Case-Based Indoor Navigation

Alessandro Micarelli and Giuseppe Sansonetti

Department of Computer Science and Automation
Artificial Intelligence Laboratory
Roma Tre University
Via della Vasca Navale, 79, 00146 Rome, Italy

Abstract. The purpose of this paper is to present a novel approach to the problem of autonomous robot navigation in a partially structured environment. The proposed solution is based on the ability of recognizing digital images that have been artificially obtained by applying a sensor fusion algorithm to ultrasonic sensor readings. Such images are classified in different categories using the well known Case-Based Reasoning (CBR) technique, as defined in the Artificial Intelligence domain. The architecture takes advantage of fuzzy theory for the construction of digital images, and wavelet functions for their analysis.

1 Introduction

In recent years, the problem of indoor robot navigation has been largely considered for the challenges that issues in several technological fields. From the motion control substrate to the artificial reasoning layer, many researchers have worked out solutions able to perform complex navigation tasks in many application fields ranging from industry to service robotics. In particular, a still open problem is the devising of efficient strategies able to cope with the problem of *self localization* in unstructured environments, i.e., the ability of estimating the position of the mobile platform when no artificial landmarks can be used to precisely indicate to the robot its position. Now, suppose to restrict the problem and choose the environment in a particular class, still very wide: an office-like environment with corridors, corners and other similar features. Suppose also that only low cost sonar sensors can be used: all localization information, that at this point have a topological character, should be easily extracted from sensory data and used to guide the platform along the path. Unfortunately, in a dynamic environment, those features (*natural landmarks*) can vary and some unknown configurations could be found leaving to the robot the choice on several strategies: one could consist in finding the nearest matching topological element in a static library; an other one could include a supervised learning stage in which the new pattern is used to increase the base library itself. This second approach is often referred as *Case-Based Reasoning (CBR)* [1, 5], and tries to catch all the learning opportunities offered both by the environment and, in an initial phase, by an external supervisor, to improve robot skill in analyzing its exteroceptive sensorial view.

Micarelli A. and Sansonetti G. (2007).
Case-Based Indoor Navigation.
In *Robot Vision*, pages 97-106
DOI: 10.5220/0002069300970106

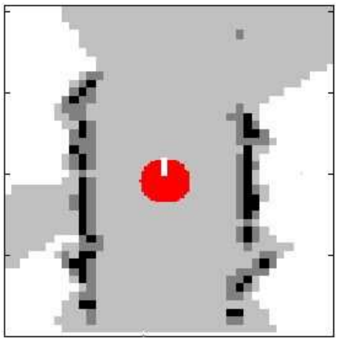

Fig. 1. Map of a corridor.

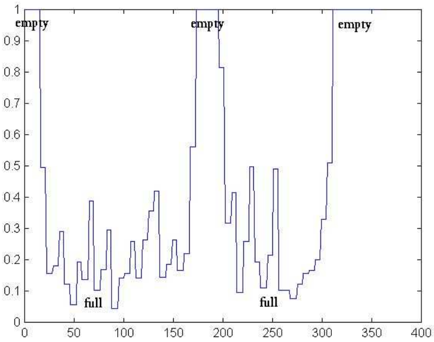

Fig. 2. Worldmark.

2 A Case-Based Approach

Autonomous navigation usually implies a recognition phase for each step taken by the robot to estimate its position, or better, to understand the particular shape of the environment (the topological feature) inside its actual range of view. In our case, this can be done comparing the actual sonar output with a set of reference signals associated with particular topological features. In most cases, association is done by comparing the actual view with a static list of models obtained with *a priori* considerations on the environment itself [4]. However, following a CBR philosophy, a learning approach can be devised in which real-world cases obtained from a supervised navigation are used to build and update a dynamic library. In this paper, we want to show how such a method can be successfully applied to help the robot during navigation in dynamic environments containing features that only partially correspond to previously known cases. In particular, the problem we intend to address concerns the recognition of a sonar-based digital image and its classification under one category belonging to a set of predetermined topological situations (Corridor, Corner, Crossing, End Corridor, Open Space).

Basically, the surrounding of the robot is represented in terms of *Fuzzy Local Maps (FLM)*, i.e., *Fuzzy Maps* [7, 8], that turned out to be extremely useful in many sensor fusion problems, obtained from a preprocessing stage applied to the sonar signals. Each FLM consists of 40 x 40 cells and, for each cell of an FLM, two values specifying the degree of membership to the set of empty cells and to the set of occupied cells are computed. An FLM, usually derived at each step merging the last n sets of collected data, is thereafter represented by two fuzzy sets: the empty cells set $\mathcal{E}$, and the occupied cells set $\mathcal{O}$. As an example, in Fig. 1 the $\mathcal{E}$ set of a FLM obtained in a corridor is reported. Different gray levels in the image represent different fuzzy values. Pixels with darker gray levels correspond to lower values of membership to the empty cells set $\mathcal{E}$, white pixels are unexplored regions, with a fuzzy value of membership to $\mathcal{E}$ equal to 0.

