In *The IEEE-RAS International Conference on Humanoid Robots* (Humanoids-09),
Paris, France, December 7-10 2009.

# Learning Sensor Models for Autonomous Information Fusion on a Humanoid Robot

Mohan Sridharan
Department of Computer Science
Texas Tech University, USA
mohan.sridharan@ttu.edu

Xiang Li
Department of Computer Science
Texas Tech University, USA
xiang.li@ttu.edu

***Abstract*— Mobile robots equipped with multiple sensors are increasingly being used in specific real-world applications [1], [2], [3], [4], primarily because of the ready availability of high-fidelity sensors [5], [6]. A robot equipped with multiple sensors, however, obtains information about different regions of the scene, in different formats and with varying levels of uncertainty. One open challenge to the widespread deployment of robots is the ability to fully utilize the information obtained from each sensor in order to operate robustly in dynamic environments. This paper presents a probabilistic approach for autonomous multisensor information fusion on a humanoid robot. The robot exploits the known structure of the environment to autonomously model the expected performance of the individual information processing schemes. The learned models are used to effectively merge the available information. As a result, the robot is able to localize mobile obstacles in its environment. The algorithm is fully implemented and tested on a physical robot platform.**

## I. Introduction

The ready availability of high-fidelity sensors at moderate costs [5], [6] has resulted in the deployment of mobile robots in specific real-world applications [1], [2], [3], [4], [7]. The widespread deployment of mobile robots equipped with multiple sensors requires the ability to operate autonomously in dynamic environments by adapting to environmental changes. There has been significant research on autonomous learning from sensory input [2], [4], [8], and processing management on mobile robots [1], [9]. However, the ability to autonomously exploit the complementary properties of the sensors to effectively merge and exploit the available information, is still lacking. Each sensor mounted on a robot typically provides information on different regions of the scene, with varying levels of uncertainty. The visual input from a color camera, for instance, is a low-cost and high-bandwidth source of information as compared to the range input from a laser range finder. Visual input is however more noisy and the visual information processing algorithms are typically computationally expensive. The problem is more pronounced on humanoid robot platforms, where the sophisticated algorithms for motion control and balance [10], [11] make it a challenge to efficiently merge the information obtained from the different sources.

Information fusion on mobile robots has been extensively researched in a range of applications [2], [12], [13], [14]. However, a major shortcoming of existing methods is that manually encoded heuristic constraints specify when and how the information obtained from each sensor is given precedence. In the DARPA grand challenges, for instance, the decision-making was primarily based on range and GPS information, while the visual input was predominantly used for only close-range obstacle avoidance [2], [4]. Such an approach that does not utilize all the available information, is likely to be at a disadvantage in dynamic environments.

This paper advocates a probabilistic approach to effectively merge the information obtained from multiple sensors, and describes an instance of this approach for the task of detecting and localizing mobile obstacles in a humanoid robot's environment. It makes the following significant contributions: (a) an approach that enables a robot to use the environmental structure to model and hence predict the performance of algorithms that process the sensory inputs, and (b) a probabilistic approach that uses the learned models to robustly combine the information obtained from the different sources. Furthermore, the robot is able to better exploit the rich information encoded in color camera images. All algorithms are fully implemented and tested on a humanoid robot platform (Aldebaran Naos [15]).

The remainder of the paper is organized as follows. Section II describes the test domain and the proposed approach, while Section III describes the experimental results. Section IV provides a brief overview of some related methods, and the paper concludes with Section V.

## II. Test Platform and Proposed Approach

This section first describes the experimental domain chosen to evaluate the proposed approach. This is followed by a description of the available information sources, the challenge task and the algorithm that effectively merges the available information in order to address the challenge task.

