Scene Modelling from 2D and 3D sensory data acquired from natural environments

Rafael Murrieta-Cid* Carlos Parra† Michel Devy‡ Maurice Briot‡

* ITESM Campus Ciudad de México
Calle del puente 222, Tlalpan, México D.F.
e-mail: rmurriet@campus.ccm.itesm.mx

†Pontificia Universidad Javeriana
Cra 7 No 40-62 Bogotá D.C., Colombia
e-mail: carlos.parra@javeriana.edu.co

‡Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS-CNRS)
7, Avenue du Colonel Roche, 31077 Toulouse Cedex 4, FRANCE
e-mail:{michel, briot}@laas.fr

Abstract -*This paper presents perceptual functions used to build a scene description from sensory data acquired in a natural environment. A mobile robot which must execute motions on such environments needs geometrical representations (ground surface,...). This knowledge can be improved by using additional information, such as color and texture. From every acquisition our modelling method builds a scene model in which (1) a hybrid segmentation algorithm provides a synthetic image description in terms of regions, and (2) each region is characterized and afterwards identified to obtain its nature (grass, rocks ...) by the use of probabilistic methods. By using the model of the environment presented here, robot visual navigation is proposed based on landmark selection.*

Keywords -*Vision, Segmentation, Color, Texture, 3D, Classification, Robot Visual Navigation.*

1 Introduction

This paper concerns the perception functions required by a mobile robot which must execute motions in outdoor environments. The classical methods proposed for such an application are basically focused on the 3-D information obtained from a laser ranger finder or a stereoscopic system [3, 6]. However, depth information is not enough to get a complete environment description. Other information given by an intensity or a color image allows to identify the nature of the elements found in the scene. This paper describes the functions required to provide such a scene description from 2D and 3D sensory data.

Our approach consists in the following steps executed in sequence:

- region extraction: first, the available images are segmented to obtain the main regions of the scene,
- region characterization: each region is characterized by a vector of 2D or 3D attributes,
- region identification: probabilistic methods are used and compared to determine the nature of each region, in order to identify the elements perceived in the environment.

This approach is applicable to different types of attributes (2-D or 3-D) and environments. In spite of classical statistical methods have been used, they are used in an innovate manner.

For several reasons, it is better to perform the interpretation of the scene in different steps by using different attributes in each one. Image regions corresponding to areas of the environment close to the sensors (in our robot, up to 5 meters) can be analyzed by using 3D and luminance attributes. In these areas the stereo-vision gives a valid information. The segmentation performed from 3D attributes delimits the regions where the 3D information is valid. Regions corresponding to areas far away from the sensor (beyond 5 meters) will be analyzed by using only the color and the texture attributes given that 3D information is not available or too noisy. For these areas, since color is a point-wise property of images and the texture involves a notion of spatial extent (a single point has no texture), the color segmentation usually gives a better compromise between the precision of region borders and the speed of computation than the texture segmentation; consequently, we decided to use the color instead of the texture to achieve the segmentation step. In order to obtain a robust color segmentation a neighborhood around points can be used in color too (2×2 or 4×4 pixels), but it can be significant smaller than that for texture.

Having done the segmentation, both texture and color features are used to characterize and to identify the image regions. In this step, texture is taken into account in order to profit from its power of discrimination. The texture and color features are associated globally with the regions provided by the segmentation step on 3D or color images; this strategy generally gives a more discriminative information than these same features calculated from an arbitrary division of the image.

Moreover, we profit by the cooperation between the segmentation and classification steps, so that the result of the first step can be checked by the second one and, if necessary corrected. Over-segmentation is corrected by classification, identification errors can be corrected by using contextual information of the environment.

The texture attributes are based on histogram analysis. The histograms change gradually in function of the view point, the distance from the sensor to the scene and the occlusions [14]. This characteristic is interesting in the field of mobile robotics where such situations happen. Given thus, if the acquisition conditions are rather stable (especially constant illumination), the number of data samples required to represent different elements that we want to identify can be reduced.

