A Probabilistic Framework for Weighting Different Sensor Data in MUREA

Marcello Restelli[1], Domenico G. Sorrenti[2], and Fabio M. Marchese[2]

[1] Politecnico di Milano, Dept. Elettronica e Informazione
piazza Leonardo da Vinci 32, 20135, Milano, Italy
restelli@elet.polimi.it
[2] Universitá degli Studi di Milano - Bicocca, Dept. Informatica
Sistemistica e Comunicazione, via Bicocca degli Arcimboldi 8, 20126, Milano, Italy
{sorrenti,marchese}@disco.unimib.it

Abstract. We shortly review a mobile robot localization method for known 2D environments, which we proposed in previous works; it is an evidence accumulation method where the complexity of working on a large grid is reduced by means of a multi-resolution scheme. We then elaborate a framework to define a set of weights which takes into account the different amount of information provided by each perception, i.e. sensor datum. The experimental activity presented, although the approach is independent on the sensory system, is currently based on perceptions coming from omnidirectional vision in an indoor environment.

1 Introduction

In previous works [1] [2] we introduced a method for robot localization, named MUREA (MUlti-Resolution Evidence Accumulation). It is an evidence accumulation method where the complexity of working on a large grid, i.e. at a useful accuracy of the localization estimate, is reduced by means of a multi-resolution scheme. This work has some correlations with other known works. In the following we very shortly review the ones which are more related to our, outlining the limitations which our proposal tries to overcome. *Grid-based Markov Localization* [3] operates on raw sensor measurements and, when applied to fine-grained grids, could turn into a very expensive computation. Our method can be classified as a grid-based method too; we propose to circumvent the problem with the use of a multi-resolution scheme. *Scan matching* techniques [4] [5], which make use of *Kalman Filtering*, are sensible to small errors in data association. Our proposal solves jointly the data association problem as well as the generation of a pose estimate. *Monte Carlo Localization* [6] have problems in recovering from unexpected events with respect to the motion model. Our approach reacts immediately, i.e. in the same activation, whenever the sensor data does not match the a priori pose estimate; in such case it turns into a global localization. The works mentioned so far, including ours, share the capability to handle so-called dense data sets.

D. Polani et al. (Eds.): RoboCup 2003, LNAI 3020, pp. 678–685, 2004.

2 Localization Algorithm

The method we proposed has three main components: the map of the environment, the perceptions and the localization engine. The environment is represented by a 2D geometrical map that can be inserted in the system through a configuration file or can be built by a SLAM system, e.g. as in [5]. The map is made up of simple geometrical primitives, like points, lines, circles, etc. On the other hand, we have sensor data. The localization engine takes in input the map of the environment as well as the perceptions, and outputs the estimated pose(s) of the robot (if the environment and/or the perceptions have inherent ambiguities).

In evidence accumulation the difficulty is in working at high resolution in order to get an accurate estimate. We divide the search-space in subregions (hereafter cells), to which we associate a counter. Each cell represents a localization hypothesis. Each perception increases the counter associated to a cell if some point, inside that cell, can be found, which is *compatible*, see [1] for details, with both the perception and the model. Then, on the basis of the votes collected by each cell, the method selects the ones which are more likely to contain the correct robot pose. Those cells undergo a refinement phase, which is where the multi-resolution comes into play. This process is further iterated on the refined hypotheses until a termination condition is matched, see [1] for details. The *compatibility verification* is in charge of checking whether a perception, e.g. a point perception, can be an observation of a certain map element, e.g. a line, for the pose associated to the cell. The more the votes of a cell, the more trustable is the match between the perceived world and its model, and the higher are the chances that the robot pose falls inside that cell. Therefore we search for the maximum in the vote accumulator; this is done at each level of resolution, which in turn increases at each iteration of the process.

3 The Weights of the Perceptions

According to what has been presented so far, the compatibility verification of a perception concludes with a vote or a non-vote to a cell, which in turn implies to increase or not, by a constant value, the cell counter (here we have the main difference with respect to other Bayes-based approaches). This is unsatisfactory since the amount of information carried by each perception is not the same. We think that the problem of detecting the perceptual saliency, i.e. the informativeness, of each sensor datum (i.e. perception) in order to give it more credit is quite general and applies to other approaches (including the method proposed in our previous work) as well. Our proposal is to analyze how much each perception is useful for the localization process and determine a criterion to define the weight that each perception should have in the voting phase. In this paper, we propose a voting approach which estimates the probability that each cell contains the actual robot pose. In this way, the selection phase will choose those cells that have an high probability of being the correct one.