### A. Test Platform

The humanoid robot platform used in our experiments is the Aldebaran Nao [15], a 58cm tall robot with 23 degrees of freedom—five in each arm and leg, two in the head, and one at the pelvis. The primary sensors are the monocular color cameras in the forehead and the nose. Only one camera can be used at a time i.e. stereo capabilities do not exist. Each camera has a $58^o$ diagonal field of view, and provides images at a maximum resolution of $640 \times 480$—the $320 \times 240$ or the $160 \times 120$ images can be used for faster processing. There are two ultrasound sensors in the chest, one each on the left and the right with a $60^o$ field of view. The robot is also

equipped with accelerometers, bump sensors, microphones, loudspeakers, LEDs and Wi-Fi for communicating with other robots or an off-board PC. However, all processing for vision, locomotion, localization and team coordination is performed in real-time (30Hz) on board the robot, using the x86 AMD GEODE 500MHz CPU that runs embedded Linux.

A standard application scenario for the Naos is RoboCup, an international research initiative with the stated goal of creating, by the year 2050, a team of humanoid robots that can beat the champion human team in a game of soccer on an outdoor soccer field [16]. The Standard Platform League [17]

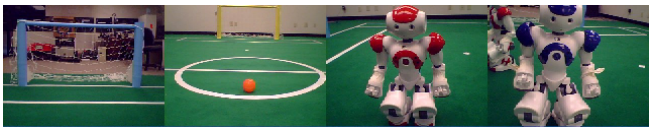

Fig. 1: Images of the Nao [15] and the soccer field.

of RoboCup has a team of humanoid robots (three per team) playing a competitive game of soccer on a $6m \times 4m$ indoor soccer field. Figure 1 shows some images of the Nao and the soccer field. The robot soccer framework is a good test platform because it presents many of the challenges that need to be addressed for deploying humanoid robots in the real-world (e.g. autonomous vision, motion, coordination). At the same time, the framework provides a moderate amount of structure that makes the domain tractable to solutions.

### B. Proposed Approach

The goal is to exploit the available information and effectively fuse the information obtained by the individual sensor processing schemes. The processing schemes under consideration are:

- *Ultrasound (US)*: each ultrasound sensor provides a reading of object distance within a $60^o$ cone, up to a maximum distance of $\approx$ 150cm. The bearing information is limited to object presence on the left and/or the right.
- *Vision–Color (VC)*: Since many objects in the domain (robots, goals, field etc) are color-coded, color segmented image regions are used as a source of information.
- *Vision–SIFT (VS)*: In order to extract maximum information from the images, the popular SIFT (Scale Invariant Feature Transform) algorithm [18] is used to characterize objects using image gradient features.

The challenge task in this paper is to localize the mobile obstacles in the environment, i.e. to compute the distance and bearing of the obstacles relative to the robot—bearing is the angle with respect to the axis pointing straight ahead. In the robot soccer domain, the major "obstacles" are the other robots (opponents and teammates). Collision with other robots can cause physical damage and provide the opponents with an advantage (the rules of the game penalize robots that collide with each other). Teammates are considered obstacles despite the Wi-Fi communication because the communication is typically delayed and noisy.

Each robot has a uniform of a specific color—robots in one team are *red* while those on the other team are *blue*. As seen in Figure 1, the uniform is characterized by four large regions (head, shoulders, chest) and being able to see at least three of these regions arranged in a specific pattern can be used to detect a robot. However, such a pattern can be seen only from specific viewpoints, and up to a distance of $\approx$ 2m. In addition, since the robot uses only one camera at a time the distance to the object is computed by analytically comparing the known object size and the size of the detected pattern (in pixels) in the image. Segmentation errors can affect the size of the detected uniform pattern and introduce noise in the computed distances. The bearing values are based on the offset of the detected pattern from the image center and are more robust to segmentation errors. The SIFT algorithm, on the other hand, characterizes objects

(a) (b)

Fig. 2: (a)-(b) Images with some gradient features superimposed.

