cycles (200 ms) means that either the security controller or the operating system is not running properly. In this case, the PIC circuit safely stops the whole system. Furthermore, this circuit ensures that the pre-defined maximal speed is never exceeded and that the PowerPC responds to each bumper contact. The central object SOUL [16] of the interaction subsystem that enables
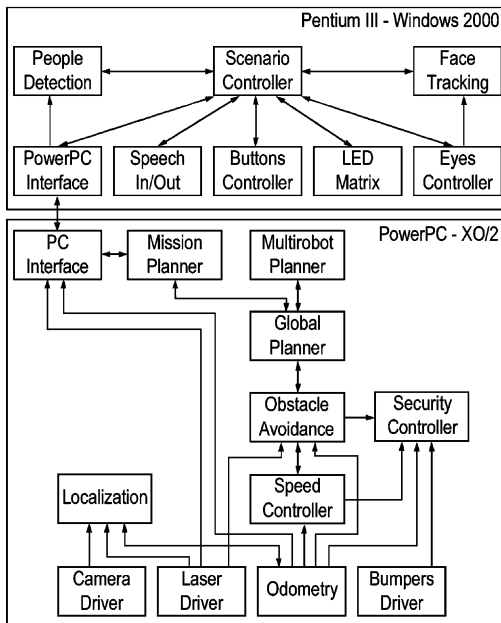
Fig. 5. Software architecture.

to create and control the scenarios is described in Section 6.

5. Navigation system

A mass exhibition imposes different challenges for the navigation system:

- Safety aspects are very important.
- The environment is cluttered with people and thus very dynamic.
- The visitors of the exhibition have no experience with mobile robots and thus their behaviors can be very strange and surprising.
- The robot has to fulfill its tour-guide task even if passing through the crowd is very difficult.

Furthermore, the navigation system has to be easily accessible by the scenario objects in order to execute different tours and interactions, and it has to run in real time on-board the robot.

In order to fulfill these requirements and specification, we combined graph-based global planning with feature-based localization, and an algorithm for obstacle avoidance combining the dynamic window ap-

proach (DWA) with an elastic band method. This results in a very reliable navigation system that is discussed in the following and presented in detail in [3].

5.1. Environment model

Our approach to environment modeling is feature-based using geometric primitives such as lines, segments and points (sometimes called landmarks). The environment topology is encoded in a weighted directed graph with nodes and edges between the nodes. Neither for global path planning nor for localization do we use a free space model like occupancy grids. The advantages of this choice are:

- *Compactness*. Indoor maps of this type (features plus graph) typically requires around 30 bytes/m^2. Further, scaling to 3D is polynomial, whereas grid maps scale exponentially.
- *Filtering*. The feature extractor automatically filters out measurement points not belonging to well-defined features.

The graph has two types of nodes: *station nodes* and *via nodes* (Fig. 6). Station nodes correspond to application-specific (x, y, θ) locations in the plane. Examples from Expo.02 include: showcase with industrial robot, tour welcome point, or location to trigger picture caption. Via nodes have two tasks. First, they correspond to topology-relevant locations like doors or corridor-crossings. Thereby the graph models the environment topology. Second, in environments with large open spaces, they can help global path planning by defining locations with favorable traversability. For example, this was used to optimize the visitor flow by planning paths to keep the robots far from each other.

The map further contains so-called *ghost points*. Ghost points are (x, y) positions in the world reference frame that act as invisible barriers. If the environment contains forbidden areas undetectable by the robot's sensors (e.g. staircases, glass doors, exits, etc.) ghost points are used to prevent the robot from this regions by injecting them into the sensor data as virtual readings (see also [11]).

The Expo.02 environment covered a surface of 315 m^2 and had 17 stations of interest for the robots. The map contained 44 line segments, 17 station nodes, 37 via nodes and 20 ghost points, as shown in Fig. 6. Its memory requirement was 8 KB, or 26 bytes/m^2.
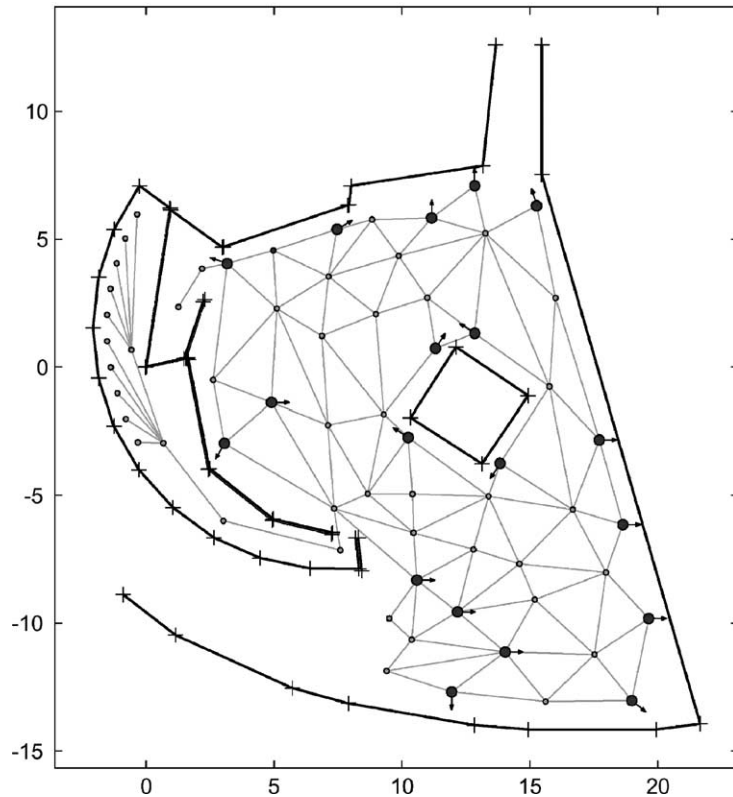
Fig. 6. The Expo.02 map with station nodes (black dots) and via nodes (small circles).

The ghost points (not shown in Fig. 6) are at the entrance (bottom of Fig. 6) and the exit (top) of the circulation area.

5.2. Local path planning and obstacle avoidance

Constraints that must be taken into account when moving Robox through the exhibition include the shape, kinematics, and dynamics of the robot on the one hand, and the environment on the other. The latter is highly dynamic in a mass exhibition, a fact that strongly influenced the design of obstacle avoidance on Robox. Based on previous experience [2] we developed a novel path planning and obstacle avoidance algorithm adapted to the challenging exhibition environment [20]. Because purely reactive obstacle avoidance methods tend to have problems with local minima, a path-planning layer is used to guide a reactive method [6,22].

Our sensor-based local path planner combines an adapted elastic band [21] with the NF1 navigation function [17]. Given a topologically correct but non-smooth path calculated via NF1, the elastic band efficiently iterates toward a smooth trajectory that adapts to changes in the environment. On Robox, elastic band iterations are implemented in a non-time critical thread running at several hertz. As soon as the elastic band "snaps", re-planning is performed in another dedicated thread. At Expo.02, this occurred approximately every 10 s. Fig. 7 illustrates such an event and Fig. 8 shows a typical situation at the exhibition.