Now, with reference to the scheme depicted in Fig. 3, let us assume that the robot has acquired a new FLM. As first step, a feature-based representation of the new FLM

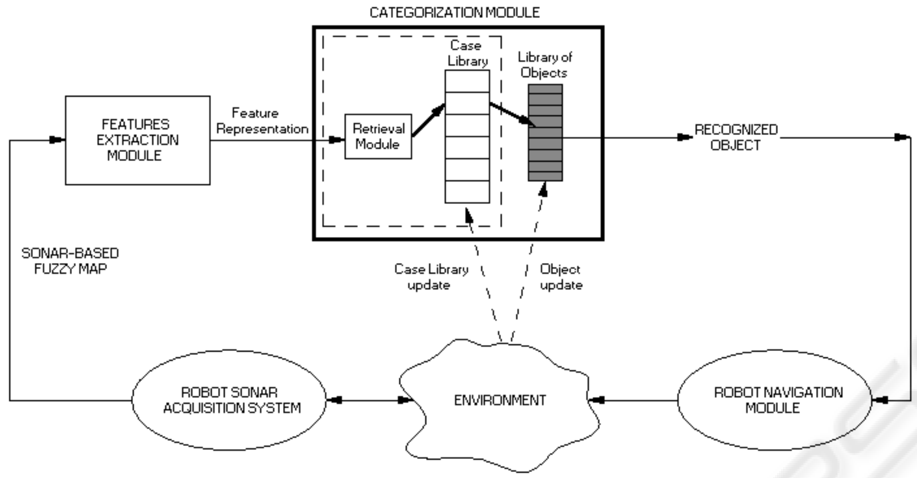

Fig. 3. Navigation architecture with Case-Based Reasoning.

is evaluated by the feature extraction module. This representation constitutes the "new case" of the proposed CBR system. The retrieval module shown in the figure will effect a search in the case library containing the old cases, based on a $<$*problem representation*, *solution*$>$ structure, which in this specific case will be $<$*FLM based representation*, *topological category index*$>$. The solution given in the old case can therefore be seen as a pointer to the "Library of Objects", containing the categories (i.e., "topological features") that could appear in the maps to be analyzed. The "recognized object" is at this point taken into consideration by the robot navigation system to plan its motion. This object, which constitutes the old solution of the case retrieved from the Case Library, will also be considered as a candidate solution of a new problem (basically, there is no need for an adaptation of the old solution to suit the new case) and if the human supervisor accepts it, the pair $<$*new FLM based feature representation*, *recognized object index*$>$ can be inserted as a new case in the Case Library.

3 Image Recognition

For sake of clarity, the pseudo-code of a rather simplified version of the classification algorithm is reported in Table 1. The complete solution, employed for the experimental performance assessment, was implemented in C language under the Linux operative system, for reasons of porting and efficiency. To handle both the new case and any of those cases dwelling in the *Case Library*, the use of a record structure comprising the three fields below was adopted:

– a one-dimensional fuzzy *worldmark* summarizing the content of the FLM;
– *object*, designed to store the label associated to the recognized object;
– *time*, reserved to the storage of information regarding the utility of the case of reference.

As indicated above, the first field is dedicated to the representation of the FLM. Specifically, in order to guarantee the applicability of the current approach to real-time control, a simplification has been introduced: the bi-dimensional fuzzy map of Fig. 1 is replaced with a one-dimensional fuzzy signal, named *worldmark*. The worldmark is computed by determining, for each direction around the robot, the value of the cell with the highest matching score to the set of empty cells, or, in other words, the cell for which the risk of belonging to a possible obstacle is minimum (see Fig. 2). Therefore the "new case" that appears in Fig. 2 consists of a vector of N elements (typically N=360) with values in the interval [0,1].

Before launching into the detailed description of the representation modalities of the aforementioned three fields, we believe it useful to provide a general overview of the entire algorithm. The domain expert's possibility to intervene in the decision task is possible both in the initial training phase of the system as well as during the verification phase for the retrieved solutions. Another aspect worthy of attention is the one related to the adoption of a double similarity test. It is manifest that as the pertinence of the case library increases, so does the probability of retrieving a candidate with a good value of similarity to the case under examination and, therefore, that the associated solution to will prove to be valid even in a contingent situation. On the other hand, a rather voluminous library presents the two following inconveniences:

– more time necessary for the retrieval of the required information;
– a depletion in terms of available space.