2 The segmentation method

Our segmentation algorithm is a combination of two techniques: the characteristic feature, thresholding or clustering, and the region growing. The method does the grouping in the spatial domain of square cells. Those are associated with the same label defined in an attribute space (i.e. color space). The advantage of this hybrid method is that it allows to achieve the process of growing independently of the beginning point and the scanning order of the adjacent square cells.

The division of the image into square cells provides a first arbitrary partition (an attribute vector is computed for each cell). Several classes are defined by the analysis of the attribute histograms, which brings the partition into the attribute space. Thus each square cell in the image is associated with a class. The fusion of the square cells belonging to the same class is done by using an adjacency graph (adjacency-4). Finally, the regions which are smaller than a given threshold are integrated into an adjacent region.

In previous works the classes were defined by detecting the principal peaks and valleys in the histogram [4]. Generally, it is possible to assume that the bottom of a valley between two peaks can define the separation between two classes. However, for complex pictures, it is often difficult to detect the bottom of the valley precisely. Several problems prevent us from determining the correct value of separation: the attribute histograms are noisy, the valley is often flat and broad or the peaks are extremely unequal in height. Some methods have been proposed in order to overcome these difficulties [8]. However, these techniques require considerably tedious and sometimes unstable calculations. We have adapted the method suggested by Otsu [7], which determines an optimal criterion of class separation by the use of statistical analysis. This approach maximizes a measure of class separability. It is quite efficient when the number of thresholds is small (3 or 4). But when the number of classes increase the selected threshold usually become less reliable. Since we use different attributes to define a class, the above problem is avoided. In his method, Otsu deals only with a part of the class determination problem. It determines only the thresholds corresponding to the separation for a given number of classes. Our contributions are:

- the partition of the attribute space which gives the best number n^* of classes, where $n^* \in [2, \ldots, N]$.
- the integration of this automatic class separation method in a segmentation algorithm thanks to a combination with the region growing technique.

For each attribute, λ^* is the criterion determining the best number n^* of classes. λ^* must maximize $\lambda_{(k)}$, $k \in [2, \ldots, N]$.

$$\lambda^* = \max(\lambda_{(k)}) \ ; \ \lambda_{(k)} = \frac{\sigma^2_{B_{(k)}}}{\sigma^2_{W_{(k)}}}$$

where $\lambda_{(k)}$ is the maximal criterion for exactly k classes.

$\sigma^2_{B_{(k)}}$ is the inter-classes variance defined by:

$$\sigma^2_{B_{(k)}} = \sum_{m=1}^{k-1} \sum_{n=m+1}^{k} [\omega_n \cdot \omega_m (\mu_m - \mu_n)^2]$$

$\sigma^2_{W_{(k)}}$ is the intraclass variance defined by:

$$\sigma^2_{W_{(k)}} = \sum_{m=1}^{k-1} \sum_{n=m+1}^{k} [\sum_{i \in m} (i-\mu_m)^2 \cdot p_{(i)} + \sum_{i \in n} (i-\mu_n)^2 \cdot p_{(i)}]$$

μ_m denote the mean of the level i of the class m, ω_m the class probability and $p_{(i)}$ the probability of the level i of the histogram.

$$\mu_m = \sum_{i \in m} \frac{i \cdot p_{(i)}}{\omega_m} \qquad \omega_m = \sum_{i \in m} p_{(i)} \qquad p_{(i)} = \frac{n_i}{Np}$$

The normalized histogram is considered to be a probability distribution: n_i is the number of samples for a given level and Np is the total number of samples. A class m is delimited by two values (the inferior and superior limits) corresponding to two levels in the histogram.

The automatic class separation method was applied to the two histograms shown in figure 1: in both cases the class division was tested for two and three classes. For the first histogram, the value λ^* corresponds to a division into two classes; the threshold is placed in the valley bottom between the two peaks. In the second histogram, the optimal λ^* corresponds to a division into three classes.

