Self-Localization of a Mobile Robot using Fast Normalized Cross Correlation

Kai Briechle & Uwe D. Hanebeck
Institute of Automatic Control Engineering,
Technische Universität München,
80290 München, Germany
e-mail: Kai.Briechle@ei.tum.de

ABSTRACT

A self-localization concept for a mobile robot is presented, which is based on angle measurements to both known and unknown landmarks. The main contributions of this paper are the following:

- A fast normalized cross correlation algorithm (NCC) that uses a sum expansion of the template function and tables containing the integral over the image function (running sum) to detect the landmarks.
- A linear solution for the relative position and orientation update of the robot using angle measurements to unknown landmarks.

Furthermore, we apply a new kind of estimator for optimal sensor data fusion in the presence of both stochastic and deterministic errors for self-localization of the robot. Experiments demonstrate the feasibility of our approach.

1 INTRODUCTION

Mobile service robots have to estimate their global position and orientation during operation to accomplish typical service tasks. The application domain, for which our self-localization system is considered, is an indoor environment, where the robot travels at relatively high speed. We assume 24 hour operation, which means, that the illumination of the environment may change significantly. Furthermore, the robot has to cope with varying floor conditions.

The sensors we use for our self-localization system are

- odometric sensors
- a panoramic camera.

Odometry is used for relative self-localization, while the panoramic camera is used for both relative and absolute self-localization with respect to environmental landmarks.

Fusing of the information of both sensors is done with a new kind of estimator, that takes into account both stochastic and deterministic errors. This leads to robust and reliable self-localization in the considered environment.

2 PRINCIPLE OF SELF-LOCALIZATION

To determine the position and orientation of the robot relative to its previous position or to the world coordinate system with the panoramic camera, we measure a set of angles to landmarks in the environment of the robot [5]. An error analysis of the self-localization algorithm that calculates the position update shows that it is necessary to use landmarks that surround the robot to minimize the error. The landmarks are detected in an omnidirectional panorama image using a modified, fast normalized cross correlation (NCC) algorithm. The angles to the landmarks depend linearly on their position in the omnidirectional image.

A linear equation for the relative robot position and orientation with respect to last frame, which is used for relative position update, is derived. One angle measurement to a single landmark is sufficient for robot position update, if other sensor data is available.

The position estimate for the robot calculated by the self-localization algorithm is then fused with data from the odometry and the gyroscope using a new set theoretic estimator to take into account all available sensor data in an optimal way.

3 NCC-ALGORITHM

As our system uses angle measurements to known or unknown landmarks for self-localization, it is necessary to track the landmarks in the camera image during robot navigation. There are a lot of different, well-known algorithms in image processing for feature tracking, including the approach of template matching, for which normalized cross correlation is a reasonable choice [1]. Experiments revealed, that high pass filtering for edge detection leads to less accurate and

0-7803-5731-0/99$10.00 ©1999 IEEE

robust performance in finding and tracking artificial, rectangular retroreflecting landmarks compared to a template matching algorithm.

Using the normalized cross correlation coefficient (1) is far better

$$\gamma = \frac{\sum_{x,y}(f(x,y) - \bar{f}_{u,v})(t(x-u,y-v) - \bar{t})}{\sum_{x,y}(f(x,y) - \bar{f}_{u,v})^2 \sum_{x,y}(t(x-u,y-v) - \bar{t})^2} \tag{1}$$

but the drawback is, that the denominator is computationally expensive compared to the nominator, which could be for example calculated in the frequency range with the Fourier theorem. In (1) $f(x,y)$ is the image function and $t(x,y)$ is the template function. At every point (u,v) of the image, at which $\gamma = \gamma(u,v)$ is determined, the energy of the image $\sum f^2(x,y)$ has to be recalculated, whereas the template energy $\sum t^2(x,y)$ has to be precalculated only once. To overcome these problems, the use of tables containing the integral over the image function $f(x,y)$ (running sum) is proposed in [1].

$$\begin{aligned} s(u,v) = f(u,v) + s(u-1,v) + s(u,v-1) + \\ - s(u-1,v-1) \end{aligned} \tag{2}$$

$$\begin{aligned} s^2(u,v) = f^2(u,v) + s^2(u-1,v) + \\ + s^2(u,v-1) - s^2(u-1,v-1) \end{aligned} \tag{3}$$