In order to avoid, at least partially, this state of affairs, the proposed architecture uses two different tests, respectively, named *reliability test* and *identity test*. The former provides indications on the possibility of successfully apply the solution of the retrieved case to the new situation, the latter controls the insertion of the new case into the system memory. The reason for the introduction of the identity test parameter is owed to circumstances where it is useless to include a new case, "quite" similar to a case stored in the library in the system memory. The reliability test is performed by comparing the current similarity metric value s_j with the reliability threshold S_a, while the identity test is performed by comparing the same value s_j with an identity threshold S_b. In Tables 2 and 3 the threshold values determined by a heuristic procedure are reported together with the percentage of coincidence between the responses given by the system and those furnished by a domain expert. Specifically, for the setup of S_a and S_b, the available memory space, the amount of resources necessary to keep in memory the pair $<$*representation of signal, represented object*$>$ and the statistics of the similarity index were considered.

Keeping in mind an "intelligent" management of the resources available to the system, a third test has been introduced. The idea that has, concretely, lead to its introduction, stems from the need to keep track, for all cases stored in memory, of the *frequency* of their appearance and the *effectiveness* of the solution associated to them. The record

Table 1. Pseudo-code for CBR.

```
Function REC(NewImage) returns RecObject
inputs : NewImage; the input image
variables : CaseLib; the case library
        $C_j$; the generic old case
        $T_{nouse}$; the inactivity time
        $S_a$; the reliability threshold
        $S_b$; the identity threshold
local variables : D.image; the image representation
        D.object; the recognized object
        $s_j$; the metric value
        tempvalue; the temporary metric value
        tempind; the temporary case index

D.image ← WAVELET(NewImage)
D.object ← 0
tempvalue ← 0
tempind ← 0
for each old case $C_j$ in CaseLib do
  begin
    $s_j$ ← COMPARE_CASE(D.image, $C_j$.image )
    if (tempvalue < $s_j$) then
      begin
        tempvalue ← $s_j$
        tempind ← $j$
      end
  end
  if (tempvalue < $S_a$) then
    begin
      D.object ← HumanExpertSolution
      $C_{n+1}$.image ← D.image
      $C_{n+1}$.object ← D.object
      $C_{n+1}$.time ← 0
    end
  else
  begin
    if ($C_{tempind}$.object = HumanExpertSolution) then
        begin
          D.object ← $C_{tempind}$.object
          $C_{tempind}$.time ← 0
        end
      else
        D.object ← HumanExpertSolution
    if (tempvalue < $S_b$) then
      begin
        $C_{n+1}$.image ← D.image
        $C_{n+1}$.object ← D.object
        $C_{n+1}$.time ← 0
      end
  end
CLEAN_LIB(CaseLib,$T_{nouse}$)
RecObject ← D.object
returns RecObject
```

field *time* was specifically introduced in consideration of these aims. Once more, the *clean library* test compares this value with a threshold T_{nouse}. If *time* exceeds T_{nouse} the case is removed from the dictionary. For the determination of the optimal value to assign to the indicator T_{nouse}, the same considerations expressed above for the parameters S_a and S_b still apply.

However, for a full understanding of the architecture proposed in this article there are still two major aspects that, as always, in any system based on cases, constitute the heart around which all the rest revolves, that is,

- the signal representation;
- the similarity metric.

These aspects are, furthermore, strongly interrelated.

3.1 The Signal Representation

Choosing the most efficient representation for a current problem constitutes the crucial moment of any application of signal processing. Here, we resorted to a *wavelet representation* of the worldmark. The wavelet representation expresses the signal of interest as superimposed elementary waves and, therefore, in this respect does not introduce any innovation compared to traditional methods, such as Fourier series expansion. However, the innovative aspect offered by wavelet functions consists in the possibility of subdividing the available data in components with differing bandwidths and time durations. Each of these components is subsequently analyzed by a resolution associated to its scale. The advantages offered by this procedure are tangible, above all, in respect to the analysis of physical situations where typical signals show discontinuity and sudden peaks, exactly as happens with worldmarks. The advantages of adopting representations in similar situations through wavelet functions, instead of traditional methods, are extensively expounded in the literature [3, 6, 2].