of interest by image gradient features that are known to be robust to scale, orientation and illumination changes [18]. Given such feature representations of the target object (in this case robots), recognition in test images can be achieved up to a distance of $\approx$ 4m. Though the bearing can be computed based on image offsets, the distance cannot be computed accurately because the gradient features are not arranged in a unique pattern. Figure 2 shows images with the gradient features superimposed.

| Scheme | Distance | | Bearing | |
|---|---|---|---|---|
| | FOV (cm) | Accuracy | FOV | Accuracy |
| Ultrasound (US) | 20 to 150 | high | $80^o$ | low |
| Vision-Color (VC) | 20 to $\approx$ 200 | medium | $190^o$ | high |
| Vision-SIFT (VS) | 20 to $\approx$ 400 | low | $190^o$ | high |

TABLE I: The field of view (FOV) and accuracy of distance and bearing computed with the processing schemes. Vision has a larger FOV because the robot pans its head while moving forward.

Table I shows that the three processing schemes have complementary characteristics. Typically, heuristic constraints are imposed (manually) on when and how the information from each of these sources should be used. Instead, we enable the robot to learn models that can predict the performance of the individual schemes. Section III describes how the models of the expected error in each processing scheme, and the SIFT model of the target objects, can be learned. Once these models have been learned, the information form the processing schemes can be merged using the algorithm summarized in Algorithm 1.

Since the obstacle estimates need to be maintained across a sequence of frames (time steps), each estimate is associated with a Kalman filter [19]. The first step is the "time update"

of the Kalman filters ($UpdateExistingEstimates()$, line 2), which adjusts the existing estimates to account for the robot's motion since the previous update. It also removes estimates that correspond to obstacles that have not been seen for some time. Each processing scheme is then used to compute the distances and bearings to the obstacles ($CurrentObstacles()$, lines 3–5). The $US$ scheme can only provide limited bearing information (left, right or both), and the $VS$ scheme cannot provide distances to obstacles.

**Algorithm 1** Multisensor Information Merging

**Require:** Learned models that predict the error in range distance and bearing for measurements from each information source.

**Require:** Learned SIFT model of the target object (in this case, robots).

1: **repeat**
2: $UpdateExistingEstimates()$
3: $\{d_{us}, dir\} = CurrentObstacles_{us}()$
4: $\{d_c, \theta_c\} = CurrentObstacles_{vc}()$
5: $\{\theta_s\} = CurrentObstacles_{vs}()$
6: $ResolveCurrentEstimates()$
7: $MergeWithExistingEstimates()$
8: **until** end of the game

The next step ($ResolveCurrentEstimates()$, line 6) first groups the individual distances and bearings from the current frame—for instance, the distances and bearings obtained from $US$ are grouped with similar values computed using $VC$. In the case of processing schemes that provide partial information (e.g. $VS$ only provides bearing), grouping is done based on the available information. This grouping is accomplished using the expected errors in the measured values. For instance, if the difference between the bearing computed using $VC$ and the bearing computed using $VS$ is more than the expected error in the individual measurements, they are not grouped together. This "threshold" can be tuned to detect obstacles at different resolutions, and more sophisticated data association can be performed [20] if required. After the grouping, values within a group are merged to obtain an estimate for each obstacle in the current frame:

$$d^j = \sum_i w^j_{d,i} d^j_i \qquad (1)$$
$$\theta^j = \sum_i w^j_{\theta,i} \theta^j_i$$

where $w_{d,i}$ and $w_{\theta,i}$ are the weights associated with the distances and bearings obtained from the $i$th source ($US$, $VC$, $VS$). The merged distance and bearing to the $j$th obstacle in the current frame ($d^j$ and $\theta^j$) are the weighted averages of the values from the individual schemes. However, $VS$ cannot measure distance, and $US$ provides limited bearing:

$$\theta^j = w_{\theta,us}\{w^j_{\theta,vc}\theta^j_{vc} + w^j_{\theta,vs}\theta^j_{vs}\} \qquad (2)$$
$$w_{d,vs} = 0, \quad w_{\theta,us} = \begin{cases} -1 & \text{if only right US triggers} \\ +1 & \text{otherwise} \end{cases}$$

