Autonomous Robots 13, 143–168, 2002

Visual Navigation in Natural Environments: From Range and Color Data to a Landmark-Based Model

RAFAEL MURRIETA-CID*
ITESM Campus Ciudad de México, Calle del puente 222, Tlalpan, México DF
rmurriet@campus.ccm.itesm.mx

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

MICHEL DEVY
Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS-CNRS), 7, Avenue du Colonel Roche, 31077 Toulouse Cedex 4, France
michel@laas.fr

Abstract. This paper concerns the exploration of a natural environment by a mobile robot equipped with both a video color camera and a stereo-vision system. We focus on the interest of such a multi-sensory system to deal with the navigation of a robot in an a priori unknown environment, including (1) the incremental construction of a landmark-based model, and the use of these landmarks for (2) the 3-D localization of the mobile robot and for (3) a sensor-based navigation mode.

For robot localization, a slow process and a fast one are simultaneously executed during the robot motions. In the modeling process (currently 0.1 Hz), the global landmark-based model is incrementally built and the robot situation can be estimated from discriminant landmarks selected amongst the detected objects in the range data. In the tracking process (currently 4 Hz), selected landmarks are tracked in the visual data; the tracking results are used to simplify the matching between landmarks in the modeling process.

Finally, a sensor-based visual navigation mode, based on the same landmark selection and tracking, is also presented; in order to navigate during a long robot motion, different landmarks (targets) can be selected as a sequence of sub-goals that the robot must successively reach.

Keywords: vision, robotics, outdoor model building, target tracking, multi-sensory fusion, visual navigation

1. Introduction

This paper deals with perception functions required on an autonomous robot which must explore a natural environment without any a priori knowledge. From a sequence of range and video images acquired during the motion, the robot must incrementally build a model, correct its estimate situation or execute some visual-based motion.

This work is related to the context of a Mars rover. The robot must at first build some representations of the environment based on sensory data before exploiting them in order to perform some tasks such as picking up rock samples. A fundamental task in this context is simultaneous localization and modeling

*This research was funded by CONACyT, México.
†This research was funded by the PCP program (Colombia—COLCIENCIAS and France) and by the ECOS Nord project number C00M01.

(SLAM). This task will be described below in more details. In this paper we do not take profit of any external robot localization system provided by DGPS (Dumaine et al., 2001) or by the cooperation between aerial and terrestrial robots.

The proposed approach is suitable for environments in which (1) the terrain is mostly flat, but can be made by several surfaces with different orientations (i.e. different areas with a rather horizontal ground, and slopes to connect these areas) and (2) objects (bulges or depressions) can be distinguished from the ground. Several experimentations on data acquired on such environments have been done. Our approach has been tested partially or totally in the EDEN site of the LAAS-CNRS (Murrieta-Cid, 1998; Murrieta-Cid et al., 1998a, 1998b; Murrieta-Cid et al., 2001), the GEROMS site of the CNES (Parra et al., 1999), or even over data acquired in the Antarctica (Vandapel et al., 1999). These sites have the characteristics for which this approach is suitable. The EDEN site is a prairie, and the GEROMS site is a simulation of a Mars terrain.

For this topics, the classical lines of research in perception for mobile robots are based on 3-D information, obtained by a laser ranger finder or a stereoscopic system (Krotkov et al., 1989; Kweon and Kanade, 1991; Betg-Brezetz et al., 1996). Our previous method (Betgé-Brezetz et al., 1995; Betg-Brezetz et al., 1996) dedicated to the exploration of such an environment, aimed to build an object-based model, considering only range data. An intensive evaluation of this method has shown that the main difficulty comes from the matching of objects perceived in multiple views acquired along the robot paths. From numerical features extracted from the model of the matched objects, the robot localization can be updated (correction of the estimated robot situation provided by internal sensors: odometry, compass, ...) and the local models extracted from the different views can be consistently fused in a global one. The global model was only a stochastic map in which the robot situation, the object features and the associated variance-covariance matrix were represented in a same reference frame (typically, the first robot situation during the exploration task). Robot localization, fusion of matched objects and introduction of new perceived objects are executed each time a local model is built from a new acquired image (Smith et al., 1990). If any mistake occurs in the object matchings, numerical errors were introduced in the global model and the robot situation could be lost.

The main reason of these failures, is that a 3D geometric representation is not enough to get a complete description of the environment. Other information such as the nature of the objects detected in the scene need to be taken into consideration. In this paper, we present an improved modeling method, based on a multi-sensory cooperation using both range and visual data in order to make the matching step more reliable. Our main contributions concern two main topics:

- The model building by using both 2D and 3D knowledges. In our approach we add to the geometric representations (intrinsic shape attributes, positions, ...), other attributes based on texture and/or color informations. From all these attributes, using an a priori learning step, a classifier can provide a semantic labelling of the detected objects or regions.
- The dynamic aspects of the visual processes both, for the incremental environment modeling and for the visual navigation towards landmarks selected as targets. The matching between landmarks detected in different perceptions is required for the global model construction and is facilitated by using the result of a tracking process. The semantic labelling is exploited to select the landmarks and to check the tracking consistency.

Our local modeling approach includes an interpretation procedure suitable for outdoor natural scenes. For every acquired image, it consists on several steps. Firstly, a segmentation algorithm provides a description of the scene as a set of regions. Our segmentation method is able to get a synthetic scene description even for complex environments and can be applied to both 2D and 3D information either in sequential or parallel manner. The segmentation technique will be presented in detail in Section 3. Then, regions obtained by the segmentation step, are characterized by using several attributes, and finally their nature is identified by probabilistic methods.

Our tracking method, has been presented in Huttenlocher et al. (1993b), Dubuisson and Jain (1997), Murrieta-Cid (1997), and Rucklidge (1997). The tracking is done using a comparison between an image and a model. The Hausdorff distance is used to measure the resemblance of the image with the model. The association between the motion estimation in the image and the scene interpretation has been used to select a landmark having the required nature and shape as a target for the tracking.

Finally, the global model is built from the successive fusion of the local models, using the tracking results. With respect to our previous work, the same localization and fusion procedures are used, but now, our global model has several levels, like in Bulata and Devy (1996): A topological level gives the relationships between the different ground surfaces (connectivity graph). The model of each terrain area is a stochastic map which gives information only for the objects detected on this area. This map gives the position of these objects with respect to a local frame linked to the area.

Let us describe the organization of this paper. In Section 2, some related works and an overview of our system are presented. In the Section 3, a general function which performs the construction of a local model for the perceived scene, will be detailed. This function is implemented as a slow loop (from 0.1 to 0.2 Hz according to the scene complexity and the available computer) from the acquisition of range and visual data to the global model updating. The landmark selection process is presented in Section 4, this one is executed only at the beginning or after the detection of an inconsistency by the modeling process. The tracking process is described in Section 5. The tracking process is implemented as a fast loop (from 2 to 4 Hz), which requires only the acquisition of an intensity image.

