Global Color Image Features for Discrete Self–localization of an Indoor Vehicle

Włodzimierz Kasprzak, Wojciech Szynkiewicz, and Mikołaj Karolczak

Warsaw University of Technology, Institute of Control and Computation Eng.,
ul. Nowowiejska 15/19, 00-665 Warsaw, Poland
W.Kasprzak@ia.pw.edu.pl
http://www.ia.pw.edu.pl/

Abstract. In autonomous indoor navigation some number of localizations and orientations of the vehicle can be learned in advance. No artificial landmarks are required to exist. We describe and compare the detection of several global features of color images (sensor data). This constitutes the measurement process in a self-localization approach that is based on Bayes filtering of a Markov environment - the posterior probability density over possible discrete robot locations (the belief) is recursively computed. The approach was tested to provide robust results under varying scene brightness conditions and small measurement errors.

1 Introduction

The localization process of an autonomous robot takes as input a previously acquired map, an estimate of the robot's current pose, and a set of sensor data acquired in current pose, and it produces as output a new estimate of the robot's pose [1,5,7]. Obviously, any input data for the localization process may be incomplete and distorted by noise or errors. In generally, pose means the position and orientation of the robot in the world coordinates or global map.

The vision data is acquired by a passive sensor, i.e. a camera does not influences the environment by its measurement process. This kind of sensor is especially applicable for indoor navigation in environments, that are populated by humans, i.e. offices, hospitals, museums, etc. [3]. Additionally, image processing methods can rely on natural landmarks, whereas this case for the active sensor devices has started to be studied only recently [2]. The use of image analysis methods in robot navigation has been intensively studied over the past 30 years [8,9,10]. In this paper we focus on general image features, that could be relatively insensitive to changing lighting conditions, but at the same time, can be relatively easy computed, to be obtained in real-time by a simple processing unit.

In theoretical terms the localization process is equivalent to a Bayes filtering of a finite environment satisfying the Markov condition, i.e. past and future data are conditionally independent if one knows the current state. During the localization process the posterior probability density over possible discrete robot

A. Gagalowicz and W. Philips (Eds.): CAIP 2005, LNCS 3691, pp. 620–627, 2005.
© Springer-Verlag Berlin Heidelberg 2005

locations (the belief) is recursively computed. We describe a detailed algorithm for the discrete self-localization scheme and we propose and test different global features of monochromatic images (if the brightness of observed scene is constant or it can be compensated) and another (more robust) set of image features, based on color information and localization ques.

2 The Self-localization Algorithm

A typical state recursive estimation can be performed in terms of a Kalman Filter [11]. It can be shown that a normal distribution of the measurement error induces a Gaussian distribution of the state's pdf provided by the Kalman Filter. This is a unimodal distribution and as a direct consequence there is always one best state estimated. Hence KF is suitable for tracking a single hypothesis but not many possibly competitive hypotheses, unless we use many instances of the filter.

2.1 The Method of State Condensation

The general discrete self-localization scheme [13], based on Bayes filtering of a Markovian environment, is also called *state condensation* or *particle filtering* [6], [7]. It assumes, that the number of states can be limited to a finite number. Only then it is computationally feasible to estimate the probability distribution over states.

By *belief* we denote the pdf of states upon the condition of a sequence of observations (measurements m_t):

$$\forall s^k : Bel_t(s^k) = p(s_t^k | m_t, m_{t-1}, \ldots, m_{t-n}) \tag{1}$$

In the learning phase the system should acquire two a priori pdf's:

1. The a priori conditional pdf of measurement upon state, i.e. for each discrete state $s \in S$ and possible measurement vector m to determine the pdf: $p(m|s)$;
2. The a priori pdf of state transition

$$p(s_{t+1}^k | s_t^l, \ldots, s_0^i) = p(s^k | s^l) \tag{2}$$

where $s_t^l, \ldots, s_0^i$ is the history of past best belief states. In autonomous navigation the action performed by the vehicle or camera are usually known, due to the odometry. Hence, this knowledge can be incorporated into the state condensation scheme - for each pair of states s^k, s^j and each possible action a to determine the pdf of state transition with respect to action: $p(s^k | s^j, a)$.

The discrete self-localization algorithm consists of the initialization step and a main iterative belief "refinement" step with sub-steps of : *belief prediction, stochastic diffusion, measurement and modification of belief* (the reaction onto the measurement) [13].