The weights represent the degree of trust associated with each processing scheme. They are obtained by normalizing the certainty associated with each scheme:

$$w_i = \frac{p_i I_i}{\sum_j p_j I_j} \qquad (3)$$
$$p_{d,i} \propto 1/f_{d,i}, \quad p_{\theta,i} \propto 1/f_{\theta,i}$$
$$f(x) = a_0 + \sum_{k=1}^{N} a_k x^k : N \in [1,3],\ x = d \text{ or } \theta$$

where $I$ is the indicator function, and $p_i \in [0,1]$ is the certainty associated with a measurement from the $i$th source. The certainty depends on the expected error, which in turn is a function of the measured distance or bearing. It is also possible to use a joint function of distance and bearing and learn the associated parameters.

The grouping process used in line 6 is also used to match the estimates from the current frame with the estimates from prior frames. The matched estimates are merged and the new estimates are added during the "measurement-update" of the Kalman filters ($MergeWithExistingEstimates()$, line 7). The prior estimates without matches in the current frame are retained until they are removed during a subsequent "time update" (line 2). The measured values can be input to the Kalman filters without the initial merging (Equation 1), but the measurements would still need to be matched with the prior estimates. In addition, using the estimated measurement errors in the Kalman filter noise models requires manual tuning and does not provide the desired accuracy.

The experiments below focus on localizing mobile obstacles. However, even when the tasks and processing schemes are different, learned models for the individual processing schemes and target object representations can be used to perform information fusion robustly.

## III. Experimental Setup and Results

This section describes the approach to learn the models required in Algorithm 1, and then describes the experiments conducted to evaluate the proposed approach.

### A. Error Models and Visual Representation

The vision system on the humanoid robot follows a specific sequence: input images are color segmented using a *color map* that maps image pixels to numerical color labels. Contiguous segmented pixels of the same color are grouped into regions that are used to detect objects based on heuristic constraints—see [13] for details.

The information obtained from the processing schemes can be merged autonomously if it were possible to predict the error in the measurements obtained from the individual schemes—a measurement with a lower error is assigned a larger weight in Equation 3. Most mobile robot environments have a moderate amount of structure, which can be used to automate tasks that usually require manual supervision. The robot soccer domain has color-coded objects at known positions. Based on prior work where this knowledge was exploited to learn the color map autonomously [8], we enable the robot to learn the required error models.

Obstacles are placed at fixed positions on the field that are known to the robot. The robot uses the cues from the

visual processing sequence described above to move through a sequence of poses (position+orientation) that it can reach with high accuracy, for instance points on the center line of the field. At each such pose, the robot compares the actual distance and bearing values against the measured values to compute the measurement errors. The error values are collected and used to train a function approximator that models the measurement error as a function of the measured distance (or bearing), and computes $p \in [0, 1]$ as the certainty of the measured values (Equation 3). Polynomial regression functions are used to approximate these errors (Equation 3), and the parameters of these functions (degree, coefficients) are learned using the collected statistics. Similar performance is achieved using more popular function approximators (e.g. neural networks [21]) but the polynomial functions require fewer samples for parameter estimation.

At each pose, the robot also projects on to the image the known positions of the obstacles within the camera's field of view. The image gradient (SIFT) features extracted from the corresponding image regions are used to generate a training database of features that represents the robot, and a similar database is created for the background i.e. the environment. During testing, a Nearest Neighbor classifier [21] compares features extracted from the images with those in the training database in order to classify features and detect obstacles.

### B. Experimental Results

Given the learned models, the test hypothesis is that the combination of the processing schemes (using Algorithm 1) performs better than the individual processing algorithms. We are primarily interested in evaluating the detection and localization accuracy, and the experiments were designed appropriately. Given the different field of view of the individual sensors, the test cases were chosen to make the task as challenging as possible.

