Memory-Based Sensor Integration

B. Crespi, D. Giordani & L. Stringa

RWCP[1)] *IRST*[2)]
I-38050 Povo, Trento, ITALY
e-mail crespi@irst.it;
Key word: Sensor Integration, Memory-based Approach, Real World Computing

Abstract

A sonar array and a camera are the sensors of a robot that learns to move inside a building. Learning is based on the association of ultrasonic and visual images with motor commands. The two sensors extract different information from the environment. Sonars produce a short-range depth image on the sonar array plane, vision gives a wide-range flat image of the space. In this work, a memory-based probabilistic framework is developed that integrates the information gathered by the sensors.

1 Introduction

The interaction of a complex organism with the environment is always based on multiple sensors. Different sensors are independent channels through which the system explores the surrounding world. Though independent and complementary, sensors acquire information on the same external structures. This characteristic leads to a robust, flexible and rich representation of the world, which is a typical, complex feature of biological systems.

One of the most interesting and difficult tasks in robotics is to attain the efficacy of the biological integration process of sensory information. The sensory system may be seen as a "sensory manifold" composed of different sensory areas patched together. An idea or a concept is a function globally defined on the manifold, that has a local, specific representation in each sensory area. Each area has its own intrinsic metrics that is a criterion of similarity between stimuli. Different sensory areas are not independent. A correlation between sensors is established in those regions in which information on the same structure is represented. The whole, integrated information forms the global, sensory representation of the external world.

Given the statistical nature of sensory data, the probabilistic language is the appropriate mathematical framework to describe sensory information processing. A state function over the "sensory manifold" represents the probability that a concept corresponds to a given (multi-sensor) stimulus. Different sensors contribute differently in concept representation. If the discrimination capability is superior in a particular area, the weight given to this area is greater. In other cases, several areas contribute in a similar fashion to classification.

The primary task for sensor fusion is to appropriately use the variety of information and to weigh the different sensorial contributions to achieve the most discriminating and descriptive representation [1, 2, 3]. In our case, for indoor navigation, the "sensory manifold" is composed of two areas, the sonar and the vision area (see Fig. 1). A sonar array is a short range, low-information-density sensor, useful to avoid obstacles and to move in crowded spaces. Vision is a wide range, high-information density sensor that is efficient when a well defined perspective is available. The sonar system is expected to take over motor control in a small or crowded room. On the contrary, vision is predominant in open spaces, where its wide range is able to encompass the whole structure of the environment and where the high density of information of the visual sensor can be used for other purposes (for example, detecting moving objects or landmarks).

This paper presents an application of a memory-based approach to multi-sensor information processing and learning for autonomous reflex navigation. Memory-based reasoning [4, 5, 6] is

[1]Real World Computing Partnership
[2]Istituto per la Ricerca Scientifica e Tecnologica

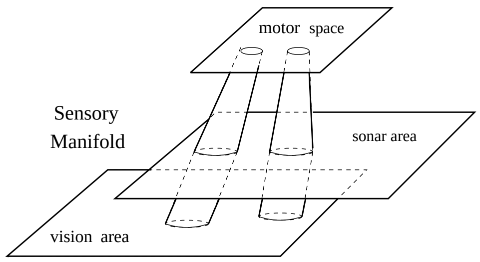

Figure 1: The sensory manifold is composed of two sensory areas. The two areas are related when the same motor command is associated with visual and ultrasonic stimuli.

a general framework for sensory information processing. Processing is based on Bayesian treatment of a large number of features; the fundamental operation is a statistical comparison between sensory inputs (images and ultrasonic pictures) and memorized perceptions of the environment.

The paper is organized as follows. Section 2 introduces the memory-based procedure used for information processing and learning. In Sections 3 and 4 the procedure is applied to sonar and vision processing. In the last sections, experiments on sensor integration and learning are presented and discussed.

2 A Memory-based Learning Procedure

Learning from examples is one of the most frequently used learning procedures. The system is given a set of examples, i.e. a set of input-output pairs, and is expected to generalize this information to other unseen inputs. The capability of a system to learn depends on two crucial factors. The first is the representativeness and completeness of the set of examples. The second is the definition of a similarity criterion that allows one to classify unseen inputs on the basis of their similarity with the given examples.