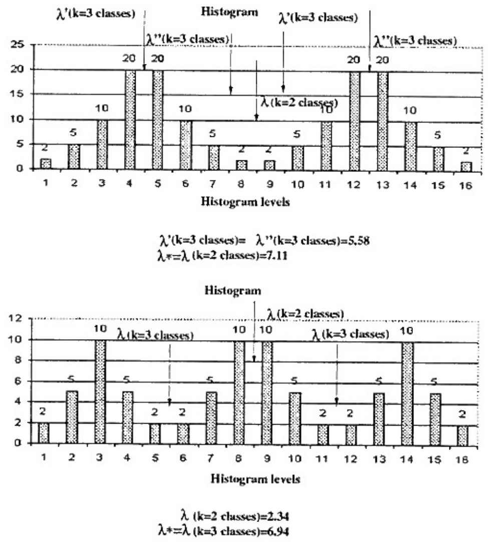

Figure 1: Threshold Localization

2.1 The 3D image segmentation

This segmentation algorithm can be applied to images of range, by the use of 3D attributes (height and normals). On our LAMA robot, the 3D image is provided by a stereo-vision algorithm [3]: height and normals are computed for each point in the 3D image. The height correspond to the distances from the 3-D points of the object, with respect to the plane which approximates the ground area from which this object is emerging. The normals (θ and ϕ) are computed in a spherical coordinate system [1]. Height and normals are coded in 256 levels.

Comparing with the method we developed previously [1], this one is more generic (we can add as many attributes as it is required) and less dependent on the parameter selection. The normal is not always the best attribute. Height is generally good enough for obtaining an acceptable 3D segmentation.

2.2 The color image segmentation

A color image is usually described by the distribution of the three color components R (red), G (green) and B (blue), moreover many other attributes can also be calculated from these components. Two goals are generally pursued: first the selection of uncorrelated color features [8, 14], and secondly the selection of attributes which are independent of intensity changes, especially in outdoor environments where the light conditions are not controlled [11]. However, some previous methods depend greatly on intensity features [5]. Several color representations have been tested: R.G.B., r.g.b. (normalized components of R.G.B.), Y.E.S. defined by the SMPTE (Society of Motion Pictures and Television Engineers), H.S.I. (Hue, Saturation and Intensity) and I_1, I_2, I_3, color features derived from the Karhunen-Loève (KL) transformation of RGB. The results of segmentation obtained by using each color space have been compared. Good results with only chrominance attributes depend on the type of images. Chromimance effects are reduced in images with low saturation. For this reason, the intensity component is kept in the segmentation step. Over-segmentation errors can occur due to the presence of strong illumination variations (i.e. shadows). However, over-segmentation is better than the loss of a border between classes. The over-segmentation errors will be easily detected and fixed during the identification step.

Finally, the best color segmentation was obtained by using the I_1, I_2, I_3 space, defined as [5, 14]: $I_1 = \frac{R+G+B}{3}$, $I_2 = (R-B)$, $I_3 = \frac{2G-R-B}{2}$. The components of this space are uncorrelated, so statistically it is the best way for detecting color variations. The number of no homogeneous regions (sub-segmentation problems) is very small (2%); a good tradeoff between fewer regions and the absence of sub-segmentation has been obtained, even in the case of complex images.

3 The identification method

The nature of the elements in the scene is estimated by comparing their attribute vectors (computed from the 3D features, color and texture features) with a database composed by different classes, issued from a learning step executed off line.

This database is a function of a certain type of environment and it is obtained by a supervised learning process. Here, we have selected 4 generic classes, which correspond to the 4 principal elements of our environment: grass, sky, tree and rock.

3.1 The region characterization

Image regions corresponding to areas of the environment close to the sensor can be characterized with the 3D and the intensity attributes. Image regions corresponding to areas far away from the sensor are characterized by using only the color and the texture attributes. Up to now, the color attributes have not been used for the closer regions, because our stereo system is composed of black and white cameras; another one composed of color cameras is under test and will be mounted on the robot as soon as possible.

The region scene, where the 3D information is valid, is delimited by the 3D segmentation. Figure 3(a) shows an example. The border is marked with a bold line. Intensity attributes to a region issued from the 3D image segmentation, are taken from the corresponding 2D region in the left image acquired from the stereo system.

