**International Journal of Research in Science And Technology** http://www.ijrst.com

(IJRST) 2012, Vol. No. 2, Issue No. II, Apr-Jun **ISSN: 2249-0604**

# "LOCALIZATION AND MAPPING USING A SINGLE-PERSPECTIVE CAMERA"

**Neetu**

*Research Scholar, CMJ University, Shillong*

## INTRODUCTION

Localization and mapping are fundamental problems in mobile robotics. On the one hand, it is crucial to know where the robot is located in its environment in order to perform high-level applications such as delivery tasks. On the other hand, maps of the environment are often not available at the outset. Therefore, the ability to build a map of the environment is often an essential requirement. In the literature, the localization problem has been intensively studied in the past. It can be divided into two different classes. In the first class, robot localization is solved under the assumption that a map of the environment is known. It is the goal to track the robot's position in a map. If the start location is unknown, we refer to it as global localization. In the second class, the environment is unknown so it is required to build a map on the fly. This problem is called simultaneous localization and mapping (SLAM). In general, SLAM is a harder problem than localization in known environments since mapping and localization needs to be solved at the same time. SLAM is often considered to be a chicken-and-egg problem: A map is necessary to localize a robot, and the robot's position and orientation should be known in order to build a map. Thus, we have got circular cause and consequence.

A large group of researchers presented solutions to SLAM by means of proximity sensors such as laser range finders, sonars, or stereo vision cameras. In this thesis, we address the problem of global localization as well as SLAM using a wheeled robot equipped with a single-perspective camera only. In contrast to laser range finders, cameras have the advantage that they are inexpensive and lightweight. However, single cameras have a serious drawback. They cannot measure the distance to landmarks in the environment. Single cameras only provide bearing information. Therefore, we refer to single cameras as bearing-only sensors. The difference between proximity sensors and bearing-only sensors is illustrated in Figure 1. The single camera used in our work is called perspective in order to highlight the restricted field of view. By contrast, robot localization were also addressed with extreme wide-angle cameras using fish eye lenses.

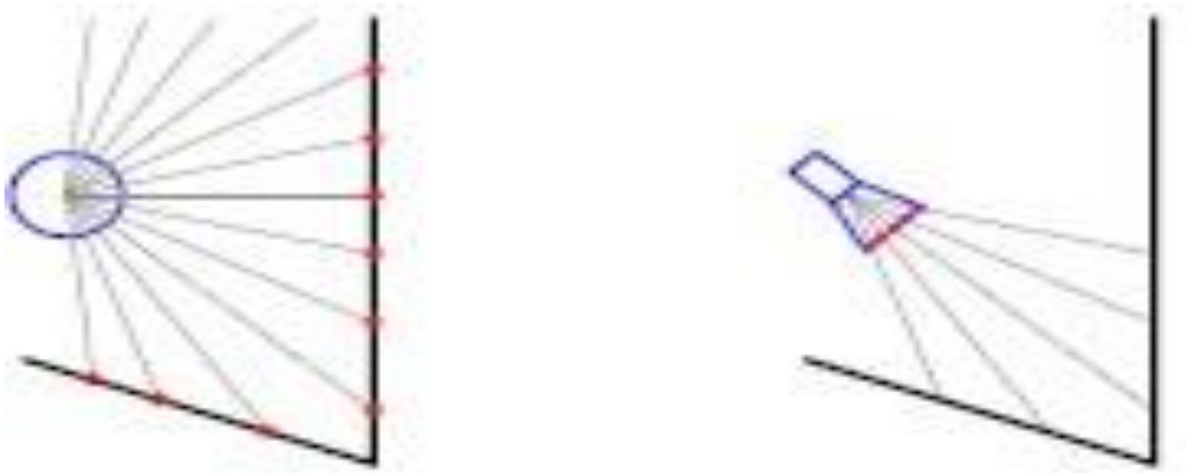

***Figure 1: An obstacle is perceived by different sensors (blue shapes). Left: Proximity Sensors perceive the distance as well as the angle to the structure. Right: For bearing-Only sensors the distance to the structure is unknown. Only the angles are observed*.**

In this thesis, we present a new approach to solve the simultaneous localization and mapping (SLAM) problem. In particular, we adapted the histogram-based landmark initialization process of Davision [7] to speeded-up robust features (SURF) [2] and Rao-Blackwellized particle filtering. Furthermore, we present a novel method to match features using a cost function that takes into account differences between the feature descriptors as well as spatial information. To find an optimal matching between observed features, we apply the Hungarian method [26] – a global optimization algorithm. Moreover, we introduce a framework which allow us to correct inaccurate odometry data based on visual information. Biased robot rotation is corrected using a gradient-descent like local optimization technique. Several experiments performed on a real-robot illustrates that our approach is reliable and robust to noise. In particular, it is shown that our feature matching technique outperforms other approaches.