The global model building and robot localization are described in Section 6. Finally, experimental results of SLAM and visual navigation obtained from a partial integration of these processes will be presented and analyzed in the Section 7. Our navigation method has been evaluated either on a lunar-like environment or on terrestrial natural areas. The experimental tested used to carry out these experiments is the robot LAMA (Fig. 1). It is equipped with a stereo-vision system composed by two black and white cameras. Additionally to this stereo-vision system a single color camera has been used to model scenes far away from the robot.

Figure 1. The robot LAMA.

2. The General Approach

2.1. Related Work

The construction of a complete model of an outdoor natural environment, suitable for the navigation requirements of a mobile robot, is a quite difficult task. The complexity resides on several factors such as (1) the great variety of scenes that a robot could find in outdoor environments, (2) the fact that the scenes are not structured, then difficult to represent with simple geometric primitives, and (3) the variation of the current conditions in the analyzed scenes, for instance, illumination and sensor motion. Moreover, another strong constraint is the need of fast algorithm execution so that the robot can react appropriately in the real world.

Several types of partial models have been proposed to represent natural environments. Some of them are numerical dense models (Krotkov et al., 1989; Hebert et al., 1989), other are probabilistic and based on grids (Lacroix et al., 1994). There exist also topological models, for instance, Dedeoglu et al. (1999), for indoor environments. In general, it is possible to divide the types of models in three categories (Chatila and Laumond, 1985):

1. geometric models: this model contains the description of the geometry of the ground surface or some of its parts.
2. topological models: this model represents the topological relationships among the areas in the environment. These areas have specific characteristics and are called “places”.
3. semantic models: this is the most abstract representation, because it gives to every entity or object found in the scene, a label corresponding to a class (tree, rock, grass. . .). The classification is based on a priori knowledge learnt off line and given to the system. This knowledge could consist in (1) a list of possible classes that the robot could identify in the environment, (2) attributes learnt for some samples of each class, (3) the kind of environment to be analyzed, . . .

A very large majority of methods proposed to model a natural environment, have been focused on geometric

models. Nevertheless there are some works which build a topological model of natural environments based on:

- Grid representations. Grids, with sometimes different hierarchical levels, are often selected for their simplicity (Metea and Tsai, 1987).
- Graph representations. Some geometrical characteristics of a geometrical model allow to define a graph of objects (Kweon and Kanade, 1991); these characteristics can be also used to split the environment in homogeneous areas (Asada, 1988), using sensor constraints (visibility of landmarks) or locomotion constraints (nature of the terrain).

Some recent works propose landmark-based navigation methods. In McKerrow and Ratner (2001), the landmarks are detected using only an ultrasonic sensor, but the environment is very simple (typically a golf course) and detected landmarks are only poles. On the opposite side, the work presented in Rosenblum and Gothard (2000) is based on very expensive FLIR cameras; from the images, attributes are extracted, and image regions are labelled Rock, Grass, Bush, Tree, ... a reactive navigation mode is based on these labelled images. Our approach is close to this previous one, but (1) we use only color cameras, (2) the robot executes either a trajectory-based or a landmark-based navigation process and (3) visual tracking is integrated so that landmarks are dynamically tracked during the robot motions.

2.2. *The Navigation Modes*

We have described on Fig. 2 the relationships between the main representations built by our system, and the different processes which provide or update these representations.

We propose here two navigation modes which can take profit of the same landmark-based model: Trajectory-based navigation or sensor-based navigation.

The sensor-based navigation mode needs only a topological model of the environment. It is a graph, in which a node (a place) is defined both by the influence area of a set of landmarks and by a rather flat ground surface. Two landmarks are in the same area if the robot can execute a trajectory between them having always landmarks of the same set in the stereo-vision field of view (max range = 8 m). Two nodes are connected by an edge if their ground surfaces have significantly different slopes, or if sensor-based motions can be executed to reach one place from the other.

The boundary between two ground surfaces is included in the environment model as a border line (Devy and Parra, 1998). These lines can be interpreted as "doors" towards other places. These entrances towards other places are characterized by their slope. A tilted surface becomes an entrance if the robot can navigate through it. An arc between two nodes corresponds either with a border line, or with a 2D landmark that the robot must reach in a sensor-based mode. In Fig. 3 are shown a scheme representing the kind of environment where this approach is suitable and its representation with a graph.

The trajectory-based navigation mode has an input provided by a geometrical planner. It is selected inside a given landmark(s) influence area. The landmarks in this type of navigation mode must be perceived by 3D sensors, because they are used to localize the robot (see Figs. 23 and 25). The sensor-based navigation mode can be simpler, because it exploits the landmarks as sub-goals where the robot has to go. The landmark position in a 2D image is used to give the robot a motion direction (see Fig. 24).

Actually, both of the navigation modes can be switched depending on (1) the environment condition, (2) whether there is 3D or 2D information and (3) the availability of a path planner. In this paper we present overall examples when 3D information is available. 3D information allows the trajectory-based navigation mode based on robot localization from the 3D landmark positions.

2.3. *Overview of Our System*

In the model proposed the main entities are: (1) ground areas defined by the surfaces (rather smooth, sloped or not) and their nature (grass, sand, earth, . . .). (2) objects defined by their shape (if 3D data is available) or any spatial area represented by a region (if only 2D data is available) and their nature (rocks, tree, bushes, . . .). (3) boundaries between these entities which can be rather approximative.

The landmark-based model proposed is built by using several processes: A segmentation algorithm provides a synthetic description of the scene. Entities issued from the segmentation stage (ground areas or objects) are then characterized and afterwards identified in order to obtain their nature (e.g., soil, rocks, trees . . .).