Similar to the training phase, the robot moved through a fixed sequence of poses with the obstacles placed at different points on the field. In our scenario, a detection accuracy $< 100\%$ reflects the inability of the robot to find the obstacles (i.e. there are no false positives). The localization errors were hence computed only when the obstacles were detected correctly. When an obstacle is detected using the processing scheme being evaluated, the robot stopped and performed additional trials, measuring the relative distance and bearing of the detected obstacles. The corresponding ground truth values were provided manually, except when the robot is well-localized and knows the global position of the obstacle it has detected. The difference between the estimated and ground truth values provide the error values summarized in Table II. The first three rows of Table II correspond to the individual processing schemes: ultrasound (US), vision-color (VC) and vision-SIFT (VS). The last row corresponds to the results obtained with our algorithm, i.e. US+VC+VS. Each entry in the distance-error and bearing-error columns was computed over $\approx 20$ different obstacle positions, with $15$ trials at each position.

The entries in the last column (labeled "Accuracy") in Table II were computed by capturing several images as the robot moved through the sequence of poses to compute the distance and bearing errors. The robot logged $\approx 400$ images for each processing scheme, with more than half the images containing the obstacles. Some of these images corresponded to situations where the obstacles were outside the angular and/or distance-based field of view of one or more schemes. These images were hand-labeled to provide the ground truth i.e. the presence or absence of obstacles, and used to compute the detection accuracy of each processing scheme. For the task of localizing moving obstacles in the robot soccer domain, an error of $\leq 10$cm in distance and $\leq 10^o$ in bearing, along with a detection accuracy $> 90\%$ would be sufficient to operate robustly. The results in Table II are analyzed by comparing them to these target values.

| Scheme | Error | | Accuracy(%) |
|---|---|---|---|
| | Distance (cm) | Bearing (deg) | |
| Ultrasound (US) | $6.5 \pm 3.6$ | $--$ | 70 |
| Vision-Color (VC) | $17.5 \pm 8.7$ | $8.5 \pm 4.0$ | 81.5 |
| Vision-Sift (VS) | $--$ | $9.1 \pm 4.5$ | 85.5 |
| $US + VC + VS$ | $9.2 \pm 5.1$ | $8.8 \pm 4.3$ | 91.5 |

TABLE II: The distance and bearing errors, and the detection accuracy of the processing schemes. Proposed approach is more robust than the individual processing schemes.

The processing scheme based on ultrasound information (row 1 in Table II) computes the distances very accurately but the bearing information is very limited. In addition, it cannot detect obstacles beyond a certain distance ($\leq 150$cm) and outside the $60^o$ cone for each sensor. Hence, though the detection accuracy is almost $100\%$ within its detection zone, the overall accuracy over the range of test cases is only $70\%$.

When the obstacles are detected using just the color information (row 2 in Table II), the error in distance estimates is higher (compared to $US$) because distance computation based on image region sizes is noisy. The bearing estimates are however reasonably accurate. The detection accuracy is not good because the uniform pattern cannot be detected from all viewpoints, and at large distances ($> 200$cm).

When the obstacles are detected using just the SIFT-based processing scheme (row 3 in Table II), the bearing estimates are statistically similar to those obtained using color information. The expected error in bearing does not change much as a function of the measured bearing, and the scale and orientation invariance results in a higher detection accuracy than the color-based scheme. The robots can now be detected at different viewpoints and up to a distance of $\approx 400$cm. The scheme fails when the obstacles are a significant distance away from the robot, or if very few SIFT features are detected on the obstacles (e.g. strong highlights). However, obstacle distances cannot be computed reliably.