For local path planning, the robot is assumed to be circular and holonomic, which can be justified by the octagonal shape of Robox and its ability to turn on the spot. Furthermore, heuristics are used to ignore certain laser readings (for instance, those that lie within the robot). This results in faster execution.
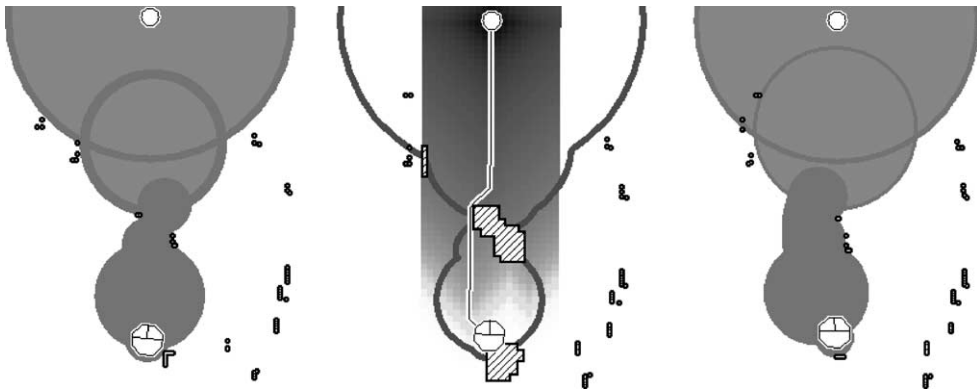
Fig. 7. Simulation screenshots (which show more information than available from the real robot) illustrate a typical replan sequence (from left to right). Laser range data are shown as white dots on black regions. The robot is the octagon, and the goal the white circle. At first, the elastic band is about to snap (gray disks, light gray indicates one of the heuristics). Robox continues to use the snapped band (second screenshot, gray curve) and calculates the NF1 in the background (the initial path is the thin black line outlined in white). The initial path becomes the new elastic band in the third picture.

The above simplifications are acceptable because the geometric, kinematic, and dynamic constraints are taken into account in the reactive obstacle avoidance layer, which is a modified version of the DWA [12]. Vehicle dynamics are modeled as maximum values for actuator speed and acceleration. Compared to the original DWA, our approach differs in the following points:

- Robox being a differential drive robot, the objective function (which trades off speed, heading, and
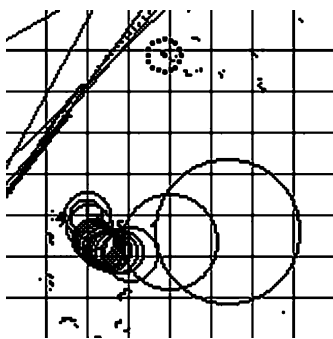

Fig. 8. A screenshot from Robox's supervision web-interface, taken during Expo.02. Such pictures are generated on the robot and help supervising its state. You can see laser data (small dots), lines from the a priori map and lines extracted from the range data, the elastic band (circles), and a ghost robot (large dots in the upper part of the image). The light gray grid shows the scale (resolution of 1 m).

clearance) is calculated in the actuator space (v_l, v_r) defined by the speed v of the left (l) and right (r) wheel. This models the physical acceleration limits of the vehicle more properly than using forward and rotational speed (v, w) in the Cartesian space.
- As in [22], robot shape is modeled as a polygon. It is not hard-coded but can be specified at boot time.
- Clearance is defined as time to collision (as opposed to distance). This resolves a singularity when turning on the spot, in which case any collision seems instantaneous because the wheel-axle mid-point does not move. Using time also causes the algorithm to choose more clearance at higher speeds.
- Virtual obstacles, such as forbidden zones or known invisible objects, are injected as ghost points into the range readings. They do not have to be treated specially by the reactive layer. This concept was also used to generate ghost robot contours at the position of the detected reflector mounted on each robot.[1]

Being part of the safety-critical software of the robot, the dynamic window runs at 10 Hz as a deadline-driven hard real-time task. Special attention

---

[1] Because all robots have their SICK sensors on the same height, they have difficulties seeing each other. This has been solved by adding two reflectors in the blind zone between the two SICK sensors.

was paid to optimize its execution time to be predictable and short. For reasons similar to those in [22], the clearance measure is pre-calculated and stored in a look-up table during startup. Refer to [20] for more technical details about the obstacle avoidance used on Robox.

5.3. Localization

Localizing a robot robustly in a mass exhibition environment is certainly a challenge. In former exhibition projects, localization was based on off-board hardware [7,24] or environment modifications [19]. In our earlier work we employed features and an *Extended Kalman Filter* (EKF) [4]. This is also the approach for the three museum robots described in [13]. However, a robot doing (single-hypothesis) pose tracking can lose its track as the inherent data association problem is ignored. With our new localization technique introduced in [1], we address the data association problem and extend the conventional EKF localization approach to a global localization technique.

Unlike POMDP or Markov approaches [7,24] where locations[2] are generated before they get evaluated by the exteroceptive sensors (as a grid or as particles), our approach to localization turns this process around: locations are generated as a direct consequence of sensory information. Features tell us *when* and *where* to place a location hypothesis. This allows us to always maintain as many hypotheses as necessary and as few as possible.

The technique for hypothesis generation is a constrained-based search in an interpretation tree [8,9,14]. This tree is spanned by all possible local-to-global associations, given a local map of observed features $L = \{l_i\}_{i=1}^{p}$ and a global map of model features $G = \{g_j\}_{j=1}^{m}$. Besides track formation, we have shown an algorithm for track splitting under geometric constraints. It relies on the same idea as hypothesis generation (search in an interpretation tree), forming thus a consistent framework for global EKF localization. It is beyond the scope of this paper to treat the framework in detail. We will therefore focus on the main ideas and results. Please refer to [1,8] for a more

complete presentation and Fig. 9 for an example of hypothesis generation in the exhibition environment.

5.3.1. Hypotheses generation

The search space for hypothesis generation is the space of all possible associations of the observed features l_i and the model features g_j. The space has the structure of a tree with p levels and $m + 1$ branches [14]. p is the number of observed features in L, m the number of model feature in G. The extra branch (called star branch) allows correct associations in the presence of outlier observations (false positives) and thus accounts for environment dynamics and map errors. During tree traversal, statistically feasible *pairings* $p_{ij} = \{l_j, g_j\}$ are sought given all uncertainties associated to the features. A pairing says that the observed feature l_i and the model feature g_j denote the same physical object in the environment (g_j is called an interpretation of l_i). Geometric constraints from the features are applied to the formation of pairings. They determine their statistical compatibility on a significance level α. Although the problem is of exponential complexity, the geometric constraints reduce enormously the space to be explored and can be classified into two categories:

1. *Location independent constraints*:
   - *Unary constraint*. We accept the pairing p_{ij} if l_i and g_j are of the same type, color, size or any other intrinsic property. Examples: the length of the observed segment l_i is equal (or smaller) than the length of the model segment g_j.
   - *Binary constraint*. Given a valid pairing p_{ij} we will accept the pairing p_{kl} only if the two local features l_i and l_k are compatible to the two global features g_j and g_l. Examples: l_i and l_k are lines with the intermediate angle φ_{ik}. Then, the pairing p_{kl} is considered compatible if $\varphi_{ik} = \varphi_{jl}$. With point features, for instance, the distances $l_i - l_k$ and $g_j - g_l$ must correspond.
2. *Location dependent constraints*: The above tests do not involve the robot position L_h. Once this is known, a further class of constraints can be applied.
   - *Visibility constraint*. This constraint only applies to features from the map. It tests whether g_j is visible from the robot position L_h. Example: lines or segments can be seen only from one