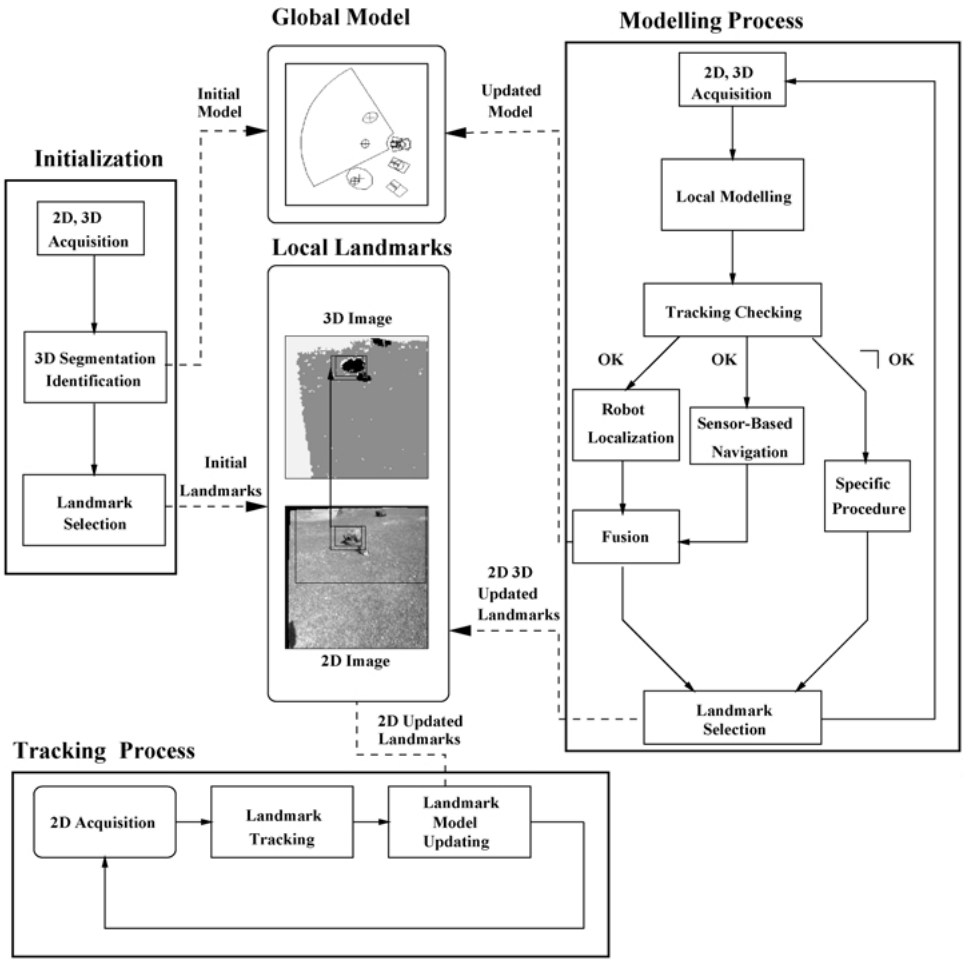


Figure 2. The general approach.

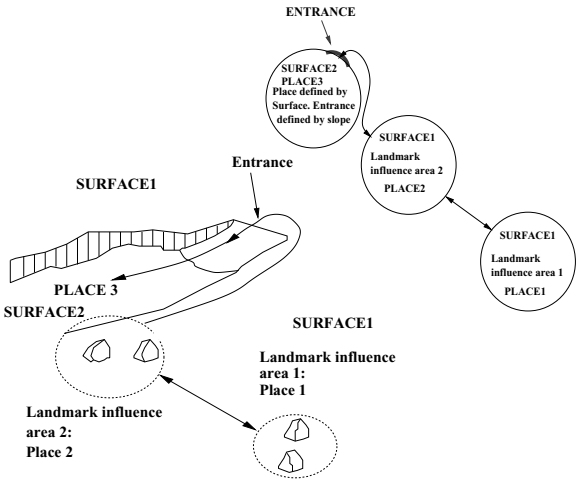


Figure 3. Topological model.

The nature of the elements in the scene is obtained by comparing an attribute vector (computed from the shape, color and texture informations extracted from sensory data associated with this element) with a database. This database is function of the type of the environment. In lunar-like environment we have chosen 3 classes (ground, rocks and sky). Terrestrial natural areas are richer regarding the type of classes that can be found, for this reason we have chosen 4 classes, which correspond to the principal elements in our scenes: Soil, rock, trees and sky. In both cases new classes inclusion as rocky soil and ground depressions (holes) are currently going on. The attributes used to characterize these environments must be different because they have different discriminative power according to the environment. For instance, in lunar like environment color in not useful given that the whole environment

has almost the same colors, but texture and 3D information are. In terrestrial natural areas the color is important because it changes drastically according to the classes the object belongs to. Information to use depend also on the sensor capabilities. Regions corresponding to areas far away from the sensor cannot be analyzed by using 3D information because this information is not available or too noisy.

These phases allow us to obtain a local model of the scene. From this model, discriminant features can be extracted and pertinent objects for the localization tasks are selected as landmarks, according to some criteria which depend on higher decisional levels, one of these landmark is chosen as a tracked target. This same landmark could also be used as a goal for visual navigation. The tracking process exploits only a 2D image sequence in order to track the selected target while the robot is going forward. When it is required, the modeling process is executed. A local model of the perceived scene is built. The robot localization is performed from matchings between landmarks extracted in this local model, and those previously merged in the global model. If the robot situation can be updated, the models of these matched landmarks are fused and new ones are added to the global model.

The matching problem of landmark's representation between different perceptions is solved by using the result of the tracking process. Moreover, some verifications between informations extracted from the 2D and 3D images allow to check the coherence of the whole modeling results; especially, a tracking checker is based on the semantical labels added to the extracted objects by the identification function.

3. Local Scene Modeling

Firstly, the local model of the perceived scene is required in order to deal with the incremental construction of a global model (Parra et al., 1999), or to select a goal for the next motion.

The construction of this local model is performed from the acquisition of a 3D image by the range sensor, and of a 2D image from the video sensor. Several processes are executed on sensory data.

The order and goal of each of these steps applied on the 2D and/or 3D images (Murrieta-Cid, 1998) are described below:

1. Image segmentation for region extraction: This segmentation is based on clustering and unsupervised classification. The image is segmented to obtain the main regions of the scene. This first step can be performed by the use of the color attribute on the 2D image or by the use of geometrical attributes on the 3D image.
2. Region characterization: Each region of the scene is characterized by using several attributes computed from the color, texture or geometrical informations (Unser, 1986; Tan and Kittler, 1994).
3. Region identification: It is based on knowledge-based classification after a supervised learning, the nature (class) of the elements (regions) in the scene is obtained by comparing a vector of features with a database composed of different classes, issued from a learning process. The database is a function of the environment type.
4. Edge-based segmentation used to split connected 3D objects belonging to the same class.

We want also to associate intensity attributes to an object extracted from the 3D image, this object creates a 2D region in the intensity image acquired at the same time than the 3D one. Depending on their properties the attributes are used to segment or characterize the image or even for both tasks. For instance, 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 on scenes far away from the sensor.

On our LAMA robot, the 3D image is provided by a stereo-vision algorithm (Haddad et al., 1998); for the 2D image, two different sensor configurations have been considered:

- Either we are only interested in the texture information, and the stereo images have a sufficient resolution. The left stereo image provides the 2D image on which the texture information will be computed; the indexes between the 3D points and the 2D points are the same, so that the object region extracted from the 3D image is directly mapped on the 2D image.
- Or we want to take advantage of a high-resolution camera, of a color camera, or of an active camera (controlled lens). In such a case, the 2D image is provided by a specific camera, and a calibration procedure must be executed off line, in order to estimate the relative position between the 2D and the 3D sensors. The 2D region created by an object extracted

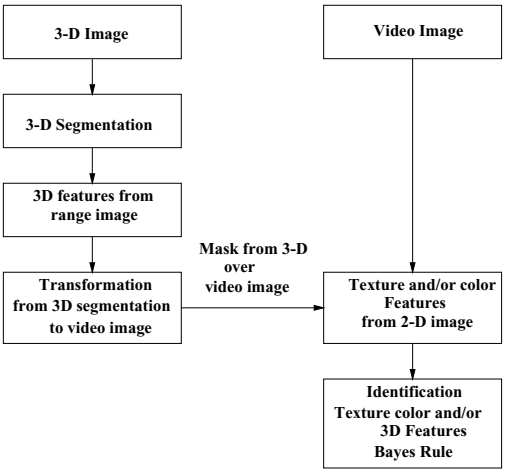


Figure 4. The local model construction.

from the 3D image, is provided by the projection on the 2D image of the 3D border line of the object.