3.2 The Similarity Metric

The last aspect to be examined concerns the choice of the metric necessary for the evaluation of the *similarity* existing between case f in input and the generic case g belonging to the *Case Library*. Regardless of the application context, a good metric must anyhow be able to guarantee an efficient compromise between the two main requisites, which are the *quality* of the recognition and the *computational complexity*. Accordingly, during the experimental activity several different metrics were tested. Among them, the relatively best results were obtained by using the *cross-correlation factor* as metric, whose expression is:

$$\underset{\theta\in[0,2\pi]}{Max} \frac{\langle f(x), g(x-\theta)\rangle}{\sqrt{\langle f(x), f(x)\rangle \langle g(x), g(x)\rangle}}$$

This quantity was calculated both in the time and frequency domains, respectively, obtaining in both cases significant results with moderate processing time, through computation resources available on the market today.

4 Experimental Results

For our tests, we used the simulator of Nomad200 by Nomadic Technologies, a mobile robot equipped with a ring of 16 equally spaced ultrasonic sensors. The procedure consists of tracing a number of global maps of hypothetical office-like environments, simulating the robot dynamics and, finally, collecting the output data. For these operations we used the real time navigation software A.N.ARCH.I.C. [9] which, together with the aforementioned simulator, made the robot virtual navigation inside the mapped environment possible producing the sequence of FLMs and corresponding worldmark, each pair related to a different position taken during the followed path. Each sequence, therefore, includes hundreds of FLMs and worldmarks, which constitute the input for the tests that we performed on our classifier. The values reported below were obtained by using a machine equipped with a Pentium M processor, 1700 MHz, and 512 MB RAM. During the testing phase, we initialized the system through representations related to four different configurations:

- corridor
- crossing
- end of corridor
- angle

providing, for each of these, three different standard schemes, in practice as it appears in the initial phase, at its basic level, and in the final phase.

Tables 2 and 3 show, in particular, the results recorded during two different series of tests of the system. The first illustrates the results obtained by performing the similarity evaluation between the input signal and the generic one inside the case library directly in the temporal domain. Instead, for the second one, the same operation was effected in the wavelet domain, i.e., the matching evaluation of the two signals was not made by estimating the cross-correlation between sequences of temporal samples, but between the corresponding residual low-frequency components, obtained through Discrete Wavelet Transform (DWT). Consequently, it is possible to appreciate in a more tangible way the extent of the possible advantages granted by the expansion of signals in series of waveform, perfectly located in time and in frequency.

To perform this experimentation, we simulated the robot navigation in an environment that Fig. 4 illustrates as a global map. In the same figure we have also traced the path followed by the robot, planned on the basis of specific methods for which further explanation is out of the scope of this paper. A sequence of 636 FLMs is thus generated, as well as a corresponding number of worldmarks. In order to streamline the experimental procedure, without, however, penalizing its efficiency, since the variation between one FLM and the subsequent one was practically insignificant, we decided to consider only one over three samples and to discard the others. As a result, the map effectively input to the system consists of only 212 FLMs.

Initially, we shall examine the values reported in Table 2. As anticipated earlier, the tests were performed by running the system beforehand through the same training session, for each test series. This fact becomes apparent by looking at the data in the 6^{th} column, since the same value recurs systematically in each line (12 cases). Actually, the coincidence does not only concern the number of cases used, but also the samples

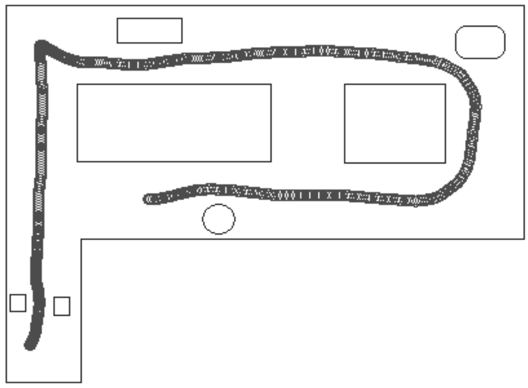

Fig. 4. Global map.

themselves. In this way, we attempted to guarantee the same initial condition in each test series.

A reading of the data discloses the consistency of the recorded fluctuations, in respect to the varying values assigned to the two similarity thresholds. For example, it is noticeable that when the reliability threshold S_a decreases, there is a proportional decrease in the number of interventions required of the domain expert by the system. Similarly, there is a clear increase in the number of cases inserted in the relative library matching an increase in the identity threshold S_b. However, the phenomenon of major importance and interest relates to the trend recorded by the factor indicated in the table as *coincidence percentage*. This factor was gathered by a comparison between the system responses and those that would have been given by the same expert who performed the training, when examining the corresponding FLM. Clearly, such a strategy is inevitably damaged by the loss of information that occurs during the passage from a bi-dimensional fuzzy map (FLM) to the corresponding polar map (worldmark). However, notwithstanding this additional source of uncertainty, the results obtained may be considered more than satisfactory.