---

[2] We use the terms location, position and pose interchangeably. They denote all the full (x, y, θ) vehicle pose.
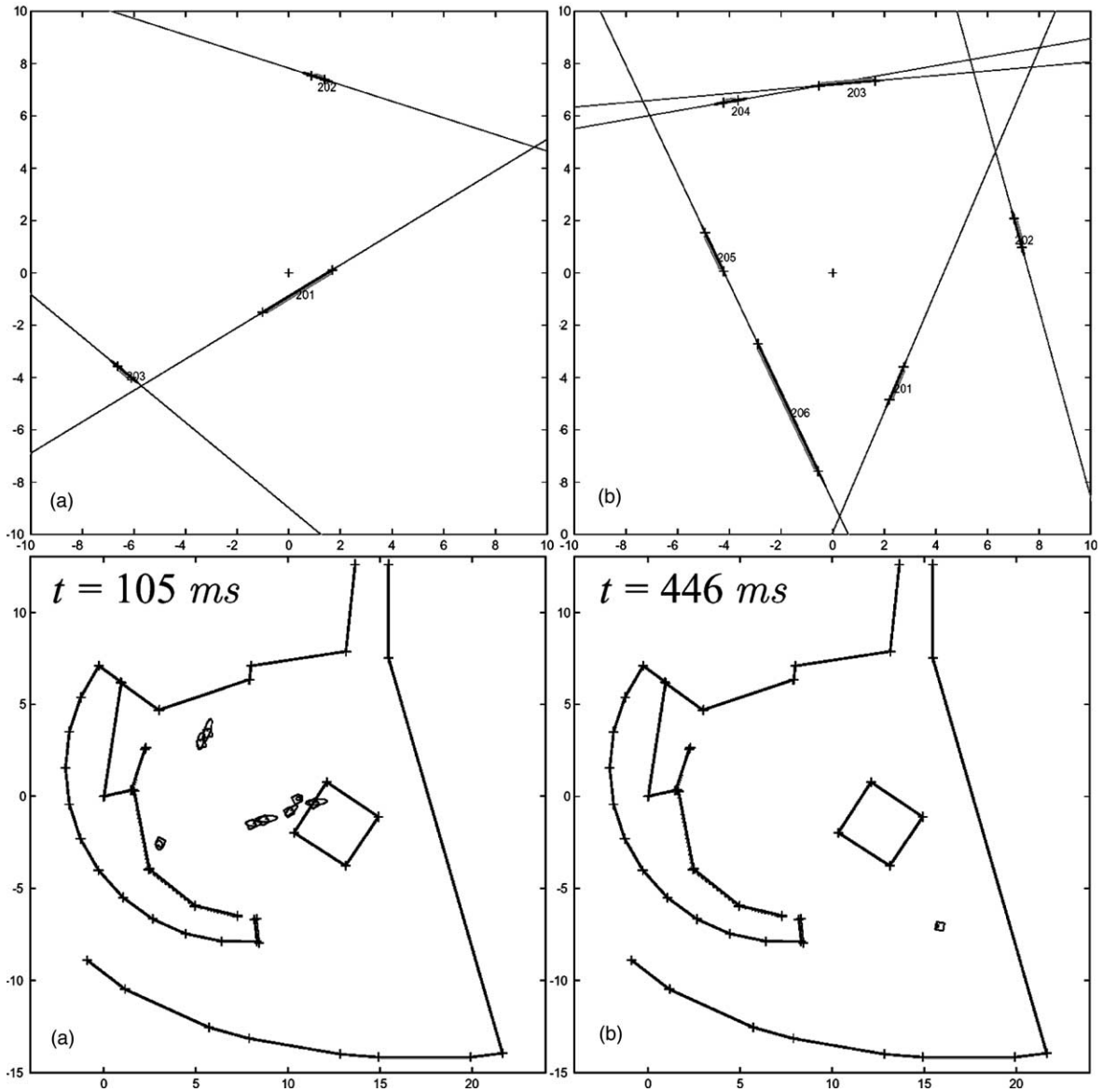
Fig. 9. Hypothesis generation. Given the local maps in (a) and (b), hypotheses are generated at locations where the local map 'fits' into the global map. In (a) there are 15 hypotheses (with their 95% error ellipse); the location is ambiguous. In (b) there is a single hypothesis; the robot is instantaneously localized. t denotes the execution time.

side. If the robot is behind a wall, one of the two lines modeling the wall is invisible. With sensor-specific parameters, the visibility constraint rejects features, which are not detectable, for instance, because they are farther than a maximal perception radius.

- *Rigidity constraint*. A pairing p_{ij} is considered compatible if l_i and g_j, transformed into the same coordinate system given L_h, coincide (are at the same position). This is what happens in the matching step of any EKF localization cycle. Usually, g_j is transformed into the frame of l_i.

- *Extension constraint.* A pairing p_{ij} is considered compatible if l_i and g_j, transformed into the same coordinate system given L_h, fully overlap. Example: an observed segment l_i must be completely contained in the transformed g_j seen from the location L_h.

5.3.2. The search algorithm

The constraints allow the discard of whole subspaces (subtrees) from the search each time an incompatible pairing is found at the root of such a subtree. With the uncertainties associated to local and global features, all decisions make use of the Mahalanobis distance on a significance level α.

Tree traversal is implemented as a recursive back-tracking search algorithm *generate_hypotheses* described in [1,8]. The strategy is to first find a minimal number of valid pairings with location independent constraints such that a location estimate can be determined in order to apply location dependent constraints, too. Each time when the algorithm reaches the bottom of the tree (that is, all observed features have been successfully assigned to a model feature or to the star branch) we have a valid robot location hypothesis. The pairings which support the hypothesis are put together in a *supporting set* $S_h = \{\{l_1, g_{j1}\}, \{l_2, g_{j2}\}, \ldots, \{l_p, g_{jp}\}\}$ and thereby constitute a location hypothesis $h = \{S_h, L_h\}$. All hypotheses together form the set of robot location hypotheses $H = \{h_i\}_{i=1}^n$.

5.3.3. Estimating the robot location from S_h

With the supporting set, the (x, y, θ) pose of the robot is not yet known. This is what the *Extended Information Filter* (EIF) does. Given a supporting set with all associated uncertainties, it estimates the robot location and its covariance in the least square sense. The difference between the EIF and the EKF is that the former is the batch estimator formulation of the latter (which is recursive). This is needed for hypothesis generation, because there is no a priori knowledge on the robot location which means formally that the state prediction covariance, usually called $P(k|k + 1)$, is infinite. With the EIF, this can be properly expressed as $P^{-1}(k + 1|k) = 0_{3\times3}$ since covariance matrices are represented in the information matrix form, that is, by their inverse.

6. Interaction and emotional expressions

Human–robot interaction with several robots in a public mass exhibition is a complex task. The goal of the Robox installation at Expo.02 was to allow the visitor a unique experience in sharing the environment with intelligent mobile robots and to interact with them. Furthermore, the robots have to guide the visitors around and explain part of the Robotics exhibition. So the robots are content (personal robot technology) and medium (tour guide) at the same time. On one hand, the interaction should be as intense and fascinating as possible in order to enable a rich visitor experience with the current state-of-the-art mobile robotics technology. On the other hand, as tour guides, the robots have to maximize the visitor and information flow, thus should restrict the visitor interaction. These contradictory goals imposed various restrictions on the exhibition and resulted in a continuous evolution of the scenarios throughout the duration of the exhibition.