## LOCALIZATION

The robot is moving through an environment. Our goal is to estimate the trajectory x1, xn – the path of the robot. The relative motion ut of the robot is acquired using an odometer sensor which measures wheel turns. Although calculations purely based on the robot odometry can serve as good estimates of the robot's pose in the first place, this approach is deemed to fail in the long run. Even very small errors in the odometry, for instance due to wheel slip or approximation errors, will accumulated over time so that the pose estimate will be far from the true pose after a certain amount of time. Therefore, the pose is corrected using camera images. Visual features in the image are treated as observations zt. A particle filter is used to represent and infer the posterior distribution bel(xt).

In this chapter, we assume that the environment is known and that a map of the environment is given. How to build a map by means of visual observations is explained in the next chapter.

**Motion Model**

We refer to the position and orientation of the robot in the 2D plain as pose x,

$$x = (x, y, \beta). \qquad (4.1)$$

The orientation β is zero if the robot is aligned with the y-axis as illustrated in Figure 2. As one can see, the coordinate system is a right-handed one since β increases as soon as the robot turns anti-clockwise.

Let us assume, our robot possesses a odometry device which provide us with the relative motion ut at discrete time steps. In addition, we assume that our robot has

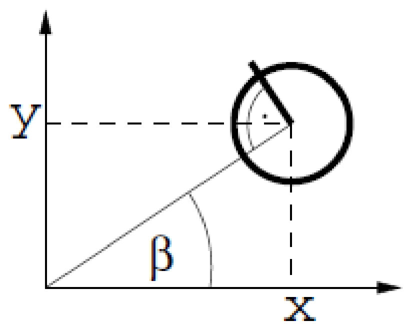


***Figure 2: Robot pose.***

a differential drive. That is it can drive forward and turn simultaneously. Sideward motion is not possible. A motion model is required p(x′|ut, xt−1) which estimates the new pose x′ given the relative motion ut and the old pose xt−1. The most common odometry-based motion model is the rotate-forward-rotate model. Here, the relative motion ut is decomposed into an rotation rot 1, a forward translation for and another rotation rot 2. It is assumed that these three component are affected by independent noise. On the one hand, rotate-forward-rotate is a good approximation of the robot motion in a theoretical point of view (see Thrun [55, pp. 132]). On the other hand rotate-forward-rotate has a serious shortcoming in practice. It does not cover backward motion. However, smallest errors in the odometry estimate could lead to a minimal pretended backward motion when the robot actually stands still. As a consequence, this minimal motion backwards would be modeled as a 180 degree rotation, a minimal forward translation and another 180 degree rotation. Thus, noise of a 360 degree rotation would be added, although the robot did not rotate at all. On this account, we use another motion model: the forward-sideward-rotate model. Here, the relative motion ut is decomposed in a forward/backward translation for, a sideward translation side and a final rotation rot . Although the real robot cannot drive sideward's, forward-sideward-rotate is an appropriate model in practice if the time difference between two odometers estimates ut and u′ t is not too large. Our forward-sideward-rotate implementation is given in Table 1. Typically the relative motion ut is given in terms of two poses: ¯xt−1 and ¯xt.

In line 2 to 4, the forward/backward translation for , the sideward translation side and the rotation rot is computed. Afterwards, noise is added (line 5 to 7) using the function sample(σ). Thereby, we sample from a Gaussian distribution with zero mean and variance σ. In our model, the noise in the forward motion, sideward motion and rotation depends on their own magnitudes |for|, |side| and |rot|. The strength of this dependency is covered by the three parameters αfor , αside and αrot . Straight

1: **function** FORWARDSIDEWARDROTATE($\mathbf{u}_t, \mathbf{x}_{t-1}$)
2: $\quad for := -(\bar{x}_t - \bar{x}_{t-1}) \cdot \sin(\bar{\beta}_{t-1}) + (\bar{y}_t - \bar{y}_{t-1}) \cdot \cos(\bar{\beta}_{t-1})$
3: $\quad side := (\bar{x}_t - \bar{x}_{t-1}) \cdot \cos(\bar{\beta}_{t-1}) + (\bar{y}_t - \bar{y}_{t-1}) \cdot \sin(\bar{\beta}_{t-1})$
4: $\quad rot := (\bar{\beta}_t - \bar{\beta}_{t-1}) \bmod 2\pi$