2.2 The Algorithm of the Self–localization Process

1. Get the goal state.
2. Initialization of a default belief state at $t = 0$ (for example by a uniformly distributed pdf) $Bel_0(s^k) = p(s_0^k|H_0)$.
3. REPEAT until the goal state is not reached:
   (a) $t = t + 1$;
   (b) find the current best state: $s_{t-1}^* = \arg\max\ p(s_{t-1}|H_{t-1})$, where $H_{t-1} = (s_{t-1}, m_{t-1}, s_{t-2}, m_{t-2}, \ldots, s_0, m_0)$ is the history of past belief states and measurements;
   (c) determine and perform the next action resulting from minimization of the distance between current best state and the goal state;
   (d) as the current action a_t and the a priori pdf $p(s_t|s_{t-1}, a_t)$ are known the predicted belief state at time t can be computed $\widehat{Bel}_t(s^k) = \sum_s [p(s_t^k|s_{t-1}, a_t)p(s_{t-1}|H_{t-1})]$
   (e) acquire the measurement m_t at new position.
   (f) with the a priori pdf $p(m_t|s_t)$ modify the belief state at time t: $Bel_t(s^k) = p(s_t^k|H_t) = c_t p(m_t|s_t)\widehat{Bel}_t(s)$, where c_t is the current normalization coefficient (the sum of belief state distribution should be equal to 1).

3 Global Image Features

Due to the iterative approach, exhibited by the self-localization procedure, individual image measurements need not to be unique for all states - they can be similar for many states. Hence, we expect that easy computable, general-nature image features that are combined with a Gaussian-like belief state filtering should already lead to robust navigation. We also expect that changes of the measurements between the learning phase and the active phase of self-navigation (due to change of scene illumination or inaccurate position of the vehicle) can be compensated by a longer belief state refinement sequence.

In this section we propose different color feature detection schemas, obtained in the RGB, HSV and YC_bC_r color spaces. A complete feature vector for a single image consists of features obtained for several sub-images. In this way we add some general feature localization information to a state's measurement vector.

3.1 Image Feature Detection Methods

The following global features are computed for every sub-image:

1. *MeanVar6* - the three mean and three standard deviation values of every color component (i.e. for H, S and V channels of the HSV color space);

$$m = [m_1, m_2, m_3, std_1^2, std_2^2, std_3^2].$$

2. *Hist6* - the three dominating color components in the image with their density values.

$$m = [I_{c1}, I_{c2}, I_{c3}, den(I_{c1}), den(I_{c2}), den(I_{c3})].$$

I_{ck} - the dominating value of the k-th color component, $den(I_{ck})$ - the number of pixels with color I_{ck} in relation to the total number of pixels.

3. *FFT6x2* - the modules of first 6 components of a Fourier transform of the image components H(hue) and S(saturation).

$$m = [|F_{(0,0)}|, |F_{(0,1)}|, |F_{(1,0)}|, |F_{(0,2)}|, |F_{(1,1)}|, |F_{(2,0)}|].$$

For a square image of size NN, the two-dimensional FFT is given as:

$$F_{(k,l)} = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} I(i,j) e^{-i2\pi(ki/N + lj/N)},$$

where $I(i,j)$ is the image in the spatial domain; the exponential term is the basis function - one such function corresponds to one Fourier coefficient $F_{(k,l)}$. The first coefficient $F_{(0,0)}$ represents the DC-component of the image and the $F_{(N-1,N-1)}$ represents the highest frequency component.

Obviously above vectors *MeanVar6* and *Hist6* are sensitive to scene illumination changes. For the HSV and YC_bC_r schemas we perform a Y-driven normalization of the color (and we can omit the Y-components from further consideration). The RGB color scheme requires an other intensity normalization scheme - in this caase we scale the color components in such a way, that the sum of intensities of all pixels is equal to some fixed reference value.

3.2 Learning the a Priori Pdf

The a priori pdf $p(m|s)$ should be computed during the learning phase. But the number of possible measurement vectors is infinite, usually there are continuous-valued components of m. In practice this pdf can be made explicit only during the active work. In the learning phase we compute and store the feature vectors associated with each discrete state.

During the active work the feature vector of current view is detected (assuming a previous normalization of the scene illumination or camera contrast).

The a priori pdf $p(m_{k+1}|s)$ is implicitly defined, as we can compute for each state s^k the value of a Gaussian distributed pdf, with mid point equal to zero, for the distance of $w|m_{k+1} - m(s^k)|^2$ (where w is a weighting vector that adjusts the intervals of particular components to some common interval).