A reactive system, as the one considered in this paper (sensors $\rightarrow$ motors), is a function between the input space, i.e. the "sensory manifold" and the output space, i.e. the space of motor commands. In learning, the function is constructed using for each sensor a set of examples, i.e. a set of pairs (image, command). The generalization capability depends on the property of the mapping to preserve similarity.

The assignment of commands to input configurations is not a deterministic process. During learning, the same configuration can be associated to different commands. This is due to the variability of learning conditions. It is assumed that variability is a stochastic process described by a probability distribution. In order to estimate the distribution, a command is not explicitly associated to an image, but a frequency matrix is constructed that memorizes the frequency of a given association. The frequency of a determined association is interpreted as the probabilistic distribution underlying the process. An image, ultrasonic or optical, is mapped to a probability density over the motor space.

$$\Phi \ : \quad image \ \rightarrow \ P(\eta) \quad where \ \eta \in Motor\,Space \tag{1}$$

The probability density is constructed in terms of a large set of elementary probabilities corresponding to the chosen features. Two images are similar if they contain the same features. If features are assumed to be mutually independent the total probability is given by the product of the elementary probabilities.

3 Sonar Processing

The sonar configuration is sketched in Figure 2. Fourteen Polaroid sonars are used to cover the front and the sides of the robot. Each sonar has a preferential direction and a detection cone of about 30 degrees around this direction. The sonar array produces a 2D ultrasonic image (distance and direction) on the array plane around the robot. To navigate, the robot associates the present sonar input with a motor command.

Motor commands are points in a 2D space, whose coordinates are steering velocity and driving velocity. The motor space is discretized. If V is the maximum driving velocity and W is the maximum steering velocity, driving velocities are given by

$$v_i = \frac{i}{4} \cdot V \qquad i = 0, 1, 2, 3, 4 \tag{2}$$

and steering velocities are

$$\omega = \frac{i}{4} \cdot W \qquad i = -4, -3, -2, -1, 0, 1, 2, 3$$

In the following a discrete command will be denoted by $\eta_k \equiv (v_i,\ \omega_j)$.

Ultrasonic images are encoded using the following procedure. Distances measured by sonars are *localized* by a sigmoidal function

$$l_s \ = \ Q \cdot \sigma(\xi_s, a)$$
$$\sigma(\xi_s, a) = \frac{2}{1 + exp\,(-\xi_s \cdot a)} - 1 \tag{3}$$

where $\xi_s = d_s/D_s$ and $a = 3$. d_s is the distance measured by sonar s and D_s is the distance between sonar s and the border of the sonar

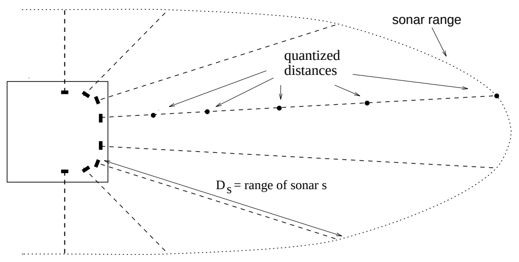

Figure 2: Sonar system. The sonar range is elliptical. Dashed lines represent sonar directions. For better visualization, the picture is not in scale. The robot's side is 50 cm, the range of lateral sonars is D=100 cm and the range of frontal sonars is D=250 cm.

area along its preferential direction. The localized distances l_s are quantized using Q values, $l_i \rightarrow q_i \in [0, Q-1]$.

Consider the probability

$$\wp(\eta \mid q_1, \, q_2 \, , ..., q_{NS}) \tag{4}$$

that a sonar input with discretized values $q_1, q_2, ..., q_{NS}$ for sonar $s = 0, 1, ..., NS$ respectively is associated to the given motor command η. The direct estimation of the probability density (4) is not possible because of the high number of variables. The number of possible configurations after quantization is still too high (Q^{NS}) to allow a direct association with motor commands. The probability density is approximated by means of lower-dimension densities. Pairs of sonars are considered. For NS sonars there are $\frac{NS \times (NS-1)}{2}$ pairs and each pair has $Q \times Q$ possible patterns. The quantities

$$\Phi_p(code_p \mid \eta) = \sum_m^{N_\eta} \alpha_r(i) \; \alpha_s(j) \tag{5}$$

where p represents pair (r, s), $code = i*Q+j$, and index m runs over the set of examples associated with command η, represent the number of times sonars r and s with values i and j respectively, are associated to a given command η. The variables α_s are binary variables, $\alpha_s(q)$=1 if sonar s has value q, otherwise it is zero.