$s(u,v)$ and $s^2(u,v)$ are the sum tables over the image function $f(x,y)$ and the image energy $f^2(x,y)$. With these tables, the denominator can be calculated in a very efficient manner, independent of the size N_x, N_y of the template, using

$$\begin{aligned} &\sum_{x=u}^{x=u+N_x-1} \sum_{y=v}^{y=v+N_y-1} f(x,y) = \\ &s(u+N_x-1, v+N_y-1) - s(u-1, v+N_y-1) \\ &- s(u+N_x-1, v-1) + s(u-1, v-1). \end{aligned} \tag{4}$$

As the number of computations required to calculate the nominator of the NCC-coefficient (1) in the frequency range with an FFT algorithm is still too high for real time processing with a standard PC, further simplification is required. The basic idea to simplify the calculation of the nominator is to expand the template function $t(x,y)$ to the weighted sum of K rectangular basis functions t_i

$$\tilde{t}(x,y) = \sum_{i=1}^{K} k_i t_i(x,y). \tag{5}$$

The function $t_i(x,y)$ is constant equal 1 inside an rectangular area $x_l^i \leq x \leq x_u^i \wedge y_l^i \leq y \leq y_u^i$ and zero otherwise. Figure 1c) shows an example of a typical template that is used as a landmark in the office environment. It is taken from a normal camera image to demonstrate how the algorithm works. This template can well be approximated by the weighted sum of 3 rectangular basis functions, which yields a new template function $\tilde{t}(x,y)$ (Figure 1e). For this example, the 3 basis functions t_i where selected manually.

For automatic determination of the basis functions, the quadratic criterion $J = \sum_{x,y}(t(x,y) - \tilde{t}(x,y))^2$ is used to calculate the quality of the approximation, and a recursive algorithm divides the template into rectangular basis functions. The iteration process is stopped, when J is below a predefined threshold. Note, however, that the approximation found by this algorithm is not globally optimal with regard to the number of basis functions required to approximate the template. Many features in an office environment can nevertheless be well approximated with few basis functions.

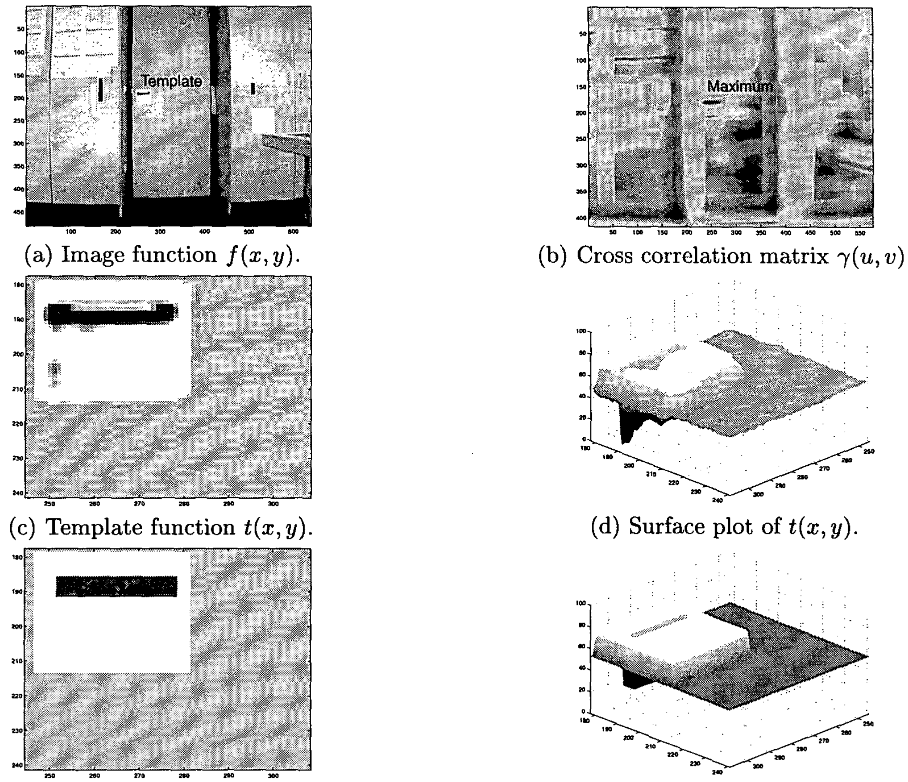
Using the sum expansion of the template function $\tilde{t}(x,y)$ allows to rewrite the nominator of the cross correlation coefficient as

$$N(u,v) = \sum_{i=1}^{K} k_i \sum_{x=x_l^i+u}^{x_u^i+u} \sum_{y=y_l^i+v}^{y_u^i+v} f(x,y), \tag{6}$$

where K is the number of basis functions and k_i is the coefficient for basis function i. With the integral table over the image function (2), that has already been calculated to simplify the determination of the denominator, which cannot be calculated in the frequency range, the inner double sum in (6) can be calculated with only 3 additions.