Proceeding with the analysis of the data reported in Table 3, which refer to the same experimental tests, but performed on the wavelet coefficients and not on their corresponding original signals, the gain is noteworthy, both in terms of coincidence percentage as well as computational complexity. In particular, it can be observed how the first factor is affected to a significant lesser degree by the variation of the values assigned to the two thresholds S_a and S_b.

Although we do not wish to dwell upon too many details of the experimentation, it should be noted, however, that to obtain the wavelet coefficients relating to sequences of 360 temporal samples we used a simple DWT with four levels and analysis filters of the type belonging to the Daubechies family (specifically, the version with four coefficients).

Another observation should be made on the processing time. In order to finalize this experimentation, for sake of clarity, we decided to operate on the group of worldmarks

Table 2. First experimental set.

S_a	S_b	Input cases	Expert interventions	Coincidence percentage	Cases before	Cases after	Processing Time (s)
0.90	0.93	212	14	84.0% (178)	12	51	10.31
0.88	0.93	212	12	84.0% (178)	12	51	10.16
0.85	0.93	212	5	81.6% (173)	12	51	9.46
0.91	0.93	212	23	94.3% (200)	12	51	10.46
0.91	0.95	212	19	94.3% (200)	12	74	13.86
0.89	0.91	212	14	88.7% (188)	12	37	8.21

Table 3. Second experimental set.

S_a	S_b	Input cases	Expert interventions	Coincidence percentage	Cases before	Cases after	Processing Time (s)
0.90	0.93	212	12	94.3% (200)	12	43	0.47
0.88	0.93	212	7	93.9% (199)	12	43	0.53
0.85	0.93	212	5	92.4% (196)	12	43	0.43
0.91	0.93	212	15	94.3% (200)	12	43	0.48
0.91	0.95	212	13	94.3% (200)	12	64	0.59
0.89	0.91	212	12	94.8% (201)	12	51	0.38

generated during the course of the overall navigation inside the simulated environment. Consequently, the time necessary to operate in real-time is decidedly less than that reported as the sum over all input cases in the table and, above all, significantly lower than the time allowed during the robot actual navigation.

5 Conclusions

Generally, the normal pattern recognition techniques require models of the objects that must be recognized and classified. The collection of models available to the classifier clearly reflects the original knowledge of the situation to be analyzed. However, in most cases, as for the robot autonomous navigation, there exists practically no prior information whatsoever. Our proposed architecture includes a feature extraction algorithm incorporated into a CBR shell, which allows a constant increase in the knowledge of the surrounding environment. We remark, however, that the possibility of updating the Object Library as well as the Case Library, although left open and in principle with no limit to the number and complexity of information that may be collected, is constrained to real-time restrictions linked to the technology that is available on the market today.

Future developments will be focused on introducing the possibility of fusing more information coming from different kind of sensors (e.g., laser scanners or cameras) into a more detailed worldmark to provide the classifier with a better and more robust input data.

References

1. Aamodt, A., and Plaza, E. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. *AI Communications*, **7**(1), pp. 39–59, 1994.
2. Chui, C.K. *An Introduction to Wavelets*. Academic Press, London, England, 1992.
3. Daubechies, I. Orthonormal bases of compactly supported wavelets. *Comm. Pure Appl. Math.*, **41**(7), pp. 909–996, 1988.
4. Fabrizi, E., Panzieri, S., and Ulivi, G. Extracting topological features of indoor environment from sonar-based fuzzy maps. *Proceedings of the 6th International Conference on Intelligent Autonomous Systems*, Venice, Italy, 2000.
5. Kolodner, J. *Case-Based Reasoning*. Morgan Kaufmann, San Mateo, CA, 1993.
6. Mallat, S.G. A Theory for Multiresolution Signal Decomposition: the Wavelet Representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(7), pp. 674–693, 1989.
7. Oriolo, G., Vendittelli, M., and Ulivi, G. Real-Time Map Building and navigation for Autonomous Robots in Unknown Environments. *IEEE Transactions on Systems, Men and Cybernetics - Part B: Cybernetics*, **28**(3), pp. 316–333, 1998.
8. Panzieri, S., Petroselli, D., and Ulivi, G. Topological localization on indoor sonar-based fuzzy maps. *Intelligent Autonomous System*, IOS Press, Amsterdam, Netherlands, pp. 596–603, 2000.
9. Panzieri, S., Pascucci, F., and Petroselli, D. **A**uton. **N**avigation **ARCH**itecture for **I**ntelligent **C**ontrol, *www.dia.uniroma3.it/autom/labrob/anarchic*, 2002.