Though not included in Table II, combining ultrasound with one of the vision-based techniques does provide an improvement, but either the detection accuracy is low (e.g. US+VC), or the distance computation is inaccurate or infeasible (e.g. US+VS with the obstacle outside the FOV of the ultrasound sensors). However, when all three processing

schemes are merged together (final row in Table II), the system is able to exploit the complementary features of the individual schemes. The localization errors are within the desired limits, and the detection accuracy is above the desired value. The distance errors are higher than those obtained with just the US scheme because the ultrasound sensors can help reduce distance errors only when the obstacle is within its field of view. Furthermore, the proposed approach is better than an ad-hoc information fusion approach with a distance error of $\approx 14.1 \pm 6.6$cm and a detection accuracy of $\approx 85\%$ after extensive manual tuning over many hours. These results show that the proposed approach is more robust than the individual schemes.

| Scheme | Time/frame (msec) |
|---|---|
| US | 33.3 |
| VC | 33.3 |
| VS | $125 \pm 52$ |
| US + VC + VS | 37.2 |

TABLE III: Computation time per frame for each processing scheme. With some approximations, the combined scheme can function at close to frame rate.

Next, Table III compares the running times of the processing schemes. The ultrasound-based and color-based processing schemes individually take very little computational effort, resulting in real-time operation (30Hz = 33msec/frame) when executed in conjunction with the existing modules (vision, localization, team coordination etc). Our SIFT implementation operates on low-resolution images to reduce the processing time, but the SIFT-based approach is still computationally expensive. However, if SIFT-based approach is applied at a reasonable frequency (e.g. once every 10 frames), the combined approach can operate at close to frame rate (on average). The incorporation of the Kalman

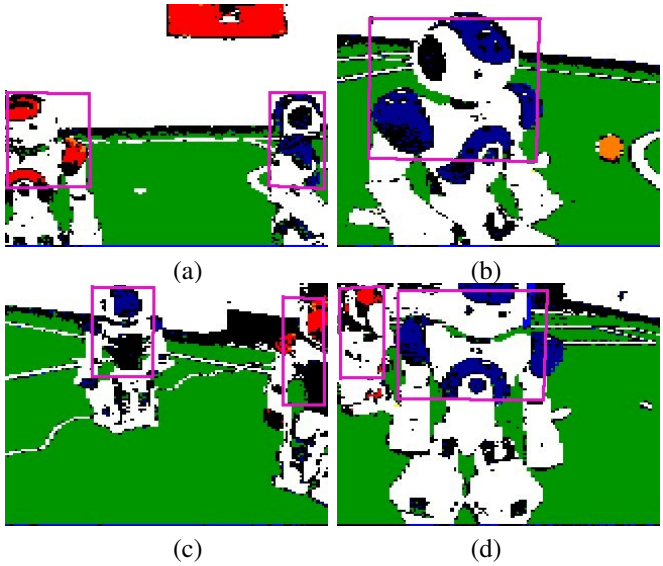


Fig. 3: (a)-(d) Image results: detected obstacles are enveloped in *pink* boxes superimposed on segmented images.

filter-based tracking of obstacles provides two benefits: (a) it propagates belief for a few frames after the last sighting of an obstacle, making it robust to intermittent noisy measurements; and (b) it can be used to incorporate the velocity of moving obstacles and account for the relative motion between the robot and the obstacle. The overall detection and tracking performance is hence observed to be statistically similar to that reported in Table II.

Some qualitative results of the robot's performance are shown in Figure 3. The detected obstacles are enveloped in rectangular bounding boxes (in *pink* when viewed in color) that are superimposed on the segmented images.

## IV. Related Work

Humanoid robots, and mobile robots in general, are increasingly being equipped with multiple sensors and used in practical applications [1], [2], [3], [4]. Since the sensors have different capabilities and the environment changes dynamically, it is essential that all the available information be used effectively.

In recent times, the DARPA challenges [2], [3], [4] have had robot vehicles equipped with several sensors (lasers, cameras, GPS etc) navigating autonomously in the real-world domains such as deserts and urban roads. However, most of the decision-making was based on input from the range-finders and GPS, and visual input was predominantly used only for close-range obstacle avoidance. In related work, Wellington et al. [12] used lasers and cameras to find the true ground height and hence traversability of vegetation-covered regions. Rankin et al. [14], on the other hand, merged stereo vision and thermal signatures to detect drop-offs at night. Murarka et al. [22] utilized stereo and range information to detect drop-offs on a robot wheelchair. However, these methods fail to explicitly model the errors associated with each information source—information fusion is typically based on manually specified heuristics.