The central object for controlling the interaction and the behavior of the robots is the *Scenario Object Utility Language* (SOUL) [16]. It is used to implement the different scenarios using all the input and output channels available on Robox.

6.1. SOUL

SOUL aims at composing the scenarios like a theater or a music composition. Through a convenient graphical interface it allows to combine different basic behaviors with synthesized speech, motion, sensory inputs and much more, and to supervise its execution (Fig. 10). The block diagram of the objects integrated in SOUL is shown in Fig. 11.

In general, we distinguish between static scenarios that are usually a fixed sequence of the tour, and dynamic scenarios that can be considered behaviors triggered by special events, e.g. if the visitors are blocking the way of the robot. If a dynamic scenario is triggered, SOUL will interrupt the current static scenario and execute a corresponding exception scenario telling the visitor that it is aware of his actions, before resuming the tour. In this sense, dynamic scenarios are more appealing for a lot of visitors, because they demonstrate awareness of the robot.

The main input and output channels available on Robox for composing the scenarios are:
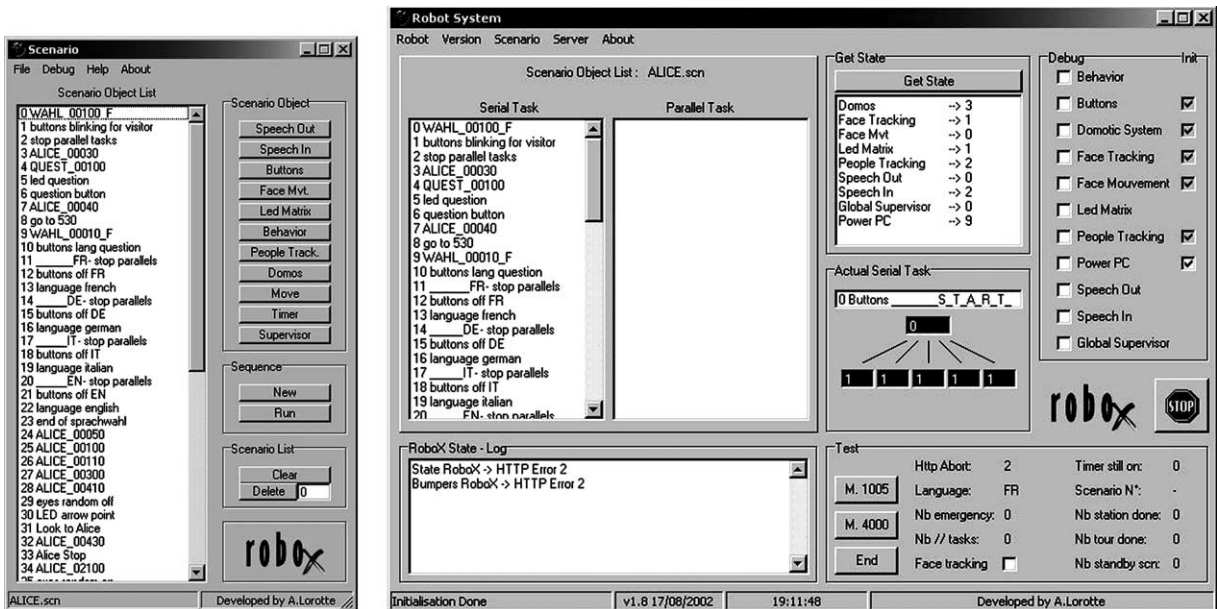
Fig. 10. Graphical interface of SOUL, allowing for convenient composition (left) and supervision (right) of scenarios.

- *Input channels*
  - ○ Robot location and state (goal reached/not reached).
  - ○ Blocked pathway (from obstacle avoidance module).
  - ○ Bumper contact (eight bumpers surrounding the robot).
  - ○ Four input buttons for visitor interaction.
  - ○ Face tracking [15,16] (through eye camera, Fig. 13).
  - ○ People-motion tracking [16] (through laser sensor, Fig. 14b).
  - ○ Speech input (only on two robots).
  - ○ Other robots' locations (through multi-robot coordination system).
  - ○ Emergency button.
  - ○ Battery status.
  - ○ Remote intervention through supervisor.
- *Output channels*
  - ○ Output of synthesized speech in English, French, German, and Italian using Mbrola [10] and LAIPTTS [23]. The pitch, rate and volume of the speech output can be selected directly in the SOUL interface.
  - ○ Robot motion (definition of next goal location, expressional motions).
  - ○ Illumination of four input buttons (e.g. green for "yes", red for "no", or different colors for language selection).
  - ○ Facial expressions composed using the eye and eyebrows motion and the LED matrix display (Fig. 12). The following seven basic expressions are used: sadness, disgust, joy, anger, surprise, fear, and calm [18].
  - ○ Direct use of icons and animations on the LED matrix display in the eye.
  - ○ Control of environment through domotic system.

6.2. People tracking

To detect visitors we relied on two sensors and dedicated algorithms. The laser scanner data is fed to a motion detector, indicating where in the robot's vicinity movement is present. Color images from the camera in the robot's left eye are used to identify skin color and subsequently human faces using heuristic filters.

6.2.1. Motion detection

Motion detection [16] distinguishes between moving and static elements of the environment. The detection of moving elements is possible because they change the range readings over time. During
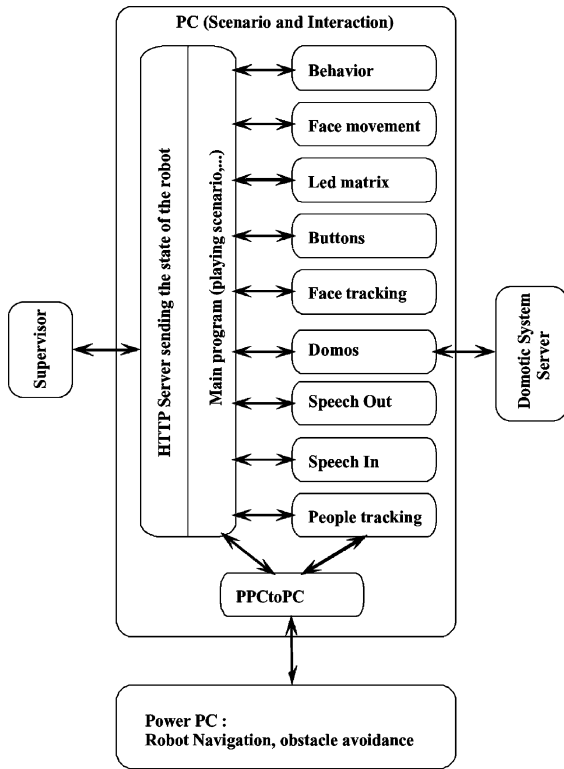
Fig. 11. Block diagram of the basic objects and connections integrated in SOUL.

initialization, we assume that the environment is static and convex. The range readings are integrated in the so-called static map, consisting of all currently visible non-moving elements. In contrast to grid-based approaches, we are working in the polar coordinates of the laser range sensor. For the available angular resolution from the laser (1° or 0.5°) this results in a polar map size of 360 or 720 elements, respectively.

In the next step, we compare the new information from the range finder with the static map. Assuming a

Fig. 12. Three facial expressions. From left to right: happy, surprised and angry.