Figure 4 shows the scene modeling process on the case of the usage of 3D information for segmentation and 2D/3D information for identification.

3.1. Scene Segmentation

The segmentation method here proposed is more robust and general than the methods that we have previously used (Betg-Brezetz et al., 1994; Murrieta-Cid et al., 1998a). It is able to get a synthetic scene description even for complex environments and can be applied to both 2D and 3D information. Scenes corresponding to areas far away from the sensor will be segmented by using only visual attributes given that 3D information is not available or too noisy. Unlike a video camera can give valid visual information (texture and color) 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.

This segmentation algorithm is a combination of two techniques: The characteristic feature, thresholding or clustering, and region growing (Murrieta-Cid et al., 2001). 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 (Murrieta-Cid et al., 1998a). 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 (Pal and Pal, 1993). However, these techniques require considerably tedious and sometimes unstable calculations. We have adapted the method suggested by Otsu (1979), 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 $\lambda_{(k)}$, $k \in [2, \ldots, N]$.

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

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:

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

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.

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

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 Fig. 5. 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.

3.1.1. The 3D 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 (Haddad et al., 1998): Height and normals are computed for each point in the 3D image. The normals (θ and ϕ) are computed in a spherical coordinate system (Betg-Brezetz et al., 1994), and are coded in 256 levels.

Once the ground regions have been extracted in the image, it remains the obstacle regions which could require a specific segmentation in order to isolate each obstacle. We make the assumption that an obstacle is a connected portion of matter emerging from the ground.

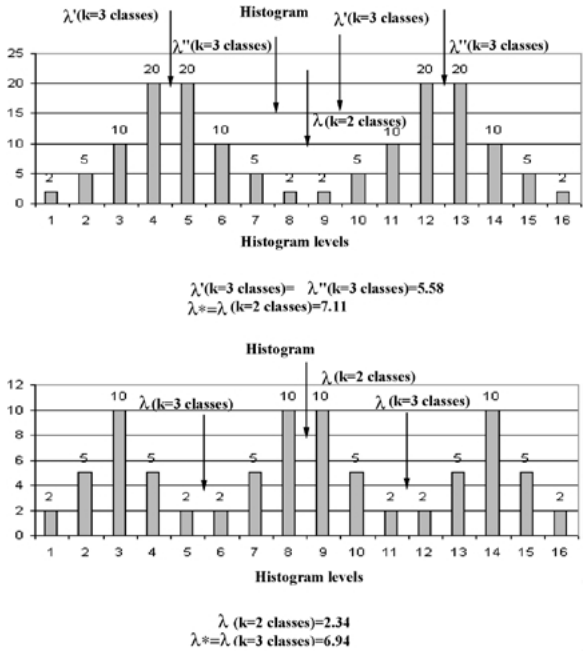


Figure 5. Localization of threshold.

Different obstacles are separated by empty space which could be identified as depth discontinuities in the 3D image. These discontinuities are detected in a depth image, in which for each 3D point of the 3D image, the corresponding pixel value encodes the depth with respect to the sensor. Thus a classical derivative filter can be applied to obtain maxima of gradient corresponding to the depth discontinuities. Classical problems of edge closing are solved with a specific filter described in Betg-Brezetz et al. (1994). Figure 6 shows a lunar-like environment, Fig. 7 shows the 3D segmentation. Figures 8 and 9 show other example. In this example a ground depression in the scene has been successfully segmented. White pixels in segmented images correspond to non correlated points (too distant 3D points, regions with low texture, shadows or occlusions).

Figure 6. Original image.

Figure 7. 3D segmentation.

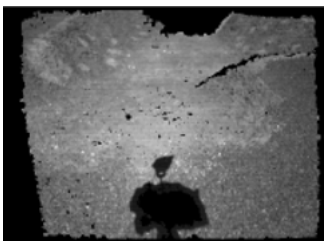


Figure 8. Original image.

Figure 9. 3D segmentation.

3.1.2. The Color Segmentation. Image regions corresponding to areas of the environment close to the sensors (in our robot, up to 8 meters) can be analyzed by using 3D and luminance attributes. Regions corresponding to areas far away from the sensor (beyond 8 meters) will be analyzed by using only luminosity attributes (the color and the texture) given that 3D information is not available or too noisy.

In terrestrial natural areas far away from the sensor color is chosen to segment the scene.

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: Firstly, the selection of uncorrelated color features (Pal and Pal, 1993; Tan and Kittler, 1994), and secondly the selection of attributes which are independent of intensity changes, especially in outdoor environments where the light conditions are not controlled (Saber et al., 1996; Ohta, 1985). 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 Ohta (1985) and Tan and Kittler (1994): $I_1 = \frac{R+G+B}{3}$, $I_2 = (R-B)$, $I_3 = \frac{2G-R-B}{2}$. 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 for complex images.

3.2. Object Characterization

Each object of the scene is characterized by an attribute vector: The object attributes correspond either to 3D features extracted from the 3D image and/or to its texture and its color extracted from the 2D image. The 3D features correspond to the statistical mean and the standard deviation of 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.

We want also to associate intensity attributes to an object extracted from the 3D image. This object creates a 2D region in the intensity image acquired at the same time than the 3D one.

The texture operators are based on the sum and difference histograms, this type of texture measure is an alternative to the usual co-occurrence matrices used for texture analysis. The sum and difference histograms used conjointly are nearly as powerful as co-occurrence matrices for texture discrimination. This texture analysis method requires less computation time and less

memory requirements than the conventional spatial grey level dependence method.

For a given region of a video image $I(x, y) \in [0, 255]$, the sum and difference histograms are defined as Unser (1986):

$$h_s(i) = \text{Card}(i = I(x, y) + I(x + \delta x, y + \delta y)) \quad i \in [0, 510]$$

$$h_d(j) = \text{Card}(j = |I(x, y) - I(x + \delta x, y + \delta y)|) \quad j \in [0, 255]$$

The relative displacement $(\delta x, \delta y)$ may be equivalently characterized by a distance in radial units and an angle θ with respect to the image line orientation: This displacement must be chosen so that the computed texture attributes allow to discriminate the interesting classes. For our problem, we have chosen: $\delta x = \delta y = 1$. Sum and difference images can be built so that, for all pixels $I(x, y)$ of the input image, we have:

$$I_s(x, y) = I(x, y) + I(x + \delta x, y + \delta y)$$
$$I_d(x, y) = |I(x, y) - I(x + \delta x, y + \delta y)|$$

Furthermore, normalized sum and difference histograms can be computed for selected regions of the image, so that:

$$H_s(i) = \frac{\text{Card}(i = I_s(x, y))}{m} \quad H_s(i) \in [0, 1]$$

$$H_d(j) = \frac{\text{Card}(j = I_d(x, y))}{m} \quad H_d(j) \in [0, 1]$$

where m is the number of points belonging to the considered region.