$$\begin{aligned} \sum_{x=x_l^i}^{x_u^i} \sum_{y=y_l^i}^{y_u^i} f(x,y) = s(x_u^i, y_u^i) - s(x_u^i, y_l^i) + \\ - s(x_l^i, y_u^i) + s(x_l^i, y_l^i). \end{aligned} \tag{7}$$

This means, that the number of calculations required to determine the nominator in (1) depends only on the number of rectangular basis functions used, but not on their size. The number of computations required to calculate the nominator of the NCC coefficient for an image of the size $M_x * M_y$ and a template of the size $N_x * N_y$ that has been approximated with K rectangular basis functions is given in table (1). It can be seen, that the number of multiplications depends linearly on the number of basis functions K. For our application, the typical size of the image function is 1500x60 pixel and 30x30 pixels for the template function. For example, two basis functions are used to approximate the rectangular landmarks we use for robot navigation. Table (2) shows, that the number of multiplications for the nominator is reduced 150 times compared to the FFT and 450 times compared to a direct calculation, assuming that the sum tables used for calculating



(a) Image function $f(x,y)$.
(b) Cross correlation matrix $\gamma(u,v)$.
(c) Template function $t(x,y)$.
(d) Surface plot of $t(x,y)$.
(e) Sum Expansion $\tilde{t}(x,y)$.
(f) Surface plot of $\tilde{t}(x,y)$.

Figure 1: Experiment: Template matching with fast normalized cross correlation.

the denominator are required for each algorithm. This means, that up to 300 basis functions may be used, before the computational load is equivalent to the FFT algorithm. It is assumed that the FFT algorithm requires that f and t be extended with zeros to a common power of two (zero padding).

$M_x = 1500, M_y = 30,$ $N_x = N_y = 30, K = 2$	Add / Sub	Mult
Direct calc.	41.0 Mio	41.0 Mio
FFT	20.0 Mio	13.4 Mio
New alg.	0.32 Mio	0.091 Mio

Table 2: Analysis of Complexity, Example.

In Figure 1b) the result of the NCC computed with the algorithm (5) (6) for the example shown in Figure 1a) is given. 3 basis functions where used to approximate the template, which is the highlighted door handle in Figure 1a). Despite this rough approximation (Figure 1e)), the NCC determines the position of the template in the original image correctly, yielding the maximum value at $[x, y]' = [245\ 178]'$. Figure 1d) and e) show a surface plot of the original and the approximated template function $t(x,y)$ and $\tilde{t}(x,y)$.

4 RELATIVE POSITION UPDATE

The algorithm described in the previous section allows to find and track natural and artificial landmarks with little computational effort in real time. As mentioned before, a typical operation area of the mobile robot contains many features, that can be used as landmarks, although their position in the world coordinate system is unknown. With these landmarks, a relative position update of the robot can be performed to improve the results of odometry, which may be disturbed by systematic errors, like unknown wheel diameters or varying floor conditions. On the other hand, from time to time, a global position update of the robot has to be calculated using known, artificial landmarks, for which a closed form, linear solution has been derived in [2]. In this way, the global position error of the robot can be constrained to an upper threshold.

Now, also for relative position update, a closed form linear solution has been derived, that yields the robot position and orientation with respect to the previous frame. Figure 2 shows the frames of three consecutive robot positions during travel at the discrete time steps

	Add / Sub	Mult
Direct calc.	$N_xN_y(M_x - N_x + 1)(M_y - N_y + 1)$	$N_xN_y(M_x - N_x + 1)(M_y - N_y + 1)$
FFT	$9M_xM_y \log_2(M_xM_y)$	$6M_xM_y \log_2(M_xM_y)$
New alg.	$(4K - 1)(M_x - N_x + 1)(M_y - N_y + 1)$	$K(M_x - N_x + 1)(M_y - N_y + 1)$

Table 1: Analysis of Complexity, nominator only.

T_{i-2}, T_{i-1} and T_i. At each frame, the angle α_i to the landmark is measured.