Gaussian distribution of the sensor readings representing a given element, we use a χ^2 test to decide whether the current reading belongs to one of the elements of the static map or originates from another element, in which case the latter is then classified as dynamic.

All readings classified as static are used to update the static map. Readings labeled as dynamic are used to validate the map. Validation is performed as follows: if a reading labeled as dynamic is closer to the robot than the corresponding value from the static map, the latter persists. If it is farther away than the map value, it is used to update the map, but remains labeled as dynamic.

All dynamic elements are clustered according to their spatial location. Each cluster is assigned a unique ID and its center of gravity is computed.

The classification, update, and validation steps are repeated for every new scan. In case of robot motion, the process becomes slightly more complicated, since the static map has to be warped to the new robot position. This is possible knowing the displacement of the robot.

A result of the motion detection is shown in Fig. 14, where information from all active robots combined provides a snapshot of the exposition and the position of the robots and visitors. There are 140 motion elements approximating the number of visitors present. However, since the laser sensors are mounted at the height of the knees, some persons might be represented by two motion elements, each corresponding to one leg.

6.2.2. Face detection

Information gathered from face tracking is used in several parts. Together with the motion detection, it helps to verify the presence of visitors. Using visual servoing, the robot can look at the person it is interacting with. The main steps of face detection and tracking are [15,16]:

- *Skin color detection*. Among the different color spaces we chose the RGB space. Green and blue values are normalized using the red channel. This partially overcomes problems with in illumination changes. Fixed ranges for blue, green and brightness values are accepted as skin color. Erosion and dilation are performed on the resulting binary image to remove small regions.

Fig. 13. Visitors at the Expo.02 seen from the robot. Skin colored regions show a light border. Dark borders indicate clusters that passed the heuristic filters. The face tracking algorithm is able to maintain the visitor IDs even with a moving camera and resulting out of focus images.

- *Contour extraction and filtering*. The binary image is clustered and the contour of each cluster is extracted. We apply heuristic filters to suppress skin color regions that are not faces. These filters are based on rectangular areas, their aspect ratio, and the percentage of skin color. Another filter we apply is based on the morphology of the skin color region. We expect holes for eyes or the mouth within the region. This reduces the cases where the robot is looking at hands or other body parts.
- *Tracking*. The system tries to update the positions of already tracked regions based on the position of clusters in the current image. Clusters that remain unassigned to previous tracks are added and tracked until they leave the camera's field of view (Fig. 13).

6.3. Composition of scenarios

Fig. 14a depicts the layout of the exhibition. Presentation stations are defined near particular objects in the exhibitions. Presentation stations may comprise several goal nodes, as is the case for the welcome point, thus tours can start simultaneously. Fourteen presentation stations were located all over the exposition space (Fig. 14a). Finally, there are goodbye stations close to the exit. Each station corresponds to one scenario in the SOUL system, providing visitors with the necessary explanatory or entertaining information. Tours can be created by a succession of several presentation stations. Two stations *photo* and *poet robot*

are not included in any of the tours and are permanently occupied with a dedicated robot. The remaining nine tour-guide robots can reach all the other stations. In order to avoid having several robots presenting at the same goal node, each robot can ask the position of all the other robots at any time. This allows each robot to dynamically select the next goal, based on visitors' preferences and availability of the stations.

6.4. Control of the visitor flow

Expo.02 was a mass exhibition with several thousand visitors per day. The Robotics exhibition hosted around 500 visitors per hour that, in average, stayed around 15 minutes in the interactive zone of the exhibition. This results in around 125 visitors which are at the same time enjoying and interacting with the robots. In order to best serve the visitor and to control the visitor flow from the entry area to the exit, the tour starts at the welcome station and will guide the visitor always closer to the exit.

Visitor flow is channeled by two factors. First, the number of stations the robot visits before going to the goodbye station is limited to five. Second, the guided exposition tour will always lead visitors closer to the exit. This eases navigation and helps to maintain the visitor flow. Technically this is realized by a list of possible next presentation stations. Each presentation scenario has an individual list, containing only
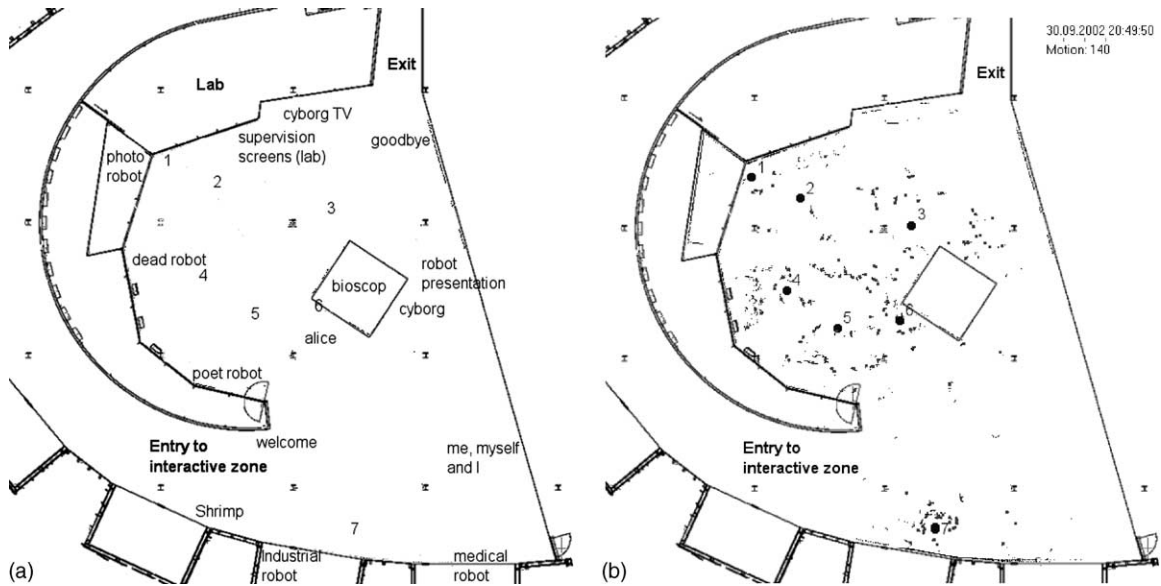
Fig. 14. (a) Scheme of the 315 m² exhibition area with the presentation stations of the tour. (b) Snapshot of the exposition with seven robots (black dots) and the visitors as (point-clusters) found by the motion detection algorithm.

stations that support the direction of the main visitor flow. When requesting exposition state from the global supervisor, the robot will seek only stations that it has not yet visited, and that are closer to the exit than it currently is. In order to further ensure proper handling of visitor density and flow, four basic exhibition modes were introduced:

- *Wait for Visitor*. If only few visitors are in the exhibition, the robots wait for one to come close enough before starting to talk and ask them which station they would like to see.
- *Visitor's Choice*. If the exhibition is at its regular visitor capacity, the robot will ask at each station whether the visitor wants to go to a next station.
- *Robot's Choice*. If the visitor density is very high, the robot will decide what the next station is and go there without asking.
- *No Move*. If too many visitors are in the exhibition area, the robot will not be able to move any more, and thus stays with one station and presents it permanently.

In order to run the *Visitor's Choice* mode, which was the regular case, the staff at the entrance of the exhibition controls the number of entering visitors.