Sensor fusion has been extensively studied in the field of networks and multiagent systems [23]. Several approaches have also been proposed for specific tasks such as image registration, using state-estimation methods such as Kalman filters or Bayesian networks [20], [21]. However, most of these strategies use heuristics that require manual supervision when applied to robot domains. Some methods are also computationally expensive for robot domains.

On humanoid robots, localization and object tracking is accomplished using the probabilistic state estimation techniques used in other robot domains (e.g. Kalman filters and Monte Carlo methods) [24]. Research in the RoboCup framework [25] and the humanoid robotics community has resulted in several innovative techniques for challenges specific to humanoid robot platforms. For instance, robust techniques have been developed to address the challenges of robot control and balance [10], [11]. Approaches have also been proposed for sensor-based navigation, for instance using stereo-vision [26]. Humanoid robots are also increasingly being used for human robot interaction studies [27]. However, as with other mobile robot platforms, there is a need for an efficient strategy to fully utilize the available information in order to operate robustly in dynamic environments. This

paper presents an approach that addresses an instance of this information fusion challenge.

## V. Conclusions and Future Work

Developments in sensor technology have resulted in the deployment of mobile robots in applications such as medicine and autonomous navigation [1], [2], [4]. A major challenge for a robot equipped with multiple sensors, is the ability to efficiently merge the information obtained from each sensor through different processing schemes, in order to operate robustly in dynamic environments.

In this paper we have presented an instance of such multisensor information fusion using range and visual information. The robot is able to autonomously learn models that predict the performance of the schemes that process the visual input (from a color camera) and range input (from ultrasound sensors). The learned models are used in a probabilistic approach that effectively merges the information obtained from the different sources. In the robot soccer domain, we have shown that a humanoid robot is able to detect and localize obstacles more robustly than what could have been accomplished in the absence of such an information fusion scheme.

In multirobot settings (e.g. robot soccer, disaster rescue), information merging can have other advantages. Information communicated by teammates can be merged to obtain robust estimates about areas that are hidden from the robot's field of view, which would prove very useful in disaster rescue scenarios [7]. In robot soccer, if the robot knows the global position of one of its teammates (e.g. the teammate communicates its pose with high certainty), relative distance and bearing to this teammate can help the robot localize itself in the global frame of reference. Furthermore, when the communication is delayed or noisy, such information fusion schemes may enable robots to coordinate better.

Currently the robot learns its errors models (and object models) in a separate training phase. However, the robot can bootstrap such that the learned models are updated over time in response to environmental changes. Obstacles and other objects that are found to be stationary can even be used as "fixed markers" that enable a robot to localize when the initial set of field markers (e.g. goals) are not visible.

This paper shows the feasibility of effectively using the available information for robust performance on a humanoid robot. The long-term goal is to enable robots to autonomously learn environmental models, effectively merge information obtained from different sources, and operate robustly in real-world application domains.

## Acknowledgments

We thank the members of the TT-UT AustinVilla robot soccer team who contributed part of the code used in our experiments. This work was sponsored in part by ONR award N00014-09-1-0658.

## References

[1] J. Pineau, M. Montemerlo, M. Pollack, N. Roy, and S. Thrun, "Towards Robotic Assistants in Nursing Homes: Challenges and Results," *Robotics and Autonomous Systems, Special Issue on Socially Interactive Robots*, vol. 42, no. 3-4, pp. 271–281, 2003.

[2] S. Thrun, "Stanley: The Robot that Won the DARPA Grand Challenge," *Journal of Field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[3] LAGR, "The DARPA Learning Applied to Ground Robots Challenge," 2005, www.darpa.mil/ipto/programs/lagr/.