The corresponding probability densities are

$$P_p(\eta_k \mid code_p) = \frac{\Phi_p(code_p \mid \eta_k) \; P(\eta_k)}{\sum_j \Phi_p(code_p \mid \eta_j) \; P(\eta_j)} \tag{6}$$

The *a priori* command probabilities are assumed to be equal, $P(\eta_k) = 1$ for every k.

It is supposed that patterns are mutually independent. Under this assumption, the total probability (4) can be written as a product of the "elementary" probabilities (6).

$$\wp(\eta_k \mid G) = \prod_{p \in Pairs} P_p(\eta_k \mid code_p) \tag{7}$$

where $k = 1, 2, ..., N_{commands}$, index p ranges over all the sonar pairs and $code_p$ is the code associated to pair p.

The frequency matrices (5) are constructed during learning. Learning is performed by an operator that drives the robot in the building. To avoid overlearning of safe driving conditions, crucial situations and associated commands are repeated several times. In practice, learning is done only when the operator enables it (pressing a button). In this way the operator can put the robot in an unsafe situation (for example, pointing toward a wall) and it memorizes the escape commands.

During performance, the robot response to the present sonar input is given by

$$v = \sum_{i,j} P(v_i \, , \omega_j) \; v_i \tag{8}$$

$$\omega = \sum_{i,j} P(v_i \, , \omega_j) \; \omega_j$$

where the indices i and j run over the motor command space.

4 Vision Processing

The visual input is processed along the lines of Ref. [5]. Inputs from the camera are gray-level images that are encoded by the system. In the learning phase, an encoded input image is associated with a motor command and memorized. During performance, the present image is compared with memories. System output is a probability distribution that determines the classification of the input image.

An input image, described by a matrix $G(x, y)$ of 128×128 pixels, is covered by N_s patches S_α of fixed size $K \times K$ (see Fig. 3). These patches may or may not be overlapping. The pixels of a patch form a 2D pattern that is mapped into a code. In this representation, an image G is described by a code vector $code(S_\alpha)_\alpha$, where $1 \leq \alpha \leq N_s = (\frac{128}{K})^2$.

The visual information contained in the image is decomposed into elementary patterns and then mapped into the code space. The encoding function quantizes the local dominant directions, i.e. the dominant directions of the edges contained in the patches S_α. In this application, the direction of the gradient of the image intensity is discretized in eight directions (see Fig. 4). In every patch

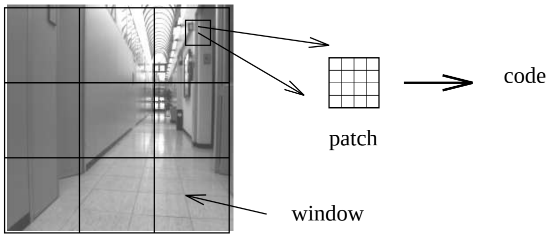

Figure 3: Image processing. Images are covered by patches. Each patch generates a code. Codes of patches belonging to the same sector are grouped together.

an activation vector is constructed by adding, for each of the eight directions, the magnitudes of the intensity gradient at each pixel,

$$A_i = \sum_{x,y \in patch} \mid \nabla I(x,y) \mid_i \tag{9}$$

where $\mid \nabla I(x,y) \mid_i$ is the magnitude of the gradient for direction i at pixel (x,y). The quantization of the activation vector produces the code.

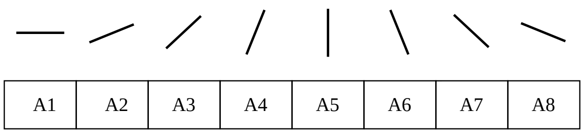

Figure 4: Local direction quantization. In each patch, the intensity gradient is quantized in eight directions. An activation vector (A1, A2, ..., A8) is constructed that represents the strength of the eight directions.