These normalized histograms can be interpreted as a probability. $\hat{P}_{s(i)} = H_s(i)$ is the estimated probability that the sum of the pixels $I(x, y)$ and $I(x + \delta x, y + \delta y)$ will have the value i. And $\hat{P}_{d(j)} = H_d(j)$ is the estimated probability that the absolute difference of the pixels $I(x, y)$ and $I(x + \delta x, y + \delta y)$ will have value j.

In this way we obtain a probabilistic characterization of the spatial organization of the image, based on neighborhood analysis. Statistical information can be extracted from these histograms. We have used 6 texture features computed from the sum and difference histograms, these features are defined in Table 1.

The histograms change gradually in function of the view point, the distance from the sensor to the scene and the occlusions (Tan and Kittler, 1994). This characteristic is interesting in the field of mobile robotics where such situations happen. Given that, if the acquisition conditions are rather stable, the number of data samples required to represent different elements that we want to identify can be reduced.

Table 1. Texture features computed from sum and difference histograms.

Texture feature	Equation
Mean	$\mu = \frac{1}{2}\sum_i i \cdot \hat{P}_{s(i)}$
Variance	$\frac{1}{2}\left(\sum_i (i - 2\mu)^2 \cdot \hat{P}_{s(i)} + \sum_j j^2 \cdot \hat{P}_{d(j)}\right)$
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	$\frac{1}{1+j^2}\sum_j \cdot \hat{P}_{d(j)}$

When the color information is available and suitable (i.e., terrestrial natural areas), 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.

3.2.1. Supervised Learning. Bayesian classification is used to identify region, this technique 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 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 the ones having the greatest discrimination power and uncorrelated. We 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 $P(X \mid C_i)$.

3.3. Object Identification

The nature (class) of an object perceived in the scene is obtained by comparing its attribute vector (computed from the 3D features and from the texture or color) with a database composed by different classes, issued from a learning step executed off-line.

This identification phase allows us to get a probabilistic estimation about the object nature. The label associated to an object will be exploited in order to detect possible incoherences at two levels:

- at first, in the modeling process, a 3D or 2D segmentation error will be detected if the extracted objects cannot be labelled by the identification function.
- then, in the tracking process, the nature of the landmark could be used in addition to the partial Hausdorff distance to detect possible tracking errors or drifts.

A Bayesian classification (Duda and Hart, 1973) is used in order to estimate the class membership for each object. The Bayesian rule is defined as

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

where

- $P(C_i)$ is the a priori probability that an object belongs to the class (C_i).
- $P(X \mid C_i)$ is the class conditional probability that the object attribute is X, given that it belongs to class C_i.
- $P(C_i \mid X)$ is the a posteriori conditional probability that the object class membership is C_i, given that the object attribute is X.

We have assumed equal a priori probability. In this case the computation of the a posteriori probability $P(C_i \mid X)$ can be simplified and its value just depend on $P(X \mid C_i)$.

The value of $P(X \mid C_i)$ is estimated by using k-nearest neighbor method. It consists in computing for each class, the distance from the sample X (corresponding to the object to identify, whose coordinates are given by the vector of 3-D information and luminosity features) to k-th nearest neighbor amongst the learned samples. So we have to compute only this distance (in common Euclidean distance) in order to evaluate $P(X \mid C_i)$. Finally, the observation X will be assigned to the class C_i whose k-th nearest neighbor to X is closest to X than for any other training class.

3.4. Experimental Results

To show the construction of the local model of the scene based on only 2D information, we present the process in a image. In the last phase of the local model, each region in the image has a class associated (nature). These regions were obtained from the color segmentation phase. However, the segmentation results in large regions, the regions do not always correspond to real objects in the scene. Sometimes a real element is over-segmented, consequently a fusion phase becomes necessary. In this step connected regions belonging to the same class are merged.

The coherence of the model is tested by using the topological characteristics of the environment (Murrieta-Cid, 1998). Possible errors in the identification process could be detected and corrected by using contextual information (i.e., grass cannot be surrounded by sky regions).

Figure 10 shows the original image. Figure 11 shows the color image segmentation and the identification of the regions. Labels in the images indicate the nature of the regions: (R) rock, (G) grass, (T) tree and (S) sky.

The Region at the top right corner of the image was identified as grass. However, this region has a relatively low probability (less than a given threshold) of belonging to this class, in this case the system can correct the mistake by using contextual information; this region is then relabeled as tree, Fig. 12 shows the final model of this scene. Figure 13 shows the gray levels used to label the classes. Figure 14 shows other scene and Fig. 15 shows the model.

Figure 10. Original image.

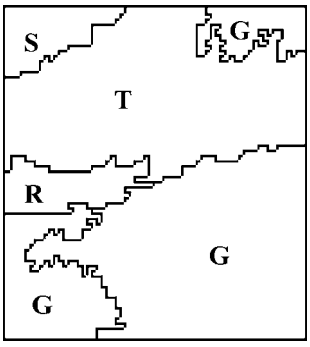


Figure 11. Segmentation and identification.

Figure 12. Final model.

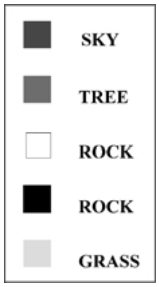


Figure 13. Classes.

Figure 14. Original image.

Figure 15. Local model.

4. Landmark Selection

The landmark selection phase is composed by two main steps. First, a local model is built from the first robot position in the environment. Then, by using this first local model, a landmark is chosen among the objects detected in this first scene.

A landmark is defined as a remarkable object, which should have some properties that will be exploited in the robot localization or in visual navigation. The two main properties which we use to define a landmark are:

- *Discrimination*. A landmark should be easy to differentiate from other surrounding objects.
- *Accuracy*. A landmark must be accurate enough so that it can allow to reduce the uncertainty on the robot situation, because it will be used to deal with the robot localization.

Depending on the kind of navigation performed (Section 2) the landmarks have different meaning. In trajectory-based navigation landmarks are useful to localize the robot (Ayala and Devy, 2000) and of course the bigger number of landmarks in the environment the better. For topological navigation the landmarks are seen as a sub-goal which the robot has to reach. For this last kind of navigation commutation of landmarks is an important issue. We are dealing with this task, based on the position of the landmark in the image (see Section 7, image 24).

Landmarks in indoor environments correspond to structured scene components, such as walls, corners, doors, etc. In outdoor natural scenes, landmarks are less structured. We have proposed several solutions like maxima of curvature on border lines (Devy and Parra, 1998), maxima of elevation on the terrain (Fillatreau et al., 1993) or on extracted objects (Betg-Brezetz et al., 1996).

In previous works we have defined a landmark as a bulge, typically a natural object emerging from a rather flat ground (e.g., a rock), only the elevation peak of such an object has been considered as a numerical attribute useful for the localization purpose. A realistic uncertainty model has been proposed for these peaks, so that the peak uncertainty is function of the rock sharpness, of the sensor noise and of the distance from the robot.