There are two main factors that determine the perceptual saliency of a perception: the *informativeness* and the *reliability*. The information carried by a perception is directly proportional to the number of localization hypothesis that it excludes. This implies that perceptions which vote for few cells are more informative; on the other hand, a perception that votes all the cells provides no information, whilst a perception that votes only one cell provides the maximum information. The reliability of a perception is linked with the probability that this perception carries erroneous information, for localization purposes. We try to capture this concept with the probability of the correctness of a perception.

Definition 1. *A perception is* correct *if it votes, i.e. is compatible with, the cell containing the actual robot pose.*

3.1 Probabilistic Analysis

We want to identify which weight to associate to each perception, in order to select the cells that are most likely to contain the actual robot pose.

Definition 2. *Let Λ be the set of cells into which the C-Space is divided, and N the cardinality of Λ.*

Definition 3. *Let P be the set of perceptions acquired at time t.*

Definition 4. *Let Φ_c be the subset of all the perceptions acquired at time t which vote for cell c: $\Phi_c = \{p \in P | p \;\; votes \;\; c\}$.*

Definition 5. *Let Ψ_c be the subset of all the perceptions acquired at time t which do not vote for cell c: $\Psi_c = P \setminus \Phi_c$.*

Definition 6. *Let r be the cell containing the actual robot pose (a priori unknown).*

Definition 7. *Let $(c = r)$ be the event where the generic cell c is actually corresponding to cell r. The complement of the event $(c = r)$ is represented by $(c \neq r)$.*

Definition 8. *Let C_p be the event where perception p votes cell c, i.e. $p \in \Phi_c$.*

Definition 9. *Let V_p be the number of cells voted by p.*

Definition 10. *Let E_π be the event where all the perceptions belonging to the set π are not correct, i.e. they do not vote for r. π can contain a single perception; in this case we write E_p. With $\overline{E_\pi}$ we mean the event where the perceptions in π are correct.*

We want to evaluate the probability of event $(c = r)$. A priori, i.e. without having any information from the perception system, this value is the same for every cell.

$$Pr[(c = r)] = 1/N \tag{1}$$

Let us compute the same probability, given that some perceptions are compatible with cell c while others are not.

$$Pr[(c = r)|\Phi_c, \Psi_c] = \frac{Pr[(c = r), \Phi_c, \Psi_c]}{Pr[\Phi_c, \Psi_c]} \tag{2}$$

$$= \frac{Pr[(c = r), \Phi_c, \Psi_c|(c = r)] \cdot Pr[(c = r)] + Pr[(c = r), \Phi_c, \Psi_c|(c \neq r)] \cdot Pr[(c \neq r)]}{Pr[\Phi_c, \Psi_c|(c = r)] \cdot Pr[(c = r)] + Pr[\Phi_c, \Psi_c|(c \neq r)] \cdot Pr[(c \neq r)]}$$

$$= \frac{Pr[\Phi_c, \Psi_c|(c = r)] \cdot Pr[(c = r)]}{Pr[\Phi_c, \Psi_c|(c = r)] \cdot Pr[(c = r)] + Pr[\Phi_c, \Psi_c|(c \neq r)] \cdot Pr[(c \neq r)]} \tag{3}$$

First we applied Bayes'formula (eq. (2)), then we develop both the numerator and the denominator through the total probability lemma, and finally we deleted the second addendum (since $Pr[(c = r)|(c \neq r)] =_{def} 0$) and obtained the expression of eq. (3). If we substitute eq. (1) in eq. (3) and divide both the numerator and the denominator by $Pr[\Phi_c, \Psi_c|(c \neq r)]$, we obtain:

$$Pr[(c = r)|\Phi_c, \Psi_c] = \frac{\frac{Pr[\Phi_c, \Psi_c|(c=r)]}{Pr[\Phi_c, \Psi_c|(c\neq r)]} \cdot \frac{1}{N}}{\frac{Pr[\Phi_c, \Psi_c|(c=r)]}{Pr[\Phi_c, \Psi_c|(c\neq r)]} \cdot \frac{1}{N} + \frac{N-1}{N}} = \frac{\Sigma_c}{\Sigma_c + (N-1)}, \tag{4}$$

where $\Sigma_c = \frac{Pr[\Phi_c, \Psi_c|(c=r)]}{Pr[\Phi_c, \Psi_c|(c\neq r)]}$.