7. Results

By October 20, 2002, after 159 days of operation, the 11 Robox accumulated:

- 13,313 hours of operational time,
- 9415 hours of motion,
- 3315 km travel distance,
- 686,000 visitors.

This is to our knowledge the biggest installation of interactive personal robots to date.

7.1. Reliability and performance

Starting with the opening of the exhibition on May 15, 2002, the robots operated fully autonomously, 7 days a week, 10.5 hours a day at the beginning, then 11 hours a day and 12 hours for the last 3 weeks. Due to some delays in the development, the software was still in the test phase during the first weeks of the exhibition. Therefore, we started with a *Mean Time Between Failure* (MTBF) of less than 1 hour. The MTBF excluding the first 4 weeks of the exhibition was 8.6 hours, which results in an average of one problem per robot per day. The large majority of failures were

caused by software problems in the interaction system (PC running Windows 2000) and were therefore not safety-critical. The navigation system (PowerPC running real-time system XO/2) had since the beginning a MTBF in the range 20–40 hours and stabilized to around 60 hours. The robot hardware, apart from some problems with the amplifiers for the drive motors, was highly reliable from the beginning. We did not encounter problems with serious vandalism. However, visitors seemed to be strongly attracted by the foam bumpers and the emergency button that were very frequently tested by them.

7.1.1. Safety

It was never observed that a robot was the cause of a dangerous situation (e.g. with small children, elderly or handicapped people). Collisions occurred but were typically provoked by the visitors themselves. The lack of additional sensors close to the floor (IR or ultrasonic), like the robot in [7], was bearable with the combination of tactile plates and foam bumpers. Blocked situations due to bumper contact could also be handled by the interaction system (robot expressing friendly menaces).

7.1.2. Performance of navigation system

The division of obstacle avoidance into a purely reactive part with high model fidelity and a planning part with local scope is a powerful conjunction. Very often, groups of visitors formed a U-shaped obstacle or left only small 'holes' for passage. The robots had no difficulty to escape from such situations, and due to the fact that we accounted for Robox true octagonal shape, narrow passages were efficiently used. Further, the elastic band generates good-looking trajectories. Our approach to the robot-sees-robot problem works in most situations. However, generating a ghost robot contour at the detected reflector position is an oversimplification in certain situations, as the ghost robot is larger than the real one, and robots sometimes stay mutually blocked even though there is sufficient space to maneuver. Such deadlocks occur most frequently when two robots approach each other head-on because obstacle speed is currently not taken into account in the obstacle avoidance method, and by the time the robots come to standstill, each perceives the other as being partly inside its outline.

The reliability of localization was a surprise in view of the environment dynamics and the fact that we used a horizontally scanning laser sensor only. A fallback solution with lamp features extracted by a camera looking to the ceiling was prepared but never used. However, lost situations occurred, their reasons being (in decreasing frequency): (i) pavilion staff members that push/rotate the robot in order to untangle congestions of visitors and robots, (ii) visitors imitating this behavior, (iii) unknown cause and (iv) failure of another component (hardware/software). Global localization as described in Section 5.3 was then very useful. Often it was possible to instantly relocalize within all people, enabling it to resume operation. The specific geometry of the Expo.02 environment was helpful here since it contained little symmetry.

Localization accuracy varies and depends on how much persons occlude the laser sensors and how much segments have been paired. Goal nodes were typically reached with a relative error less than or equal to 1 cm. This was measured by the traces on the floor at goal locations left by the robots after several months of operation. From a practicability point of view, localization was implemented as a non-real-time thread. Its cycle time was slowed down to 2 Hz in favor of other concurrent non-real-time processes (mainly communication). With a single hypothesis to be tracked, cycle times of 10 Hz and more are achievable on the Robox hardware.

Besides the specific qualities of our approach described so far, we believe that the use of geometric features for navigation is also an appropriate choice in highly dynamic environments. During feature extraction, sensor readings are sought which satisfy a spatial model assumption (e.g. from a line, a corner, or a ceiling light). Thereby, the extraction process acts as a filter saying which reading is to be taken for localization and which one is to be ignored. This filter relies typically on sound regression techniques and works independently on whether the robot is localized or not (as opposed to the 'distance filter' in [24]). A group of people standing around the robot does not produce evidence for the line extraction. Spurious segments, for example, on line-like objects carried by people, can occur and are treated by the star-branch in the localization algorithm.

7.1.3. Performance of interaction system

The final interaction software SOUL was operational since July 1, 2002. The exhibition mode *Visitor's Choice* was active approximately 95% of the time, the mode *Robot's Choice* during the remaining 5%. We experienced around 20 days with more than 5000 visitors. Even in this crowded environment, robots managed to move to their goal in a reasonable time, so that the mode *No Move* was never used. The mode *Wait for Visitor* was never used, since the robots were most of the time surrounded by interested visitors anyway. With five stand-by scenarios, robots ran out of those scenarios less than once a week.

Visitors stayed between 10 and 45 minutes with the robots. We tried to control this by changing the tour length from two to ten stations without noticing an impact on the visitor's stay. People just moved on to the next robot or even stayed with the current one. Here, enhanced environmental information, like motion information of the visitor or face recognition might help creating more convincing scenarios. We found that visitors quit a robot approximately after four stations, which was the actual tour length. The average number of visitors after installing the global supervisor (visitor flow control and robot coordination) rose slightly to around 4600 per day. This makes it hard to prove a quantitative effect on the visitor flow. However, observation of the crowd shows that visitors appreciated having the choice to go to a station. This adds a little interactive element to the tour.

7.1.4. Visitors' experience and feedback

The visitor's experience is in general very positive, with more than 83% of the visitors rating the exhibition as very good or good and less than 5% of the visitors rating the exhibition as bad. However, we encountered various problems with the first concept of the exhibition:

- Guiding visitors at public exhibition by a robot is somewhat difficult, because playing and interacting with the robot seems to be more attracting to the visitors. Some visitors were also not very patient and not willing to follow the instructions of the robot. However, even if a lot of people did not follow a whole tour of the robots, most of them had an exciting first contact with a personal robot.

- Due to the large number of visitors and robots sharing the exhibition hall, it was sometimes difficult to understand the artificial voice of the robot.

- The basic goal of the exhibition was to experience the increasing closeness between man and machine and not to present technical details of the robot. However, plenty of visitors were eager to get some insight on the robots. We therefore added a station at which the robot was presenting his functionalities with the help of a PowerPoint presentation.

- Some visitors were disappointed about the performance and intelligence of the robot. Their overrated expectations are mainly coming from science fiction movies and popular science presentations not really reflecting the state-of-the-art.

In order to quantify the visitors' appreciation and perception of the exhibition we made two public inquiries. A first questionnaire answered by 2043 visitors at the exit of the exhibition focused on the social impact and future of robotics technology in a large sense and is not discussed here. A second questionnaire filled in by 206 visitors concentrated on the general appreciation and quality of the interaction. Some principal results are summarized below:

- In a first question the visitors were asked to judge on the general experience, like amusement value or interactivity of the robot. Multiple answers were possible. 60% of the visitors judged the exhibition interesting and 40% amusing, and only 4% were bored by the exhibition. However, only about 12% of the visitors perceived the robots as really interactive. This might be due to the fact that we had to limit the interactivity in order to
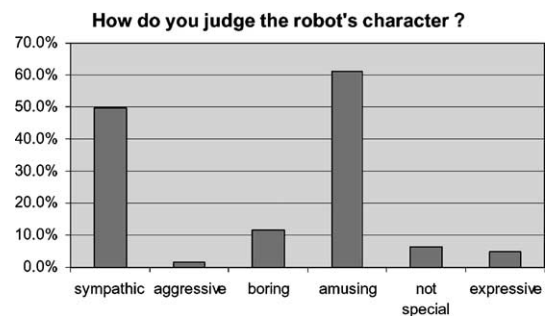

Fig. 15. Visitors' feedback on the robot's character.

reach a reasonable visitor flow. However, natural and easy to understand multi-modal interaction is still an open research issue and need further investigation.