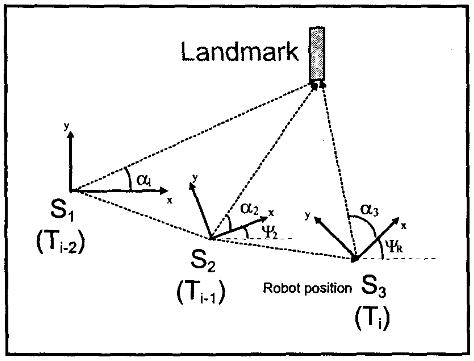


Figure 2: Relative position estimation.

If an estimate of the transformation ${}^1T_2$ from frame S_1 to frame S_2 with the translation $[\Delta x_{12}\ \Delta y_{12}]$ and the rotation ψ_2 is given from odometry and previous measurements of the camera sensor, the linear equation (8) can be derived from geometrical analysis of the robot positions relative to the landmark

$$\begin{bmatrix} \sin(\alpha_3)A \\ \cos(\alpha_3)A \\ B\Delta \end{bmatrix}' \begin{bmatrix} \mu \\ \nu \\ \xi \end{bmatrix} = C\Delta. \tag{8}$$

The vector $\underline{x} = [\mu\ \nu\ \xi]$ contains the transformed position and orientation of the robot at frame S_3 with

$$\begin{bmatrix} \mu \\ \nu \\ \xi \end{bmatrix} = \begin{bmatrix} \Delta x_{23} + \Delta y_{23}\tan(\psi_R) \\ -\Delta y_{23} + \Delta x_{23}\tan(\psi_R) \\ -\tan(\psi_R) \end{bmatrix} \tag{9}$$

and the coefficients are calculated by

$$\begin{aligned} A &= -\sin(\alpha_1 - \alpha_2 - \psi_2) & (10)\\ B &= \cos(\alpha_2 + \psi_2 - \alpha_3) & (11)\\ C &= -\sin(\alpha_2 + \psi_2 - \alpha_3) & (12)\\ \Delta &= \Delta x_{12}\sin(\alpha_1) - \Delta y_{12}\cos(\alpha_1). & (13) \end{aligned}$$

In these equations, α_i is the angle to the landmark measured at frame i, and Δx_{23} and Δy_{23} are the components of the vector from frame S_2 to frame S_3 measured in frame S_1. (8) can be used for position update with a single angle measurement, if it is fused with results from other relative sensors, like odometry or the gyroscope, in our case. If three landmarks are tracked simultaneously, the resulting linear system of equations can be solved for $[\mu\ \nu\ \xi]$ and the position and orientation of the robot relative to the previous frame can be directly calculated using the inverse transform of the state variables.

To estimate the resulting uncertainty which has to be calculated for sensor data fusion, the Jacobian matrix of the measurement error is determined at the estimated position $\hat{x} = [\hat{\mu}\ \hat{\nu}\ \hat{\xi}]$ of the robot.

5 ABSOLUTE POSITION UPDATE

As relative self-localization suffers from accumulating errors, the robot has to perform an absolute position update using landmarks, whose positions in the world coordinate system are known, from time to time. In [2], an efficient linear solution has been derived for the problem of calculating the position and orientation of the robot, if the angles α_i to N known landmarks have been measured. Our system can measure angles to both known and unknown landmarks at the same timestep. The results of the relative and the absolute position update algorithm have to be fused, taking into account the respective uncertainties. The positions of the known landmarks may be either taken from a global map of the environment or estimated simultaneously during robot operation.

6 DATA FUSION

As our robot system uses two sensors to determine its position in the operating space, that have different noise characteristics and lead to both systematic and stochastic errors, the data has to be fused taking into account both types of errors in an optimal way. For example, the odometric measurements suffer heavily from systematic errors, that cannot be completely avoided. In [4], a new filter is proposed for state estimation from noisy observations that are simultaneously corrupted by uncertainties with known distribution and uncertainties with known bounds. The new filter unifies Kalman filtering and set theoretic filtering: A Kalman filter is attained, when the bounded error goes to zero, and a set theoretic estimator is attained, when the stochastic error vanishes. When both types of uncertainty are present, the new estimator provides solution sets that are uncertain in a statistical sense.

7 IMPLEMENTATION

To measure the angles to the landmarks surrounding the robot, we use a panoramic camera that yields an omnidirectional view of the environment, (Figure 3).