The texture operators are based on the sum and difference histograms [15]. This texture measurement is an alternative to the usual co-occurrence matrices. The sum and difference histograms used conjointly are nearly as powerful as co-occurrence matrices for texture discrimination. Moreover, it requires less computation time and memory storage than the classical spatial gray level dependence method.

If m is the number of points belonging to a region, the normalized sum and difference histograms are defined:

$$H_s(i) = \frac{Card(i \in I_s(x,y))}{m} \qquad H_d(j) = \frac{Card(j \in I_d(x,y))}{m}$$
$$H_s(i) \in [0,1] \qquad H_d(j) \in [0,1]$$

Texture Feature	Equation
Mean	$\mu = \frac{1}{2}\sum_i i \cdot \hat{P}_{s(i)}$
Variance	$\frac{1}{2}(\sum_i (i-2\mu)^2 \cdot \hat{P}_{s(i)} + \sum_j j^2 \cdot \hat{P}_{d(j)})$
Energy	$\sum_i \hat{P}^2_{s(i)} \cdot \sum_j \hat{P}^2_{d(j)}$
Entropy	$-\sum_i \hat{P}_{s(i)} \cdot \log \hat{P}_{s(i)} - \sum_j \hat{P}_{d(j)} \cdot \log \hat{P}_{d(j)}$
Contrast	$\sum_j j^2 \cdot \hat{P}_{d(j)}$
Homogeneity	$\frac{1}{1+j^2}\sum_j \cdot \hat{P}_{d(j)}$

Table 1: Texture Features computed from the sum and difference histograms

The relative displacement $(\delta x, \delta y)$ must be chosen in such a manner that the computed texture attributes allow to discriminate the interesting classes. For our problem we have chosen: $\delta x = \delta y = 1$.

Histograms provide a probabilistic characterization of the spatial organization in a region, based on neighborhood analysis. Statistical information can be extracted from these histograms. Six texture features, defined in Table 1, are computed from the sum and difference histograms.

When the color information is available, in addition to these texture features the statistical means of I_2 and I_3 are used to characterize the color in a region. In order to reduce the dependency of intensity changes in the identification step, the intensity component has been dropped out.

3.2 The region classification

Two classification techniques have been compared: the Bayesian classification and an hierarchical one, based on the concept of the average mutual information, which generates an efficient partitioning tree.

3.2.1 The Bayesian classifier

This classical probabilistic approach is based on the Bayesian rule defined as [2]:

$$P(C_i \mid X) = \frac{P(X \mid C_i)P(C_i)}{\sum_{i=1}^{n} P(X \mid C_i)P(C_i)}$$

As an equal *a priori* probability is assumed, the computation of *a posteriori* probability $P(C_i \mid X)$ can be simplified and its value depends only on $P(X \mid C_i)$. The value of $P(X \mid C_i)$ is estimated by using the k-nearest neighbor method. The observation X will be assigned to the class C_i so that the sample which is the $k-th$ nearest neighbor to X, is closest to X than to any other training class. The Bayesian classification does not perform a feature selection, the whole vector of previously defined attributes has to be computed for each sample; nevertheless, in order to reduce the computational running time of both classification and characterization steps, a data analysis is performed off line to decrease the dimension of the attribute space. This data analysis is composed of two steps, Analysis of capacity of discrimination and analysis of correlation the first one is done by using the Fisher's criterion and the second one is based on PCA.

The acknowledge of the discrimination power for each feature (computed from the Fisher criterion), the variance of the samples over the axis and the correlation among them (computed from the PCA) allows us to select theses ones having the greatest discrimination power and uncorrelated. We had decided to use the pertinent subset of original features instead of their linear combination, given that these last ones force the computation of several original features per factorial axis. Additionally to employ linear combination of original features does not have interest, since the k-nearest neighbor method is used to estimate $P(X \mid C_i)$.

The features chosen were: Entropy, Homogeneity, I2, I3, and Mean. These features have the greatest discrimination capacity. Entropy, Homogeneity and I_2 are correlated among them, however these ones had been conserved because it have maximal dispersion and appropriate discrimination capacity.