- The functionality of the four input buttons was easily learned by 66% of the visitors through their interaction with the robot, 21% learned by imitation of other visitors. Only about 13% needed support from the staff or did not understand the functionalities at all.
- Around 75% of the people were able to follow the spoken explanations of the robot very well. This is also verified by the fact that 76% of the visitors learned from the robot that it is using a laser for navigation and detection of its environment.
- The appearance and character of the robot was appreciated very much by the visitors (see Fig. 15). This might also explain the fact that over 70% of

the people would not hesitate to ask the robot for information or help if it would offer his service in a supermarket or railway station.

8. Conclusion

The scale and duration of the Expo.02 project was a unique opportunity to validate techniques and to gather long-term experience on various levels. The framework presented in this paper meets all initial requirements. We described the basic hardware and software concept of Robox and discussed the navigation and interaction system. The results after 159 days of operation are very positive and prove the feasibility of the selected concepts, hardware, software and algorithms. The presented exhibition robot project with 11 fully autonomous robots, 5

Fig. 16. The robot doing its job in the exhibition. The bright spot on the robot is the reflector mounted so that the robots can see each other.

months of operation and 686,000 visitors enabled us to improve our current technology and to learn through an outstanding real-world research platform (Fig. 16).

Acknowledgements

References

[1] K.O. Arras, J.A. Castellanos, R. Siegwart, Feature-based multi-hypothesis localization and tracking for mobile robots using geometric constraints, in: Proceedings of the IEEE International Conference on Robotics and Automation, Washington, DC, USA, 2002.

[2] K.O. Arras, J. Persson, N. Tomatis, R. Siegwart, Real-time obstacle avoidance for polygonal robots with a reduced dynamic window, in: Proceedings of the IEEE International Conference on Robotics and Automation, Washington, DC, USA, 2002.

[3] K.O. Arras, R. Philippsen, N. Tomatis, M. de Battista, M. Schilt, R. Siegwart, A navigation framework for multiple mobile robots and its application at the Expo.02 exhibition, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Taiwan, 2003.

[4] K.O. Arras, N. Tomatis, B. Jensen, R. Siegwart, Multisensor on-the-fly localization: precision and reliability for applications, Robotics and Autonomous Systems 34 (2–3) (2001) 131–143.

[5] R. Brega, N. Tomatis, K.O. Arras, R. Siegwart, The need for autonomy and real-time in mobile robotics: a case study of XO/2 and Pygmalion, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000), Takamatsu, Japan, 2000.

[6] O. Brock, O. Khatib, High-speed navigation using the global dynamic window approach, in: Proceedings of the IEEE International Conference on Robotics and Automation, Detroit, MI, USA, 1999.

[7] W. Burgard, A.B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, S. Thrun, Experiences with an interactive museum tour-guide robot, Artificial Intelligence 114 (1999) 3–55.

[8] J.A. Castellanos, J.D. Tardos, Mobile Robot Localization and Map Building: A Multisensor Fusion Approach, Kluwer Academic Publishers, Dordrecht, 1999.

[9] M. Drumheller, Mobile robot localization using sonar, IEEE Transactions on PAMI 9 (2) (1987) 325–332.

[10] T. Dutoit, et al., The MBROLA project: towards a set of high-quality speech synthesizers free of use for non-commercial purposes, in: Proceedings of the ICSLP'96, 1996.

[11] D. Fox, W. Burgard, S. Thrun, A hybrid collision avoidance method for mobile robots, in: Proceedings of the IEEE International Conference on Robotics and Automation, Leuven, Belgium, 1998.

[12] D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance, IEEE Robotics and Automation Magazine 4 (1) (1997) 23–33.

[13] B. Graf, R.D. Schraft, et al., A mobile robot platform for assistance and entertainment, in: Proceedings of the International Symposium on Robotics, Montreal, Quebec, Canada, 2000.

[14] W.E.L. Grimson, T. Lozano-Pérez, Localizing overlapping parts by searching the interpretation tree, IEEE Transactions on PAMI 9 (4) (1987) 469–482.

[15] A. Hilti, I. Nourbakhsh, B. Jensen, R. Siegwart, Narrative-level visual interpretation of human motion for human–robot interaction, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, Hawaii, 2001.

[16] B. Jensen, G. Froidevaux, X. Greppin, A. Lorotte, L. Mayor, M. Meisser, G. Ramel, R. Siegwart, Visitor flow management using human–robot interaction at Expo.02, in: Proceedings of the Workshop on Robots in Exhibitions, IEEE/RSJ International Conference on Intelligent Robots and Systems, Lausanne, Switzerland, 2002.

[17] J.-C. Latombe, Robot Motion Planning, Kluwer Academic Publishers, Dordrecht, 1991.

[18] L. Mayor, B. Jensen, A. Lorotte, R. Siegwart, Improving the expressiveness of mobile robots, in: Proceedings of ROMAN-2002, Berlin, Germany, 2002.

[19] I. Nourbakhsh, J. Bodenage, S. Grange, R. Lutz, R. Meyer, A. Soto, An affective mobile robot educator with a full-time job, Artificial Intelligence 114 (1–2) (1999) 95–124.

[20] R. Philippsen, B. Jensen, R. Siegwart, Smooth and efficient obstacle avoidance for a tour guide robot, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Taiwan, 2003.

[21] S. Quinlan, O. Khatib, Elastic bands: connecting path planning and control, in: Proceedings of the IEEE International Conference on Robotics and Automation, Atlanta, GA, USA, 1993.

[22] C. Schlegel, Fast local obstacle avoidance under kinematic and dynamic constraints, in: Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Victoria, BC, Canada, 1998.

[23] B. Siebenhaar-Rölli, et al., Phonetic and timing considerations in a Swiss High German TTS system, in: E. Keller, G. Bailly, A. Monaghan, J. Terken, M. Huckvale (Eds.), Improvements in Speech Synthesis, Wiley, New York, 2001, pp. 165–175.

[24] S. Thrun, M. Beetz, M. Bennewitz, W. Burgard, A.B. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, D. Schulz, Probabilistic algorithms and the interactive museum tour-guide robot minerva, International Journal of Robotics Research 19 (11) (2000) 972–999.

[25] N. Tomatis, G. Terrien, R. Piguet, D. Burnier, S. Bouabdallah, R. Siegwart, Designing a secure and robust mobile interacting robot for the long term, in: Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Taiwan, 2003.

[26] T. Willeke, C. Kunz, I. Nourbakhsh, The history of the Mobot museum robot series: an evolutionary study, in: Proceedings of FLAIRS 2001, Key West, FL, 2001.