During learning, images are associated with motor commands. These associations are memorized in the frequency matrices $\phi_\alpha(code_p \mid \eta_k)$ which represent the number of times pattern $code_p$ has appeared in images associated to a certain η_k in patch S_α.

The image G is divided into larger subregions called *windows*. For example, in Figure 3, the image is divided into nine windows, $W_1, W_2, ..., W_9$. Patches belonging to the same window are grouped together and the corresponding ϕ_α are added,

$$\Phi_w(code_p \mid \eta_k) = \sum_{S_\alpha \in W_w} \phi_\alpha(code_p \mid \eta_k) \tag{10}$$

where $w = 1, 2, ..., N_{windows}$, $k = 1, 2, ..., N_{commands}$, and p ranges over all the possible patterns. The frequency matrices Φ_w are the memory of the system. If the number of windows is large enough, the procedure can be seen as a phase-space coding of the visual information, i.e. a mixed representation of the image, where position and direction are considered equally.

During performance, the command associated to the present image is found by computing the Bayesian probabilities

$$P_w(\eta_k \mid code_p) = \frac{\Phi_w(code_p \mid \eta_k) \cdot P(\eta_k)}{\sum_j \Phi_w(code_p \mid \eta_j) \cdot P(\eta_j)} \tag{11}$$

for every pattern that appears in the image. The *a priori* command probabilities $P(\eta_k)$ are assumed to be equal. It is supposed, for simplicity, that patterns are mutually independent. The image G is associated with the motor command that corresponds to the expectation value of the total probability density

$$\wp(\eta_k \mid G) = \prod_w \prod_{p \in W_w} P_w(\eta_k \mid code_p) \tag{12}$$

where $k = 1, 2, ..., N_{commands}$ and p ranges over all the patterns that appear in window W_w of the input image.

Vision learning is performed the same as for sonar learning. Since monocular vision uses the slopes of the edges to estimate the relative position of the robot, the vision system cannot be expected to be able to discriminate images that do not present a well-defined perspective.

5 Sonar and Vision Integration

The information the two sensors acquire from the environment is quantitatively and qualitatively different. The sonar array scans a slice of the 3D space around the robot. It produces a 2D image, whose coordinates are direction and distance on the sonar array plane. The direction resolution is coarse, because of the wide detection cone and the limited number of sonars. Since angular resolution decreases with distance, the sonar range has to be kept short. The information content of an ultrasonic image is explicit and direct, and can be analyzed quickly. Because of these characteristics, sonars are efficient for navigation in crowded places, where it is necessary to detect objects within a short range and to respond quickly.

Vision is a high-information density, wide-range sensor. The optical high resolution offers a detailed description of the whole 3D space around the robot. Information is complex and not directly accessible. In monocular vision, distances are not contained explicitly in images but have to be estimated from the projective geometry of the space. The procedure requires a global analysis of the image and a longer reaction time. Therefore vision is used when the robot is in open space.

The processing method used by the two sensors is based on the same memory-based method. In particular, the outputs are functions over the same motor space. This feature allows an immediate integration of the two sensors. Integration is performed by adding the probability densities computed by the sonar and vision modules,

$$P = w_s \cdot P_s + w_v \cdot P_v \tag{13}$$

where the weights w_s and w_v represent the relative importance attributed to sonar and vision.

The weights are chosen to be functions of the minimal distance $\xi_i = d_i/D_i$ defined in eq. (3),

$$w_s = exp\left[-\left(\frac{d_i}{D_i}\right)^2\right] \tag{14}$$

$$w_v = 1 - w_s .$$

The above definition implies that the robot uses prevalently sonars when it is close to an obstacle and vision when the minimum distance is close or beyond the sonar range border.

The chosen motor command is the expectation value of driving and steering velocity with respect to the probability density (13)

$$v = \sum_{i,j} \left[w_s \cdot P_s(v_i, \omega_j) + w_v \cdot P_v(v_i, \omega_j)\right] v_i$$

$$\omega = \sum_{i,j} \left[w_s \cdot P_s(v_i, \omega_j) + w_v \cdot P_v(v_i, \omega_j)\right] \omega_j . \tag{15}$$

6 Experiments

The robot used for the experiments is a mobile platform (see Fig. 5) developed at IRST. Robot motion is controlled by setting the values of the steering (angular) and the driving velocities. The size of the robot is 50×50 cm. The on-board processor is a 486 DX2 CPU (66 MHz). The sensors are a gray-level camera and 14 Polaroid sonars arranged as shown in Figure 2.