In a segmented 3D image, a bulge is selected as candidate landmark if:

- It is not occluded by another object. If an object is occluded, it will be difficult to find in the following images and will not have a good estimate on its top.
- Its topmost point is accurate. This is function of the sensor noise, resolution and object top shape.
- It must be in "ground contact".

These criteria are used so that only some objects extracted from an image are selected as landmarks. The most accurate one (or the more significant landmark cluster in cluttered scenes) is then selected in order to support the reference frame of the first explored area. Moreover, a specific landmark must be defined as the next tracked target for the tracking process. Different criteria, coming from higher decisional levels, could be used for this selection, for example:

- Track the sharper or the higher object: it will be easier to detect and to match between successive images.
- Track the more distant object from the robot, towards a given direction (visual navigation).
- Track the object which maximizes a utility function, taking into account several criteria (active exploration).
- Or, in a teleprogrammed system, track the object pointed on the 2D image by an operator.

In order to navigate during a long robot motion, a sequence of different landmarks (or targets) is used as sub-goal the robot must successively reach (Murrieta-Cid et al., 2001). The landmark change is automatic. It is based on the nature of the landmark and the distance between the robot and the target which represents the current sub-goal. When the robot attains the current target (or, more precisely, when the current target is close to the limit of the camera field of view), another one is dynamically selected in order to control the next motion (Murrieta-Cid et al., 1998b).

At this time due to integration constraints, only one landmark can be tracked during the robot motion. We are currently developing a multi-tracking method.

This landmark will be used for several functions:

- It will support the first reference frame linked to the current area explored by the robot so that, the initial robot situation in the environment can be easily computed.
- It will be the first tracked target in the 2D image sequence acquired during the next robot motion (tracking process fast-loop). If visual navigation is chosen in the higher level decision system as a way to define the robot motions during the exploration task, this same process will be also in charge of generating commands for the mobile robot and for the pan and tilt platform on which the cameras are mounted.
- It will be detected again in the next 3D image acquired in the modeling process, so that the robot situation could be easily updated, as this landmark supports the reference frame of the explored area.

Moreover, the first local model allows to initialize the global model which will be upgraded by the incremental fusion of the local models built from the next 3D acquisitions. Hereafter, the automatic procedure for the landmark selection is presented.

The local model of the first scene (obtained from the 3-D segmentation and identification phases) is used to select automatically an appropriated landmark, from an utility estimation based on both its nature and shape (Murrieta-Cid et al., 1998a). Landmarks can be used to both, localization and navigation tasks. Localization based on environment features improves the autonomy of the robot.

Figure 16 shows the original image, Fig. 17 shows the automatic selection of a landmark based on its nature and shape.

When several elements having the same nature are present in the scene, the local model of the scene can be used to select one according to its two-dimensional representation (i.e., the longest region belonging to the class rock, present in the image). It is also possible to track portions of landmark to decrease the computation running time of the tracking process. One criteria is to select the element with the largest elongation when there are several elements of the same nature. This criteria is as follows: The first step is to select the longest region in the image. The major vertical axis of the object is found, and a window is constructed around it. The window width is determined as a fraction of the size of the major vertical axis, only the points belonging

Figure 16. Original image.

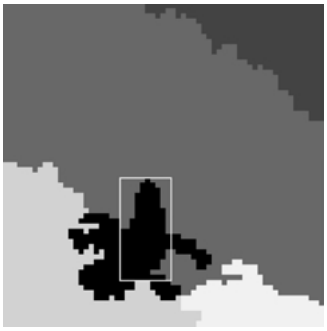


Figure 17. Landmark selection.

to the region of the class chosen and falling within the window are taken into consideration. In addition very narrow elements are avoided.

5. The Tracking Process (Fast-Loop)

The target tracking problem has received a great deal of attention in the computer vision community over the last years. Several methods have been reported in the literature, and a variety of features have been proposed to perform the tracking (Delagnes et al., 1994; Jiansho and Tomasi, 1994; Yue, 1995).

Our method is able to track an object in an image sequence in the case of a sensor motion or of an object motion. This method is based on the assumption that the 3D motion of the sensor or the object can be characterized by using only a 2D representation. This 2D motion in the image can be decomposed into two parts:

- A 2D image motion (translation and rotation), corresponding to the change of the target's position in the image space.
- A 2D shape change, corresponding to a new aspect of the target.

The target tracking results are used to perform robot localization from matching between landmarks used as targets and it is also used to keep the direction of a landmark in order to send the robot there (visual navigation).

Other works have used target tracking results to improve robot localization (Mallet et al., 2001). In this work the tracking is done by using a correlation function. The targets are small windows (typically 10×10 pixels) having some discrimination properties. The robot position estimation is improved by merging 2D information get from video image with 3D data. This work is similar to our approach, however there is a important conceptual difference, in our approach the targets are landmarks having a semantic meaning, one of ours final goals is to command the robot with semantic instead of numerical vectors. For instance the command of going from (x_1, y_1) to (x_2, y_2) can be replaced with "*Go from the tree to the rock*".

The tracking is done using a comparison between an image and a model. The model and the image are binary elements extracted from a sequence of gray levels images using an edge detector similar to Canny (1986).

This target tracking method is well adapted to natural environments because it does not need any kind of structured models. The method tracks a configuration of points. Besides, in natural environments there is enough texture, therefore it is possible to get points of maximal gradient. The method is based on the assumption that between to consecutive images the appearance of the configuration of points will not change drastically. This happens in non-structures environments contrarily to structured environments which are often modeled with polyhedral objects which quickly change their appearance when the sensor is in motion.

The target tracking method is well adapted for the environment type we are dealing with. Nevertheless when the tracking is performed over very complex images (too much texture) some errors can happen. The error can also occur when the robot motion between two consecutive scenes is large (see Fig. 23 VI.b) because the aspect and position of the target changes a great deal. In order to detect these errors the local model of the scene is built with a lower frequency than the target tracking process. The coherence of the both processes is checked by using the class of the target.

A partial Hausdorff distance is used as a resemblance measurement between the target model and its presumed position in an image.

Given two sets of points P and Q, the Hausdorff distance is defined as Serra (1982):

$$H(P, Q) = \max(h(P, Q), h(Q, P))$$

where

$$h(P, Q) = \max_{p \in P} \min_{q \in Q} \| p - q \|$$

and $\| \cdot \|$ is a given distance between two points p and q. The function $h(P, Q)$ (distance from set P to Q) is a measure of the degree in which each point in P is near to some point in Q. The Hausdorff distance is the maximum among $h(P, Q)$ and $h(Q, P)$.

By computing the Hausdorff distance in this way we obtain the most mismatched point between the two shapes compared consequently, it is very sensitive to the presence of any outlying points. For that reason it is often appropriate to use a more general rank order measure, which replaces the maximization operation with a rank operation. This measure (partial distance) is defined as Huttenlocher et al. (1993a):

$$h_k = K^{\text{th}}_{p \in P} \min_{q \in Q} \| p - q \|$$

where $K^{\text{th}}_{p \in P} f(p)$ denotes the $K^{-\text{th}}$ ranked value of $f(p)$ over the set P.