3.2.2 The Hierarchical classifier

The second classification method is based on an algorithm for the partitioning of the feature space (see [13] for more details). This algorithm determines the number of hyper-planes (parallels to feature axes), the associated features and their order so as to divide the feature space. This algorithm has inherent feature selection capability. The algorithm gives rise to a locally optimal decision tree by maximizing the amount of average mutual information gained at each partitioning step. The average mutual information obtained about a set of classes C_k from the observation of an event X_k, at a node k in a tree T is defined as:

$$I_k(C_k, X_k) = \sum_{C_k} \sum_{X_k} p(C_{ki}, X_{kj}) \cdot \log_2 [\frac{p(C_{ki} \mid X_{kj})}{p(C_{ki})}]$$

Event X_k represents the measurement value of a feature selected at node k and has two possible outcomes; measurement values are greater or smaller than a threshold associated with that feature at that node.

In addition to the partitioning of the feature space, identification security areas have been associated with each partition. The statistical means and standard deviations of each feature conserved at each terminal node are computed using only the training subset, which defines the partition concerned. For the attribute j, these areas are determined by:

$$L_{inf}(j) = \hat{m}_k(j) - \frac{C_c \hat{\sigma}_k(j)}{\sqrt{n_k}} \quad ; \quad j \subset \Omega_k$$

$$L_{sup}(j) = \hat{m}_k(j) + \frac{C_c \hat{\sigma}_k(j)}{\sqrt{n_k}} \quad ; \quad j \subset \Omega_k$$

where $L_{inf}(j)$ and $L_{sup}(j)$ are the inferior and superior limits of the area, $\hat{\sigma}_k(j)$ is the standard deviation, $\hat{m}_k(j)$ is the statistical means, Ω_k is the feature subset in the terminal node k, C_c is the confidence coefficients and n_k is the number of samples in the partition k. Thus, the local distribution of the training set into each partition is also taken into account. Areas of the feature space that are outside the confidence borders can be interpreted as regions of non-classification.

3.3 Fusion of regions and coherence of the model

At this step of the process, each region given by the segmentation step has been associated with a class. Some drawbacks of the segmentation can be corrected now: the segmentation provides large regions, that do not always correspond to the real borders between classes; the area corresponding to the same class is often over-segmented. Consequently, a fusion step is necessary in order to merge adjacent regions belonging to the same class.

By the use of some contextual characteristics of the environment the model consistency can be tested; possible errors in the identification process could be detected and corrected by using simple contextual rules (example: "grass cannot be surrounded by sky regions").

4 Robot visual navigation based on landmarks

This approach has been applied to several tasks in the context of mobile robotics. First, a partial implementation was used to select an appropriate landmark in function of its nature and shape (2-D) [4]. Then, the nature and the 3D representation of the objects present in the scene allow the construction of a complete and global model of the environment, which will be updated during the robot motion. Some objects in this map can be used to localize the robot. Some of the developments presented here are used for this purpose in the case of planetary environment [9].

We are currently investigating visual robot navigation based on landmark identification. To navigate during a long robot motion, a sequence of different landmarks (or targets) is used as sub-goal that the robot must successively reach. The landmark change is automatic. It is based on the nature of the landmark the distance between the robot and the landmark which represents the current sub-goal. When the robot attains the current landmark (or, more precisely, when the current landmark is close to the limit of the camera field of view), another one is dynamically selected in order to control the next motion.

The selection of the target (as a sub-goal) is linked to the landmark definition. A landmark should be a single discriminant structure or a discriminant configuration of features. The landmark should have some properties that distinguish them from other objects: **Discrimination** a landmark should be easy to differentiate from other surrounding objects. **Accuracy** a landmark must be useful to reduce the uncertainty of the robot position. In our case landmarks are selected between unstructured objects, mainly rocks on the ground. An object is selected as landmark if: It is not occluded by another object or by the image contour and its topmost point is the most accurate.