The Numerator of $\boldsymbol{\Sigma_c}$. The numerator of Σ_c is the probability that the perceptions which belong to Φ_c vote c and those that belong to Ψ_c do not, given that c is the cell containing the actual robot pose. This probability equals the probability that the Φ_c perceptions are correct, and that the other Ψ_c perceptions are not correct (see def. 1).

$$Pr[\Phi_c, \Psi_c|(c = r)] = Pr[\overline{E_{\Phi_c}}, E_{\Psi_c}] = \prod_{p \in \Phi_c} (1 - Pr[E_p]) \cdot \prod_{p \in \Psi_c} Pr[E_p], \tag{5}$$

Eq. (5)exploits the fact that these events, i.e. whether the perceptions are correct or not, are statistically independent (the noise, which makes them vote or not a cell c, acts independently on each perception).

The Denominator of $\boldsymbol{\Sigma_c}$. As far as the denominator of Σ_c is concerned, if we assume that the perceptions vote independently a cell c that does not contain the actual robot pose, we can introduce the following factorization:

$$Pr[\Phi_c, \Psi_c|(c \neq r)] = \prod_{p \in \Phi_c} Pr[C_p|(c \neq r)] \cdot \prod_{p \in \Psi_c} Pr[\overline{C_p}|(c \neq r)] \tag{6}$$

The probability that a perception p votes a cell c, given that c does not contain the actual robot pose, is:

$$Pr[C_p|(c \neq r)] = Pr[C_p|(c \neq r), E_p] \cdot Pr[E_p] + Pr[C_p|(c \neq r), \overline{E_p}] \cdot Pr[\overline{E_p}] =$$

$$= \left(\frac{V_p}{N-1}\right) \cdot Pr[E_p] + \left(\frac{V_p - 1}{N-1}\right) \cdot (1 - Pr[E_p]) = \frac{V_p - (1 - Pr[E_p])}{N-1} \tag{7}$$

The probability that a perception p votes a cell c is equal to the ratio of the number of cells voted by p (V_p) and the total number of cells (N). Given that $c \neq r$ the number of possible cells becomes $N-1$. If we also know that the perception p is not correct, c could be any of the Vp cells voted by p. On the other hand, if we know that the perception p is correct, there are only $V_p - 1$ cells that are voted by p and are different from r.

Let us now consider the probability that p does not vote c, given that c does not contain the actual robot pose; its value is:

$$Pr[\overline{C_p}|(c \neq r)] = Pr[\overline{C_p}|(c \neq r), E_p] \cdot Pr[E_p] + Pr[\overline{C_p}|(c \neq r), \overline{E_p}] \cdot Pr[\overline{E_p}] = \\ = \left(\frac{N - V_p - 1}{N-1}\right) \cdot Pr[E_p] + \left(\frac{N - V_p}{N-1}\right) \cdot (1 - Pr[E_p]) = \frac{N - V_p - Pr[E_p]}{N-1}, \quad (8)$$

which comes from considerations similar to those used for eq. (7).

***Σ_c* Expression.** Finally, using the results of eq. (5), (6), (7) and (8) we can obtain the expression for Σ_c and, from it, the expression of $Pr[(c = r)|\Phi_c, \Psi_c]$ (see eq. (4)):

$$\Sigma_c = \prod_{p \in \Phi_c} \frac{(N-1) \cdot (1 - Pr[E_p])}{V_p - (1 - Pr[E_p])} \cdot \prod_{p \in \Psi_c} \frac{(N-1) \cdot Pr[E_p]}{(N - V_p) - Pr[E_p]}. \quad (9)$$

It is worthwhile observing that the introduction of Σ_c allows each perception to provide a contribution, i.e. a factor of Σ_c, which is independent on the contribution of the other perceptions. Moreover, from eq. (4) we can notice that $Pr[(c = r)|\Phi_c, \Psi_c]$ is a function which is monotonically larger than Σ_c. This implies that cells which have higher values of Σ_c will also have higher probabilities of being the cell that contains the actual robot pose.

In the following section we will show how this result leads to the implementation of a weighting method that can be used in our localization approach.