5.1. *Finding the Model Position*

The first task to be accomplished is to define the position of the model M_t in the next image I_{t+1} of the sequence. The search for the model in the image (or image's region) is done in some selected direction. We are using the unidirectional partial distance from the model to the image to achieve this first step.

The minimum value of $h_{k1}(M_t, I_{t+1})$ identifies the best "position" of M_t in I_{t+1}, under the action of some group of translations G. It is possible also to identify the set of translations of M_t such that $h_{k1}(M_t, I_{t+1})$ is no larger than some value τ, in this case there may be multiple translations that have essentially the same quality (Huttenlocher et al., 1993b).

However, rather than computing the single translation giving the minimum distance or the set of translations, such that its correspond h_{k1} is no larger than τ, it is possible to find the first translation g, such that its associated h_{k1} is no larger than τ, for a given search direction.

Although the first translation which $h_{k1}(M_t, I_{t+1})$ associated is less than τ it is not necessarily the best one, whether τ is small, the translation g should be quite good. This is better than computing all the set of valuable translation, whereas the computing time is significantly smaller.

5.2. *Building the New Model*

Having found the position of the model M_t in the next image I_{t+1} of the sequence, we now have to build the new model M_{t+1} by determining which pixels of the image I_{t+1} are part of this new model.

The model is updated by using the unidirectional partial distance from the image to the model as a criterion for selecting the subset of images points I_{t+1} that belong to M_{t+1}. The new model is defined as:

$$M_{t+1} = \{q \in I_{t+1} \mid h_{k2}(I_{t+1}, g(M_t)) < \delta\}$$

where $g(M_t)$ is the model at the time t under the action of the translation g, and δ controls the degree to which the method is able to track objects that change shape.

In order to allow models that may be changing in size, this size is increased whenever there is a significant number of nonzero pixels near the boundary and is decreased in the contrary case. The model's position is improved according to the position where the model's boundary was defined.

The initial model is obtained by using the local model of the scene previously computed. With this initial model the tracking begins, finding progressively the new position of the target and updating the model. The tracking of the model is successful if:

$$k1 > fM \mid h_{k1}(M_t, I_{t+1}) < \tau$$

and

$$k2 > fI \mid h_{k2}(I_{t+1}, g(M_t)) < \delta,$$

in which fM is a fraction of the number total of points of the model M_t and fI is a fraction of image's point of I_{t+1} superimposed on $g(M_t)$.

5.3. *Our Contributions Over the General Tracking Method*

Several previous works have used the Hausdorff distance as a resemblance measure in order to track an

object (Huttenlocher et al., 1993b; Dubuisson and Jain, 1997). This section enumerates some of the extensions that we have made over the general method (Murrieta-Cid, 1997).

- Firstly, we are using an automatic identification method in order to select the initial model. This method uses several attributes of the image such as texture and 3-D shape.
- Only a small region of the image is examined to obtain the new target position, as opposed to the entire image. In this manner, the computation time is decreased significantly. The idea behind a local exploration of the image is that if the execution of the code is quick enough, the new target position will then lie within a vicinity of the previous one. We are trading the capability to find the target in the whole image in order to increase the speed of computation of the new position and shape of the model. In this way, the robustness of the method is increased to handle target deformations, since it is less likely that the shape of the model will change significantly in a small δt. In addition, this technique allows the program to report the target's location to any external systems with a higher frequency (for an application see Becker et al. (1995)).
- Instead of computing the set of translations of M_t, such that $h_{k1}(M_t, I_{t+1})$ is no larger than some value τ, we are finding the first translation whose $h_{k1}(M_t, I_{t+1})$ is less than τ. This strategy significantly decreases the computational time.

Recently other work (Ayala et al., 2000) has improved the target tracking approach here presented, the robustness has been increased by (1) a refinement of the target model, (2) usage of a target search strategy that sweeps space of possible translation following a spiral trajectory (having as result an error mean of target image localization equal to zero) and (3) an alternative strategy to select the target by doing a motion detection in the image, based on background model provided by a Gaussian mixture.

5.4. Experimental Results: Tracking

The tracking method was implemented in C on a real-time operating system (Power-PC), the computation running time is dependent on the region size examined to obtain the new target position. For sequences the code is capable of processing a frame in about 0.25 seconds. In this case only a small region of the image is examined given that the new target position will lie within a vicinity of the previous one. Processing includes, edge detection, target localization, and model updating for a video image of (256×256 pixels).

Figure 18 show the tracking process in a lunar-like environment. Figure 18(a) shows initial target selection, in this case the user specifies a rectangle in the frame that contains the target. An automatic landmark (target) selection is possible by using the local model of the scene. Figure 18(b)–(e) shows the tracking of a rock through an image sequence. The rock chosen as target is marked in the figure with a boundary box. Another boundary box is used to delineate the improved target position after the model updating. In these images the region being examined is the whole image, the objective is to show the capacity of the method to identify a rock among the set of objects.

Next example illustrates the target tracking process in a terrestrial natural environment.

We underline that the local model of the scene is used to select automatically an appropriated target (see Figs. 16 and 17). This approach allows the selection of a landmark as target based on its nature and shape.

Figures 19–22 show the tracking of a rock, this rock is marked in the figure with a boundary box. Another larger boundary box is used to delineate the region of examination.

Only a small region of the image is examined to obtain the new target position, as opposed to the entire image. Another larger boundary box is used to delineate the region of examination. In this manner, the computation time is decreased significantly. The local exploration of the image is justified because if the execution of the code is quick enough, the new target position will then lie within a vicinity of the previous one.

6. The Global Model Building and Robot Localization (Slow-Loop)

The local models extracted from the acquired 3D images are fused in order to build a global model in an incremental way. After each 3D acquisition, a local model is firstly built from the 3D image, by the use of the method described in Section 3. Then the global model must be updated by merging it with the local one. This fusion function allows to improve the robot estimate position and attitude (Sutherland and Thompson, 1994; Smith et al., 1990).