Roland Siegwart is a director of the Autonomous Systems Lab (ASL) at the Swiss Federal Institute of Technology Lausanne (EPFL). He received his Master's degree in Mechanical Engineering in 1983 and his Ph.D. in 1989 at the Swiss Federal Institute of Technology Zurich (ETH). In 1989/1990 he spent one year as postdoc at Stanford University. From 1991 to 1996 he worked part time as R&D director at MECOS Traxler AG and as a lecturer and deputy head at the Institute of Robotics, ETH. In 1996 he joined EPFL as full professor where he is working in robotics and mechatronics, namely mobile robot navigation, space robotics, human–robot interaction, all terrain locomotion and micro-robotics. He is a member of various scientific committees and cofounder of several spin-off companies.

Kai Oliver Arras is a Ph.D. student with the Autonomous Systems Lab (ASL) at the Swiss Federal Institute of Technology Lausanne (EPFL). He received his Master's degree in Electrical Engineering from the Swiss Federal Institute of Technology Zurich (ETHZ) in 1995 and worked as a research assistant in Nanorobotics at the Institute of Robotics in Zurich. In 1996 he joined Prof. Siegwart to help setting up the Autonomous Systems Lab at EPFL, where he is working on several aspects of mobile robotics. His fields of interest include system integration, feature extraction, local and global localization, SLAM, robot art, and robots in exhibitions.

Samir Bouabdallah is research assistant and Ph.D. student at the Autonomous Systems Lab (ASL) at the Swiss Federal Institute of Technology Lausanne (EPFL). He got his Master's degree in Electrical Engineering from Abu Bakr Belkaid University (ABBU), Tlemcen, Algeria, in 2001. His master thesis was the development of an autonomous mobile robot for academic research. His current research interests are control systems for all terrain locomotion and flying robots.

Daniel Burnier is manager of the electronics workshop of the Autonomous Systems Lab (ASL) at the Swiss Federal Institute of Technology Lausanne (EPFL). He received his Diploma in Electrical Engineering from the University of Applied Science of Yverdon in 1986. Until 1988 he worked at the same university. In 1988 he joined the company CYBELEC where he worked as R&D engineer on the digital control of machine tools. After 5 years as medical auxiliary he joined the Autonomous Systems Lab in 2000, where he is involved in the development of mobile robots and manages various development projects.

Gilles Froidevaux studied Microengineering at the University of Neuchâtel and at the Swiss Federal Institute of Technology Lausanne (EPFL) and received his Master's degree in 2001. His master thesis was the mechanical design and control of a robot arm. In spring 2001, he joined the Expo.02 robotics team at EPFL to work on the human–robot interaction with a main focus on face tracking. After the successful completion of the Expo.02 project he founded the engineering company FiveCo—Innovative Engineering together with colleagues of the project team.

Xavier Greppin received his Master's degree in Microengineering from the Swiss Federal Institute of Technology Lausanne (EPFL) in 2001. During his master thesis he developed a small mobile robot with wireless control. In spring 2001 he joined the Expo.02 robotics team at EPFL to work on the human–robot interaction with a main focus on the design of realization of the face and its control. After the successful completion of the Expo.02 project he founded the engineering company FiveCo—Innovative Engineering together with colleagues of the project team.

Björn Jensen is a Ph.D. student at the Autonomous Systems Lab (ASL) at the Swiss Federal Institute of Technology Lausanne (EPFL). He received his Master's degree in Electrical Engineering and Business Administration from the Technical University of Darmstadt, Germany in 1999. His main interest is in enhancing man–machine communication using probabilistic algorithms for feature extraction, data association, tracking and scene interpretation.

Antoine Lorotte decided to leave Paris to study at the Swiss Federal Institute of Technology Lausanne (EPFL), where he got his Master's degree in Microengineering in 2001. During his masters project at the Autonomous System Lab (ASL) he developed a complete system of home automation (domotic). In spring 2001 he joined the Expo.02 robotics team at EPFL to work on the human–robot interaction with a main focus on the Scenario Object Utility Language used on the Robox. After the successful completion of the Expo.02 project he founded the engineering company FiveCo—Innovative Engineering together with colleagues of the project team.

Laetitia Mayor studied at EPFL and Carnegie Mellon University and received her Master's degree in Microengineering from the Swiss Federal Institute of Technology Lausanne (EPFL) in 2002. In her master thesis she developed a concept for emotional human–robot interaction. In spring 2002 she joined the Expo.02 robotics team at EPFL to work on emotional human–robot interaction and the development of scenarios. After the successful completion of the Expo.02 project she joined the company SYNOVA, specialized in laser cutting.

Mathieu Meisser was born in Ticino, southern Switzerland. After college he decided to move to Lausanne to study Microengineering and received his Master's degree in 2001. In his master project he developed a small mobile robot with wireless control. In spring 2001 he joined the Expo.02 robotics team at EPFL to work on the human–robot interaction with a main focus on the speech synthesis. After the successful completion of the Expo.02 project he founded the engineering company FiveCo—Innovative Engineering together with colleagues of the project team.

Roland Philippsen received his Master's degree in Microengineering in 2000 at the Swiss Federal Institute of Technology Lausanne (EPFL). During his master thesis at Stanford University he analyzed the material an information flow through the partially automated DNA sequencing. He is currently working on his Ph.D. in path planning and obstacle avoidance for mobile indoor robots at the Autonomous Systems Lab (ASL).

Ralph Piguet is the head of the electronics workshop of the Autonomous Systems Lab (ASL) at the Swiss Federal Institute of Technology Lausanne (EPFL) since 1997. He received his Diploma in Electrical Engineering from the University of Applied Science of Lausanne in 1996. From 1992 to 1997 he worked with Bobst SA in Lausanne. During this period, he was involved in the development of imaging devices and quality control systems as well as with the training of apprentices. Since 1997 he is involved in various developments of mobile robots. He is also executive board member of the spin-off company BlueBotics that realized the exhibition robots.

Guy Ramel is native of Geneva. After finishing the technical school in electronic and the Engineer School of Geneva (EIG) he studied physics at the Swiss Federal Institute of Technology Lausanne (EPFL). He received this diploma in physics (2001) with a work in astrophysics and an engineer project in neural networks. He is actually pursuing a Ph.D. in the field of multi-modal interaction and human–robot interactions at the Autonomous Systems Lab (ASL) at the Swiss Federal Institute of Technology Lausanne (EPFL).

Gregoire Terrien received his Master's degree in Microengineering in 2000 from the Swiss Federal Institute of Technology Lausanne (EPFL). He did his master thesis at Carnegie Mellon University (CMU) in the field of mobile robots teleoperation based on sensor fusion. After that he started to worked a the Autonomous Systems Lab (ASL) at EPFL, and since 2001 he holds also a part-time position as executive board member with BlueBotics SA, a spin-off company of the ASL. He is involved in different projects from mechanical and electrical design to low-level and high-level programming.
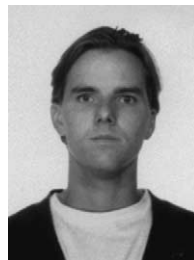
Nicola Tomatis received his Master's degree in Computer Science in 1998 from the Swiss Federal Institute of Technology Zurich (ETH). After this, he moved to the Swiss Federal Institute of Technology Lausanne (EPFL), where he received his Ph.D. in 2001. His research covered metric and topological (hybrid) mobile robot navigation, computer vision and sensor data fusion. Since autumn 2001 he holds a part time position as senior researcher with the Autonomous Systems Lab, and is executive board member with BlueBotics SA spin-off, which is involved in mobile robotics.