5 Experimental results

The construction of the model of the scene (segmentation, region characterization and identification) is done in about 2.5 seconds on SPARC 20. The results of the identification by using 2-D features were as follows: the database was generated from 60 images, the segmentation on these images provided 418 regions, these regions were employed as the training set. The identification was performed over 30 images, none of which were included in the training set. The hierarchical classifier gives 92 % of successful regions classification and the Bayesian classification 89 %.

The hierarchical classifier gives somewhat better results than the Bayesian classification. However, as the number of terminals is larger than the number of actual classes, the classification time and memory space requirements are larger than for the Bayesian classification. By using a reduced space of attributes, the Bayesian classification gives a better compromise between computational running time and correct identification than the hierarchical classifier.

Figure 2 show images corresponding to natural environment. The segmented and classified image is presented in 2 (e), 2 (f), 2 (g), 2 (h), Figures 2 (a), 2 (b), 2 (d) show objects (trees, rocks ...) which are far from the sensor. In this case, a laser ranger finder or a stereoscopic system cannot give a valid 3D information and so a 3D model cannot be directly built, unlike video camera, which can give valid texture and color information to build a 2-D model. This model can be used in order to give to the robot a goal (direction) corresponding to a landmark of a requested class and 2-D shape.

In order to show the approach's capability to work in different environmental conditions, the images were taken in different light conditions and the sensor was placed at different distances from the elements in the scenes.

Figures 2 (d) shows a new class (road of soil). This one was included in the data base. The objective is to present the possibility to increase the number of classes.

This method had been also tested over more than 100 scenes closed to the sensor. The attributes employed to characterize the elements were the texture features presented in § 3.1 and the statistical mean and the standard deviation of the height from the 3D points of the element. The figure 3 shows some results. Figures 3 (a), 3 (e) and 3 (i) show the video image. The segmentation is indicated by using lines on the image. Figures 3 (b), 3 (f) and 3 (j) present the height image. White pixels correspond to non correlated points (too distant 3D points, regions with low texture or occlusions). Figures 3 (c), 3 (g) and 3 (k) show the classified and merged images. Non correlated points are detected, these ones labeled in black. Figures 3 (d), 3 (h) and 3 (l) show the 3D images.

5.1 Experiments of robot visual navigation based on landmarks

We illustrate this task with a experiment carried out with the mobile robot LAMA. The figure 4 (a) shows the video image, figure (b) presents the 3-D image and the figure (c) shows the 3-D image segmentation, classification and boundary box including the selected landmark. The selection was done taking into account 3-D shape and nature.

The second line of figure 4 represent the tracking of a landmark through an image sequence. The landmark is marked on the picture with a little boundary box. The tracking process is performed based on a comparison between a model of the landmark and the image. In [4] is described in detail the tracking technique used. When the landmark position is close to the image edge, then it is necessary to select another landmark. So the figure 4 III presents the new landmark selection based on image segmentation and classification. The next sequence of tracking is shows on the line IV of figure 4 and the next landmark commutation is presents on line V. Finally on the line VI the robot continue navigation task.

6 Discussion and future work

Given that the identification step is based on supervised learning process, its good performance depends on the utilization of a database representative enough of the environment. However if the robot navigates just in a single type of environment (i.e terrestrial or planetary terrains), this limit is not a big deal because a specific environment can be represented by a reduced number of classes. If different types of environment are considered, it can be possible to solve the problem by a hierarchical approach: a first step could identify the environment type (i.e. whether the image shows a forest, a desert or an urban zone) and the second one the elements in the scene. The first step has been considered in recent papers [10]: these approaches are not able to identify the elements in the scene but the whole image like an entity. After having obtained the scene type, our identification method could be used