The corridors of the institute are approximately two meters wide and contain plants, xerox machines and other obstacles. They are illuminated through windows and transparent ceilings.

As previously discussed, monocular vision is reliable if a well defined perspective is present, for example in a corridor. Vision starts to fail when images do not contain enough geometric information on the environment, for example when the robot faces a wall or is in a corner. In general this happens when distances are small. Vision learning experiments were performed and the results

Figure 5: The robot.

are similar to those obtained in [5]. The difference is that here the association is between image and command instead of between image and position.

Sonars provide a direct measurement of distances between the robot and walls or obstacles. They are more effective than vision for navigation when perspective is hidden and indispensable to move in crowded spaces and to detect near obstacles. As explained in Section 3, the sonar range is limited to an elliptical shaped area around the robot. The range is kept short to allow encoding of the large number of distance-direction configurations necessary to deal with complicated spatial situations. Using sonars the robot is able to move along corridors avoiding near-by obstacles.

The integration of the two sensors improves the robot's performance in respect to the single sensor case. In particular vision makes the trajectory of the robot more stable when a good perspective is present (vision smooths out the discontinuities due to sonar measurements). In a simple environment, for example a corridor with large obstacles, the robot trajectories are very similar to the ones traced by the robot guided by the operator.

In crowded and complicated situations, i.e. with many obstacles, the behavior of the robot is more discontinuous. This is because the operator is able to devise a strategy to overcome a complicated spatial situation, while the robot responds only on the basis of the present input.

7 Discussion

A multi-sensor system can be seen as a "sensory manifold", composed of different but connected sensory areas. Different sensory areas can be considered local metric spaces in which distance measures similarity between stimuli. A "concept" is a probability density over the manifold that receives contributions from each sensory area. Perception of the same structure determines connections between areas. For example, sonar images and optical images associated with the same motor command have the same (or similar) meaning with respect to an action. As a result of this task-oriented topology, perception from different sensors can be easily integrated.

This paper presented a limited implementation of this idea in autonomous navigation, but the method can be extended to more complex systems. It could also be extended to include subareas, in which specific independent features of the same sensor are separated. For example, as in the visual cortex, visual information can be separated into form, movement, colors, etc, that are subsequently integrated.

The system described here is a reactive system, the response depends only on the present input, i.e. on the present situation. This sets a serious limit on the capability of a reactive system to cope with complex situations that require strategies. In a bottom-up approach, the limit could be overcome by keeping a temporal log of the sensory information gathered by the robot, i.e. by considering time sequences of recordings to create a map of the environment. The sensory input is used not only to execute a motor command, but to enlarge or refine its knowledge base. In this way, a single sensory datum, which is naturally partial and often ambiguous, can be linked to a more complete set of information. These topics will be considered in a future work.

References

[1] Gregory D. Hager. Task-Directed Sensor Fusion and Planning. *Kluwer Academic Publisher* (1990).

[2] *Intelligent Autonomous Systems IAS-3.* Proceedings of the International Conference, Pittsburgh, Pennsylvania 1993 IOS, Press.

[3] T. Coianiz and M. Aste and B. Caprile, *Kalman Filtering for Multi-Sensor Data Fusion: an Application to Autonomous Navigation*, IRST, Technical Report 9307-14, July 1993.

[4] C. Stanfill and D. Waltz. *Towards memory-based reasoning.* Comm. of the ACM, 29:1213–1228, (1986).

[5] B. Crespi, C. Furlanello and L. Stringa. A Memory-based Approach to Navigation. *Biological Cybernetics* 69, 385-393 (1993).

[6] L. Stringa. *S-nets: A Short presentation.* In E. R. Caianiello, editor, Proceedings of the Second Italian Workshop on Parallel Architectures and Neural Networks World Scientific Publishing Co., 177–184, (1990),