3.2 Implementation

We would like, for each perception, to just add a quantity to the voted cells. To this aim we apply the logarithmic operator to Σ_c and, since it maintains the ordering, we will be able to discriminate among the localization hypotheses.

$$A_c = \ln(\Sigma_c) = \sum_{p \in \Phi_c} ln \frac{(N-1) \cdot (1 - Pr[E_p])}{V_p - (1 - Pr[E_p])} + \sum_{p \in \Psi_c} ln \frac{(N-1) \cdot Pr[E_p]}{(N - V_p) - Pr[E_p]}. \quad (10)$$

A_c is the accumulator value, after the voting phase. From eq. (10) we can see that each perception must be associated to a couple of weights:

$$w_p = \begin{cases} ln \frac{(N-1) \cdot (1 - Pr[E_p])}{V_p - (1 - Pr[E_p])}, if p \in \Phi_c \\ \\ ln \frac{(N-1) \cdot Pr[E_p]}{(N - V_p) - Pr[E_p]}, if p \in \Psi_c \end{cases} \quad (11)$$

In order to compute the weights for each perception p, we have to know the values of N, V_p, and $Pr[E_p]$, for each resolution level. Moreover, we would like to know these values before starting the localization process. For this reason, the set P is partitioned in subsets g called *groups*, e.g. the group of the "blue direction perceptions", the group of the "white point perception between 50cm and 100cm" and so on. In the off-line phase, for each group and for each resolution level, the three values are estimated and the weights are computed by means of eq. (11). These values will then be used as an approximation of the actual (unknown) values. The rationale behind this is that all the perceptions in a group are quite homogeneous, so that their V_p and $Pr[E_p]$ values can be approximated by the group values, determined off-line. This process occurs at each resolution level since the weights change accordingly with it.

Definition 11. *N^l is the number of cells at level l.*

Definition 12. *V_g^l is the number of cells that a perception of the group g votes at level l.*

Definition 13. *$Pr[E_g^l]$ is the probability that a perception of the group g does not vote the cell containing the actual robot pose, at level l.*

The estimation of V_g^l is accomplished considering one perception $p \in g$ and testing its compatibility with all the cells at level l. Since the number of cells voted by a perception is independent of the sensors and time of acquisition, this estimation process can be carried out off-line. The same does not hold for $Pr[E_g^l]$. The value of this probability is a priori unknown, since it is related to the sensor that generates the perception, and, above all, it may change with time. At each level l, after the voting phase, $Pr[E_p^l]$ is estimated for each perception p, by considering the probabilities $Pr[(c = r)|\Phi_c, \Psi_c]$ for those cells that p has not voted. Since we know the value $Pr[(c = r)|\Phi_c, \Psi_c]$ only for the cells which have been analyzed (i.e. which resulted from the refinement of the cells selected al level $(l-1)$), and in order to normalize the sum of the $Pr[(c = r)|\Phi_c, \Psi_c]$, we propose to use the following approximation instead:

$$Pr[E_p^l] \approx \frac{\sum_{c \in \Xi_{\overline{p}}^l} Pr[(c = r)|\Phi_c, \Psi_c]}{\sum_{c \in \Xi^l} Pr[(c = r)|\Phi_c, \Psi_c]} = \frac{\sum_{c \in \Xi_{\overline{p}}^l} \frac{e^{A_c}}{e^{A_c}+N^l-1}}{\sum_{c \in \Xi^l} \frac{e^{A_c}}{e^{A_c}+N^l-1}} \tag{12}$$

where Ξ^l is the set of the cells analyzed at the current level l, and $\Xi_{\overline{p}}^l$ is the subset of Ξ^l whose elements are the cells that have not been voted by p. This approximation is justified because we do not analyze the cells with a low probability of being the correct cell. The estimate of $Pr[E_g^l]$ is computed as the average $\widehat{Pr[E_g^l]}$ of the $Pr[E_p^l], \forall p \in g$, which, by means of eq. (11), allows to determine the weights for the next iteration of the localization process.

4 Localization Experiments

We made some experiments with a robot equipped with an omnidirectional vision system, based on a multi-part mirror [7]. This mirror allows the vision system to have a resolution of about $40mm$, for distances under $6m$. We put the robot into a RoboCup middle-size league playground. We defined three map elements of type *line* and attribute *blue* to describe the blue goal and four lines to describe the borders of the half-field used. The vision sensor produces two types of perceptions: white point perception and blue direction perception.