5: $\quad for_{noisy} := for + sample(\alpha_{for} \cdot |for|)$
6: $\quad side_{noisy} := side + sample(\alpha_{side} \cdot |side|)$
7: $\quad rot_{noisy} := rot + sample(\alpha_{rot} \cdot |rot|) + sample(\alpha_{rot\,for} \cdot |for|)$

8: $\quad x' := x_{t-1} - for_{noisy} \cdot \sin(\beta_{t-1}) + side_{noisy} \cdot \cos(\beta_{t-1})$
9: $\quad y' := y_{t-1} + for_{noisy} \cdot \cos(\beta_{t-1}) + side_{noisy} \cdot \sin(\beta_{t-1})$
10: $\quad \beta' := (\beta_{t-1} + rot_{noisy}) \bmod 2\pi$

11: $\quad$ **return** $(x', y', \beta')$
12: **end function**

***Table 1: Forward-sideward-rotate motion model. Remark that ut = (¯xt−1, ¯xt−1).***

forward translation is a very common motion. However, an intended forward motion could lead to an unnoticed rotation, e.g. due to bumps on the ground or wheel slip. Therefore, rotnoisy is also affected by the magnitude of the forward-backward translation |for | which is parameterized by foreside. On the other hand, we ignore that a rotation could affect for noisy. or sidenoisy. Only in artificial cases where the robot rotates constantly, rotation would have a serious impact on the noise in the translation. At last, the robot pose is updated in line 8 to 10. The use of the modulo operator ensures that the angle β always lies in the interval [0, 2π].

## BIBLIOGRAPHY

[1] A. Abdel-Hakim and A. Farag. CSIFT: A SIFT descriptor with color invariant characteristics. In Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), volume 2, pages 1978–1983, 2006.

[2] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. In Proc. of the European Conf. on Computer Vision (ECCV), 2006.

[3] M. Bennewitz, C. Stachniss, W. Burgard, and S. Behnke. Metric localization with scale-invariant visual features using a single perspective camera. InProc. of the European Robotics Symposium (EUROS), Palermo, March 2006.

[4] P. Biber and W. Straer. The normal distributions transform: A new approach to laser scan matching. In Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2003.

[5] D. Brown. Decentering distortion of lenses. Photometric Engineering, 32(3):444–462, 1966.

[6] A. Censi, L. Iocchi, and G. Grisetti. Scan matching in the hough domain. In Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA), 2005.

[7] A. Davison. Real-time simultaneous localisation and mapping with a single camera. In Proc. of the Int. Conf. on Computer Vision, Proc. of the Int. Conf. on Computer Vision (ICCV).

[8] A. Davison, I. Reid, N. Molton, and O. Stasse. MonoSLAM: Real-time single camera SLAM. IEEE Transaction on Pattern Analysis and Machine Intelligence, 29(6), 2007.

[9] F. Dellaert, S. Seitz, C. Thorpe, and S. Thrun. EM, MCMC, and chain flipping for structure from motion with unknown correspondence. Machine Learning, 2003.

[10] R. Diestel. Graph Theory, volume 173 of Graduate Texts in Mathematics. Springer, Heidelberg, 2005.

[11] T. Duckett, S. Marsland, and J. Shapiro. Fast, on-line learning of globally consistent maps. Journal of Autonomous Robots, 12(3):287 – 300, 2002.

[12] M. Fischler and R. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM, 24:381–395, 1981.

[13] J. Folkesson and H. Christensen. Outdoor exploration and SLAM using a compressed filter. In Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA), Taipei, Taiwan, 2003.

[14] U. Frese, P. Larsson, and T. Duckett. A multilevel relaxation algorithm for simultaneous localisation and mapping. IEEE Transactions on Robotics, 21(2):1–12, 2005.

[15] G. Grisetti, C. Stachniss, S. Grzonka, and W. Burgard. A tree parameterization for efficiently computing maximum likelihood maps using gradient descent. In Proc. of Robotics: Science and Systems (RSS), 2007. to appear.

[16] D. H¨ahnel, D. Fox, W. Burgard, and S. Thrun. A highly efficient FastSLAM algorithm for generating cyclic maps of large-scale environments from raw laser range measurements. In Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2003.

[17] C. Harris and M. Stephens. A combined corner and edge detector. In Proc. Of the 4th Alvey Vision Conference, 1988.

[18] R. Hartley. Euclidean reconstruction from uncalibrated views. Application of Invariance in Computer Vision, 1994.