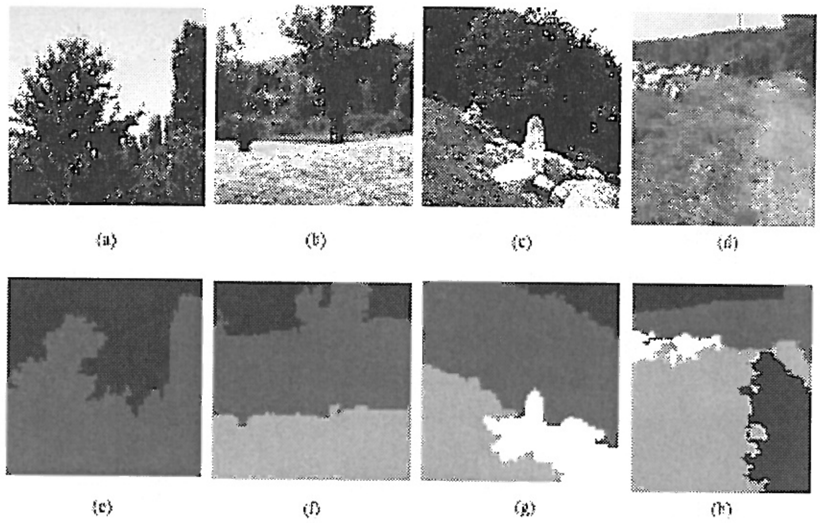

Figure 2: Identification from 2D attributes on different scenes

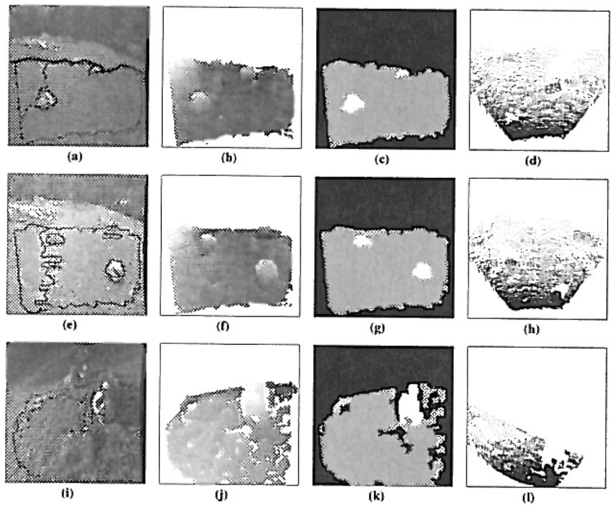

Figure 3: Identification using 2D and 3D attributes

to realize the second step. In this case a database organized in function of the types of environment is suitable. It allows to reduce the number of classes, then decreasing the complexity of the problem (i.e. in lunar environment the tree class is not looked for, but the depression class "holes" is). Additionally it is easier to profit from contextual information when the environment type is known. We propose this strategy as a first perspective.

Acknowledges: This work was funded by CONACyT (México) and by the ITESM Campus Ciudad de México.

References

[1] S. Betg-Brezetz, R. Chatila, and M. Devy. Natural Scene Understanding for Mobile Robot Navigation. In *Proceedings of the IEEE ICRA'94*, California, USA, Mai 1994.

[2] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis.* Wiley & sons, 1973.

[3] H. Haddad, M. Khatib, S. Lacroix, and R. Chatila. Reactive navigation in outdoor environments using potential fields. In *International Conference on Robotics and Automation ICRA'98*, pages 1332–1237, may 1998.

[4] R. Murrieta-Cid, M. Briot, and N. Vandapel. Landmark identification and tracking in natural environment. In *International Conference on Intelligent Robots and Systems IROS'98*, Victoria, Canada, October 1998.

[5] Y. Ohta. *Knowledge-Based Interpretation of Outdoor Natural Color Scenes.* Morgan Kaufman Publishers, Inc., Palo Alto, CA, 1985.

[6] C.F.Olson and L.H.Matthies. Maximum Likelihood Rover Localization by Matching Range Maps. In *International Conference on Robotics and Automation ICRA'98*, pages 272-277, may 1998.