The a priori pdf $p(m_{k+1}|s)$ is defined according to the difference of both measurement vectors: the current measurement m_{k+1} at time k+1 and the stored measurement m_s for $\forall s \in s$. The conditional probability density is modelled by a 1–D Gaussian normal distribution, with its mid point corresponding to the zero value of a weighted difference $\sum_{i=1}^{N \times p} w_i |m^i_{k+1} - m^i_s|^2$ (where w is a weighting vector that scales the expected ranges of particular feature elements to some common level).

3.3 Test Scenes

In our experiments the camera was mounted on a mobile platform [14] (Fig. 1). The on-board processor with clock frequency of 900 MHz was able to process around 2 images per second. Three degrees of freedom of the vehicle were allowed: a translation along the X and Z axes by pre-defined unit steps and a rotation around its Y axis by an angle of $\pm 45^o$. The "on-ground" locations of states and of possible directions in three test scenes are shown in Fig. 2 and 3.

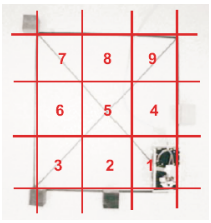
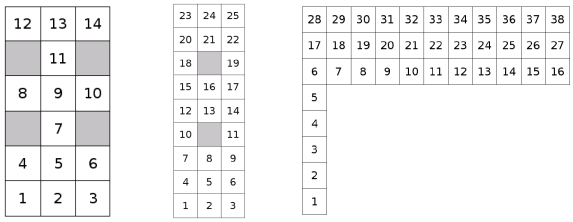
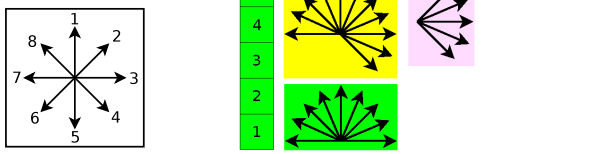


Fig. 1. An image is divided into 9 sub-images - a separate feature vector is computed for every sub-image



Office - 112 states. *Seminar room - 200 states.* *Corridor - 222 states.*

Fig. 2. The distribution of mobile platform positions during the learning phase for 3 test scenes



(a) *(b)*

Fig. 3. The possible orientations: (a) for the *Office* and *Seminar room*, and (b) for the *Corridor*

Three scenes with different illumination conditions, spatial distributions and different colors were available for testing (Fig. 4). Obviously, we expect that the camera holds the white color balance properly both during learning and active localization work. As we measure intensity-normalized color coefficients, some changes in illumination are not significantly disturbing the data if only the color balance remains to be constant.

Fig. 4. Examples of images acquired in different states and scenes and states

4 Test Results

4.1 Statistics of Features

An exemplary distribution of features over states is shown in Fig. 5. It is visible that the particular feature values are very often the same for different states, i.e. a single use of feature can not fully differentiate between two states. Another question is, how much sensitive these features are with respect to errors in robot's position/orientation and to scene illumination changes. The *p-value* of two distributions expresses the correctness of a hypothesis, that both distributions are statistically equivalent. If the p-value is equal to zero, then the above hypothesis is wrong and both features can be treated as being different. We computed the *p-value* for all pairs of feature vectors, where the first element of the pair corresponded to the state of the original scene, and the second element - to the compatible state in the real scene. Usually, the compatible views are displaced by several pixels and their illumination conditions are also slightly different. The feature set FFT6x2 performed best of all, i.e. the other two sets were more more sensitive to changes in state positions.

4.2 Test Runs

For every test scene and for each measurement method we have run the self-localization process 100 times, with randomly chosen start and goal states. A 99-

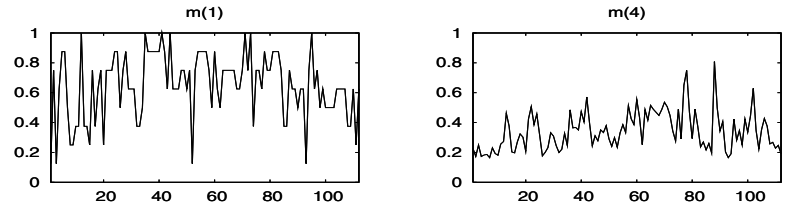


Fig. 5. The distribution of 2 measurement features over state from the *MeanVar6*-set of HSV color space for the *Office scene*

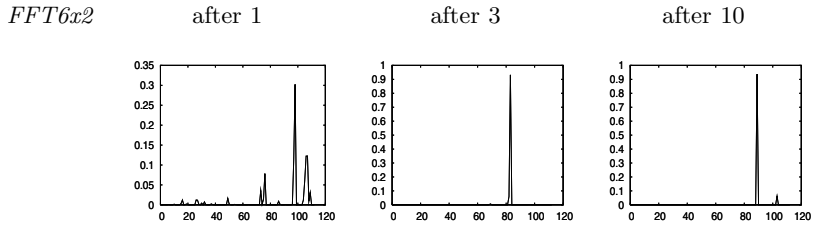


Fig. 6. Illustration of belief-state propagation - the belief state distribution after 1-st, 3-th and 10-th iteration