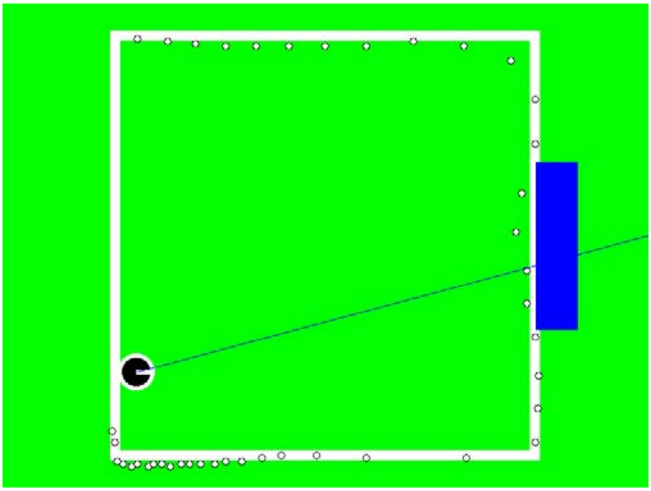

Fig. 1. White point and blue direction perceptions re-drawn on the map, altogether with the map elements; the black and white circle represents the actual robot pose

Table 1. Values of the positive weights for some group (columns) and for some level (rows)

Lev	Blue	White 0-100	White 100-200	White 200-300
1	0.00	0.00	0.00	0.00
5	0.38	0.00	0.00	0.01
10	1.37	0.34	0.49	0.74
15	1.93	1.63	5.49	5.75

Table 2. Values of the negative weights for some group (columns) and for some level (rows)

Lev	Blue	White 0-100	White 100-200	White 200-300
1	0.00	0.00	0.00	0.00
5	-1.35	0.00	0.00	-0.12
10	-2.05	-1.28	-1.50	-1.74
15	-2.16	-2.11	-2.30	-2.30

We devised a preliminary experiment for validating the definition of the weights. In table 1 and table 2, we have reported, respectively, the positive and negative weights associated with some group of perceptions at some levels of resolution. We can notice that perceptions begin (i.e. when considering large cells) with a zero weight for both positive and negative weights. The reason is that the localization hypotheses are too wide and the perceptions vote all the cells, bringing no information about the robot pose. On the other hand, as the resolution increases, we have that the positive weights grow, while the negative weights decrease their value. The values of table 1 and table 2 were computed following the procedure described in section 3.2, with the values of $Pr[E_g^l$ initialized to 0.1. In the experiment reported in Figure 1 we had several white point perceptions and only one blue direction perception (which coulb be outweighted by the white ones). If we had considered the white point perceptions only, we would have had ambiguities, because of the environment symmetries. In such case (all the perceptions have the same weight) the localization ends up with four distinct poses, at the same level of likelyhood: $(465cm, 143cm, 180^o)$, $(101cm, 220cm, 270^o)$, $(34cm, -143cm, 0^o)$, $(382cm, -229cm, 90^o)$. With the weighting mechanism here presented, the correct solution could be reached: actual value $(25cm, -150cm, 0^o)$, estimated value $(26cm, -138cm, -1^o)$.

References

1. Restelli, M., Sorrenti, D.G., Marchese, F.M.: Murea: a multi-resolution evidence accumulation method for robot localization in known environments. In: proceedings of the International Workshop on RoboCup - Robot Soccer World Cup VI, Springer-Verlag, LNCS series (to appear)
2. Restelli, M., Sorrenti, D.G., Marchese, F.M.: A robot localization method based on evidance accumulation and multi-resolution. In: proceedings of 2002(IEEE/RSJ) Int. Conf. on Intelligent Robots and Systems (IROS2002). (2002)
3. Burgard, W., Fox, D., Hennig, D., Schmidt, T.: Estimating the absolute position of a mobile robot using position probability grids. In: AAAI/IAAI, Vol. 2. (1996) 896–901
4. Gutmann, J., Burgard, W., Fox, D., Konolige, K.: An experimental comparison of localization methods. In: proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS98). (1998)
5. Lu, F., Milios, E.: Globally consistent range scan alignment for environment mapping. Autonomous Robots **4** (1997) 333–349
6. Dellaert, F., Fox, D., Burgard, W., Thrun, S.: Monte carlo localization for mobile robots. In: proceedings of the IEEE International Conference on Robotics and Automation (ICRA99). (1999)
7. Marchese, F.M., Sorrenti, D.G.: Omni-directional vision with a multi-part mirror. In: proceedings of the International Workshop on RoboCup. Volume 2019 of LNAI., Springer-Verlag (2001) 179–188 P. Stone, T. Balch, G. Kraetzschmar Eds.