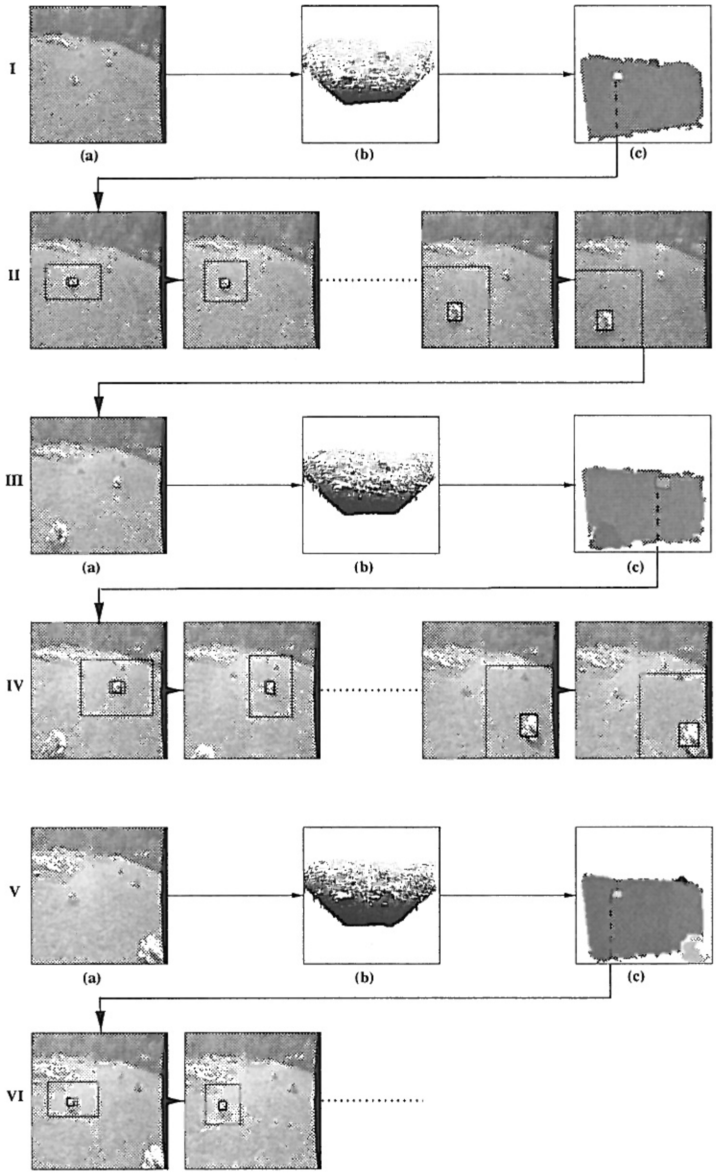

Figure 4: Visual robot navigation based on landmarks

[7] N. Otsu. A Threshold Selection Method from Gray-Level Histrograms. *I.E.E.E. Transaction on Systems, Man and Cybernetics*, 9(1):62–66, january 1979.

[8] N.R. Pal and S.K. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(9):1277–1294, 1993.

[9] C. Parra, R. Murrieta-Cid, M. Devy, and M. Briot. 3-D modelling and robot localization from visual and range data in natural scenes. In *International Conference on Vision Systems (ICVS)*, 1999.

[10] Y. Rubner, C. Tomasi, and J. Guibas. A metric for distributions with applications to image databases. In *IEEE International Conference on Computer Vision*, Bombay, India, 1998.

[11] E. Saber, A.M. Tekalp, R. Eschbach, and K. Knox. Automatic Image Annotation Using Adaptative Color Classification. *Graphical Models and Image Processing*, 58(2):115–126, march 1996.

[12] S. Rasoul Safavian and David Landgrebe. A survey of decision tree classifier methodology. *I.E.E.E. Transaction on Systems, Man and Cybernetic*, 1991.

[13] I.K. Sethi and G.P.R. Sarvarayudu. Hierarchical classifier desing using manual information. *I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence*, 1982.

[14] T.S.C Tan and J. Kittler. Colour texture analysis using colour histogram. *I.E.E Proc.-Vis.Image Signal Process.*, 141(6):403–412, december 1994.

[15] M. Unser. Sum and difference histograms for texture classification. *I.E.E.E. Transactions on Pattern Analysis and Machine Intelligence*, 1986.