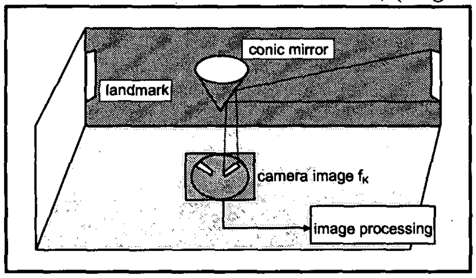


Figure 3: Landmark measurement (principle).

In order to compute the angles to the landmarks measured by the sensor, it is necessary to transform the camera image from cartesian coordinates to polar coordinates. The transformed image has the angle ϕ as the new x-axis and the radius r as the new y-axis. Because the angular image resolution is decreasing towards the center of the image, it is sufficient to transform a ring that contains the acquired landmarks. To determine the angles to the landmarks, the landmarks have to be extracted and tracked in the resulting image in polar coordinates. Then their position in the ϕ direction is calculated, which corresponds to the angle α_i to the i-th landmark in the robot coordinate system. As the transformation to the image in polar coordinates suffers from errors caused by the discretisation of the camera image, it is necessary to filter the image with a low-pass filter before further processing can be applied, to obtain the angles to the landmarks.

Figure 4 shows the omnidirectional camera sensor with the ring of LEDs used to illuminate the landmarks. The sensor is used for both relative and absolute position update, measuring angles to unknown landmarks and retroreflecting landmarks whose position is known. Lighting of the landmarks is used to make the localization process as robust as possible against varying lighting conditions, and to allow operation even in complete darkness. The CCD camera is a digital FireWire camera [6] mounted vertically inside the sensor. All image processing algorithms describe above as well as the self-localization algorithm are run on a standard Intel Pentium-III with 500 MHz, that is equipped with an FireWire adapter card.

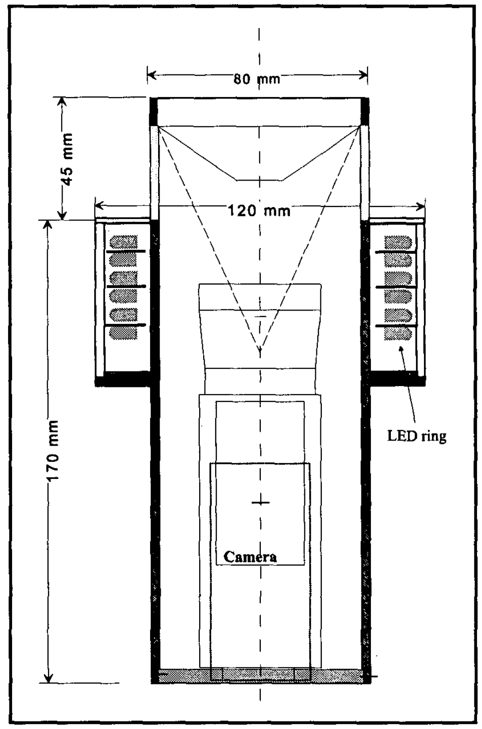


Figure 4: Camera sensor.

8 EXPERIMENTS

8.1 Relative position update

Figure 5 shows the results of the relative position update algorithm, when three unknown landmarks 1, 2 and 3 are tracked. The positions of frame S_1 and S_2 are assumed to be known from previous localization measurements and all further positions of the sensor, which is moved from the point "Start" to the point "End", are estimated using an one step estimation of frame S_3. It can be seen, that the results are fairly accurate with a deviation of about 5 cm although no other sensors are used to improve the position estimate.

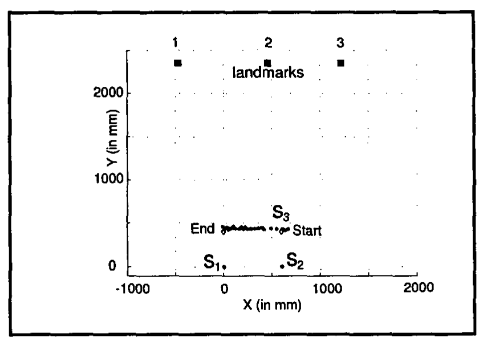


Figure 5: Relative localization experiment.

If more than 3 landmarks are available for position estimation, what is often the case in well structured office environments, the resulting measurement equations can easily be added to the equation system, yielding a redundant system of equations.