[4] DARPA, "The DARPA Urban Robot Challenge," 2007, http://www.darpa.mil/grandchallenge/index.asp.

[5] Hokuyo, "Hokuyo Laser," 2008, http://www.hokuyo-aut.jp/products/urg/urg.htm.

[6] "Videre Design Camera," http://www.videredesign.com/stereo_on_a_chip.htm.

[7] J. Casper and R. R. Murphy, "Human-robot Interactions during the Robot-assisted Urban Search and Rescue Response at the WTC," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 33, no. 3, pp. 367–385, 2003.

[8] M. Sridharan and P. Stone, "Global Action Selection for Illumination Invariant Color Modeling," in *The IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2007.

[9] M. Sridharan, J. Wyatt, and R. Dearden, "HiPPo: Hierarchical POMDPs for Planning Information Processing and Sensing Actions on a Robot," in *International Conference on Automated Planning and Scheduling (ICAPS)*, September 14-18 2008.

[10] J. Rebula, F. Canas, J. Pratt, and A. Goswami, "Learning Capture Points for Humanoid Push Recovery," in *The IEEE International Conference on Humanoid Robots*, Pittsburgh, 2007.

[11] J. Pratt and B. Krupp, "Design of a Bipedal Walking Robot," in *Proceedings of the SPIE, volume 6962*, 2008.

[12] C. Wellington, A. Courville, and A. T. Stentz, "Interacting Markov Random Fields for Simultaneous Terrain Modeling and Obstacle Detection," in *Robotics: Science and Systems*, June 2005.

[13] P. Stone, M. Sridharan, D. Stronger, G. Kuhlmann, N. Kohl, P. Fidelman, and N. K. Jong, "From Pixels to Multi-Robot Decision-Making: A Study in Uncertainty," *Robotics and Autonomous Systems: Special issue on Planning Under Uncertainty in Robotics*, vol. 54, no. 11, pp. 933–943, 2006.

[14] A. Rankin, A. Huertas, and L. Matthies, "Nighttime negative obstacle detection for off-road autonomous navigation," in *SPIE*, 2007.

[15] Nao, "The Aldebaran Nao Robots," 2008, http://www.aldebaran-robotics.com/.

[16] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, and E. Osawa, "Robocup:The Robot World Cup Initiative," in *ICRA*, February 1997, pp. 340–347.

[17] SPL, "The RoboSoccer Standard Platform League," 2008, http://www.tzi.de/spl/.

[18] D. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision (IJCV)*, vol. 60, no. 2, pp. 91–110, 2004.

[19] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of ASME, Journal of Basic Engineering*, vol. 82, pp. 35–45, March 1960.

[20] R. Brooks and S. Iyengar, *Multi-Sensor Fusion: Fundamentals and Application with Software*. Prentice Hall, 1998.

[21] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer-Verlag New York, 2008.

[22] A. Murarka, M. Sridharan, and B. Kuipers, "Detecting Obstacles and Drop-offs using Stereo and Motion Cues for Safe Local Motion," in *The IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2008.

[23] L. Panait and S. Luke, "Cooperative Multi-Agent Learning: The State of the Art," *Autonomous Agents and Multi-Agent Systems*, vol. 11, no. 3, pp. 387–434, November 2005.

[24] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. Cambridge, USA: MIT Press, 2005.

[25] L. Iocchi, H. Matsubara, A. Weitzenfeld, and C. Zhou, Eds., *RoboCup-2008: Robot Soccer World Cup XII*. Berlin: Springer Verlag, 2009.

[26] J.-S. Gutmann, M. Fukuchi, and M. Fujita, "Real-time path planning for humanoid robot navigation," in *IJCAI*, 2005.

[27] M. A. Goodrich and A. C. Schultz, "Human-Robot Interaction: A Survey," *Foundations and Trends in Human-Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2007.