100 success rate appeared (i.e. the final goal state was reached) if color features expressed in the HSV and YC_rC_b color spaces were used. A particular self-localization process is illustrated on Fig. 6. At the start point the belief state distribution is an uniform distribution. After 3-4 steps the appropriate state, that corresponds to the real position, can already be selected as the belief value for such state dominates already the beliefs of remaining states. We assume that obstacles or moving persons in front of the vehicle will be detected by other sensors than vision. In order to limit their influence onto the measurement data we expect to "observe" higher wall sections instead of the floor.

5 Summary

Three different color image feature detection schemas were proposed and their use as the measurement step in a discrete self–localization process was experimentally verified. It was shown that even for natural scenes with changing illuminations and small perturbations of the odometry data, the use of even a small set of features, expressing only global information of a particular view, allows a robust and error-free self-localization (according to our test runs).

Acknowledgments

The authors would like to thank the Polish Ministry of Science and Information Society Technologies for supporting this work by the grant MNiI 4 T11A 003 25.

References

1. J. Borenstein, H. Everett, L. Feng, *Navigating Mobile Robots*. Wesley, Mass., 1996.
2. R.G. Brown, B.R. Donald, "Mobile Robot Self-Localization without Explicit Landmarks", *Algorithmica*, Springer Publ., vol. 26, pp. 515-559, 2000.
3. W. Burgard, A. Cremers, D. Fox, D. Hahnel, G.Lakemeyer, D. Schulz, W. Steiner, S. Thrun, "Experiences with an Interactive Museum Tour-Guide Robot", *Artificial Intelligence*, vol. 114, No. 1-2, pp. 3-55, 1999.
4. J. Denzler J., M. Zobel, "Automatische farbbasierte Extraktion naturlicher Landmarken und 3D-Positionsbestimmung. auf Basis visueller Information in indoor Umgebungen", V. Rehrmann (ed.), *Vierter Workshop Farbbildverarbeitung*, Fohringer-Vg., Koblenz, pp. 57-62, 1998.
5. D. Fox D., W. Burgard W., S. Thrun, "Markov Localization for Mobile Robots in Dynamic Environments", *Journal of Artificial Intelligence Research*, vol. 11, pp. 391-427, 1999.
6. D. Fox, S. Thrun, W. Burgard W., F. Dellaert, "Particle Filters for Mobile Robot Localization", In: Doucet A., DeFreitas N., Gordon N. (Eds), *Sequential Monte Carlo Methods in Practice*, Springer Publ., Berlin, etc., 2000.
7. M. Isard, A. Blake, "CONDENSATION - Conditional Density Propagation for Visual Tracking", *International Journal on Computer Vision*, vol. 29, No. 1, pp. 5-28, 1998.
8. W. Kasprzak, *Adaptive computation methods in image sequence analysis*, Prace Naukowe - Elektronika, No. 127, Warsaw Univ. of Technology Press, 2000.
9. C. Thorpe C. (ed.), *Vision and Navigation: The Carnegie Mellon Navlab*, Kluwer Academic Publ., Norwell, Mass., pp. 25-38, 1990.
10. I. Masaki, *Vision-based Vehicle Guidance*, Springer, New York etc., 1992.
11. Y. Bar-Shalom, T.E. Fortmann, *Tracking and Data Association*. Academic Press, 1988.
12. B. Heigl, J. Denzler, H. Niemann, "Combining Computer Graphics and Computer Vision for Probabilistic Visual Robot Navigation", *Proceedings of SPIE's 14th Annual International Symposium on Aerospace/Defense Sensing, Simulation and Controls*, Orlando, Florida, April 2000.
13. W. Kasprzak, W. Szynkiewicz, "Using color image features in discrete self-localization of a mobile robot", *9th IEEE International Conference on Methods and Models in Automation and Robotics* (August 2003, Miedzyzdroje, Poland), IEEE conference 8780, pp. 1101 - 1106, 2003.
14. B. Siemiatkowska, R. Chojecki, "Mobile Robot Navigation Based on Omnidirectional Sensor", *Proc. 1st European Conference on Mobile Robots ECMR'03*, EURON Conference, Radziejowice, Poland, September 2003, pp. 101-106, 2003.