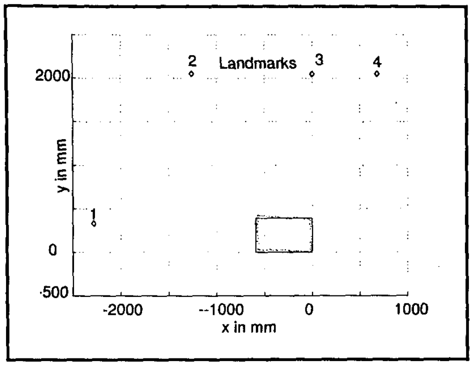


Figure 6: Absolute localization experiment.

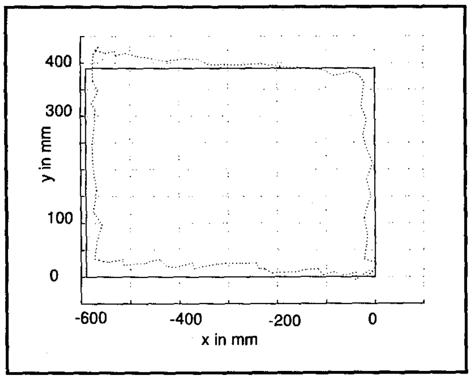


Figure 7: Absolute localization experiment (Zoom).

8.2 Absolute position update

Figure 6 shows an experimental setup with 4 known landmarks marked as diamonds, the trajectory the robot travelled and its estimated position using the absolute position update algorithm. In case of steady lighting conditions, any regions with sufficient contrast, whose position is known, may be taken as landmarks. For the experiment, black rectangular paper landmarks fixed on a white wall and gray cupboards where chosen. For operation at night, or in rooms with varying illumination, the sensor can light the landmarks using its LED Ring. It can be seen from Figure 6 that the robot position in the global frame displayed with the dotted line is estimated with a maximum error of 50 mm.

9 CONCLUSIONS

We have proposed a concept for self-localization of a service robot, that is reliable and robust, using measurements from both odometric sensors and a panoramic camera.

A new fast algorithm for the computation of the normalized cross correlation has been derived, that uses a sum expansion of the given template function t and rectangular basis functions. The number of calculations required depends linearly on the number of basis functions used, but not on the size of the template. It has been shown that it is possible to use this approximation to track features in the operating space of the robot with up to 150 times less calculations compared to the use of a standard FFT algorithm.

A linear solution for relative position update of the robot using angle measurements has been derived and validated in simulations. A single landmark is sufficient for a position update, if further information from other sensors is included. If three or more landmarks can be tracked, a relative position update can be calculated from the resulting system of linear equations directly.

With the new set theoretic and stochastic estimator (SSI) derived in [3, 4] it is possible to fuse the various sensor data of our localization system taking into account both systematic and stochastic errors in an optimal way. Further investigations will concentrate on the question whether it is possible to use the proposed algorithms for simultaneous map building and localization and how relative and absolute position update can be combined in an optimal manner.

References

[1] J. P. Lewis: *Fast Normalized Cross-Correlation*, Industrial Light and Magic.

[2] U. D. Hanebeck: *Lokalisierung eines mobilen Roboters mittels effizienter Auswertung von Sensordaten und mengenbasierter Zustandsschätzung*, Dissertation, TU Muenchen, Fortschrittsberichte VDI, Reihe 8: Meß-, Steuerungs- und Regelungstechnik, Nr. 643, VDI Verlag, Duesseldorf, 1997, ISBN 3-18-364308-1.

[3] U. D. Hanebeck, J. Horn: *A New State Estimator for a Mixed Stochastic and Set Theoretic Uncertainty Model*, Proceedings of SPIE Vol. 3720, AeroSense Symposium, Orlando, Florida, USA, April 1999.

[4] U. D. Hanebeck, J. Horn: *A New Estimator for Mixed Stochastic and Set Theoretic Uncertainty Models Applied to Mobile Robot Localization*, Proceedings of the 1999 IEEE International Conference on Robotics and Automation (ICRA'99), Detroit, Michigan, USA, May 1999.

[5] Y. Yagi, Y. Nishizawa and M. Yachida: Map-Based Navigation for a Mobile Robot wtih Omnidirectional Image Sensor COPIS, IEEE International Transactions on Robotics and Automation, Vol. 11, page 1062-1067, 1995.

[6] Sony: *Color digital camera DFW-V300*, Service Manual, Sony Corporation, Engineering Services Dept. 1998