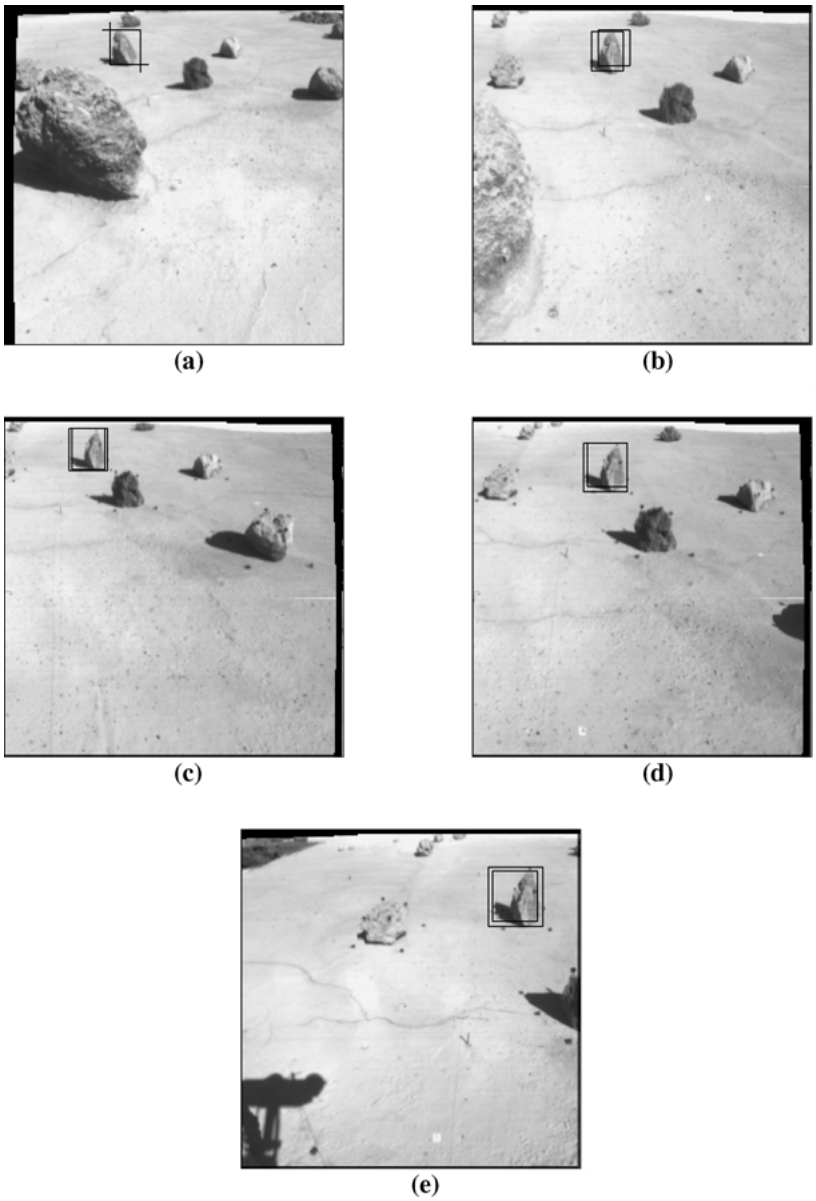


Figure 18. Visual tracking.

6.1. Robot Localization and Global Model Fusion

The modeling process has an estimate of the robot situation provided by internal sensors (on the LAMA robot: Odometers and inclinometers). This estimate may be quite inaccurate, and moreover systematically implies cumulative errors. The robot situation is represented by an uncertainty vector $(x, y, z, \theta, \phi, \psi)$. The estimated errors are described by a variance-covariance matrix. When these errors become too large, the robot must correct its situation estimate by using other perceptual data; we do *not* take advantage of any a priori knowledge, such as artificial beacons, nor of external positioning systems, such as GPS. The self-localization function requires the registration of local models built at successive robot situations. Some works deal with this problem by performing optical odometry (Mallet et al., 2001).

Figure 19. Visual target tracking.

Figure 20. Visual target tracking.

Figure 21. Visual target tracking.

Figure 22. Visual target tracking.

The global model here proposed has two main components: The first one describes the topological relationships between the detected ground areas, the second one contains the perceived informations for each area. The topological model is a connectivity graph between the detected areas (a node for each area, an edge between two connected areas). In this paper, we focus only on the knowledge extracted for a given area. The information related to a given area corresponds to the list of objects detected on this area, the ground model, and the list of the different robot positions when it has explored this area.

The global model construction requires the matching of several landmarks extracted in the local model and already known in the current global model. This problem has been solved using only the 3D images (Betg-Brezetz et al., 1996), but the proposed method was very unreliable in cluttered environment (too many bad matchings between landmarks perceived on multiple views). Now, the matching problem is solved by using the visual tracking process. The landmark selected as the target at the previous iteration of the modeling process, has been tracked in the sequence of 2D images acquired since then. The result of the tracking process is checked, so that two situations may occur:

- in the local model built from the current position, we find an object extracted from the 3D image, which can be mapped on the region of the tracked target in the corresponding 2D image. If the label given by the identification function to this region, is the same than the label of the target, then the tracking result is valid and the tracked landmark gives a first good matching from which other ones can be easily deduced.
- if some incoherences are detected (no mapping between an extracted 3D object and the 2D tracked region, no correspondence between the current label of the tracked region and the previous one), then some specific procedure must be executed. At this time, as soon as no matchings can be found between the current local and global models, a new area is open. It means that the landmark selection procedure is executed again in order to select the best landmark in the local model as the new reference for the further iterations.

When matchings between landmarks can be found, the fusion functions have been presented in Betg-Brezetz et al. (1996). The main characteristics of our method is the uncertainty representation; at instant k,

a random vector $\mathbf{X}_k = [\mathbf{x}_r^T\ \mathbf{x}_1^T \ldots \mathbf{x}_N^T]_k^T$ and the associated variance-covariance matrix represent the current state of the environment. It includes the current robot's situation and the numerical attributes of the landmark features, expressed with respect to a global reference frame. Robot situation and landmark feature updates are done using an Extended Kalman Filter (EKF).

6.2. Experimental Results of Modeling Using 2D and 3D Information

Figure 23 shows a partial result of the exploration task, involving concurrently the modeling and the tracking processes. Figure 23 I.a shows the video image, Fig. 23 I.b presents the 3-D image segmentation and classification, two grey levels are used to label the classes (rocks and soil). Figure 23 I.c shows the first estimation of the robot position. A boundary box indicates the selected landmark (see Fig. 23 I.a). This one was automatically chosen by using the local model. The selection was done by taking into account 3-D shape and nature of the landmark.

Figure 23 II and 23 III shows the tracking of the landmark, which is marked in the figure with a boundary box. Another larger boundary box is used to delineate the region of examination.

Figure 23 IV.a presents the next image of the sequence, Fig. 23 IV.b shows the 3-D segmentation and identification phases used to build the local model. The visual tracking is employed here to solve the matching problem of landmark's representation between the different perceptions. Figure 23 IV.c presents the current robot localization, the local model building at this time is merged to the global one. In this simple example, the global model contains only one ground area with a list of three detected landmarks and a list of two robot positions.

The target tracking process goes on in the next images of the sequence (see Fig. 23 V and 23 VI.a). The robot motion between the image V and VI.a was too important, so the aspect and position of the target changes a great deal; it occurs a tracking error (see the in Fig. 23 VI.b, the window around the presumed tracked target). A new local model is built at this time (Fig. 23 VI.b). The coherence of the both processes (local model construction and target tracking) is checked by using the nature of the landmark. As the system knows that the target is a rock, this one is able to detect the tracking process mistake given that the model of the landmark (target) belongs to the class soil.

7. Integrated System

The complete system here proposed is shown in Fig. 2. During robot motion a slow and a fast processes are simultaneously executed. The slow process is used to build a landmark-based model of the environment. The fast process is used to track the landmarks. The coherence of the results of the executed task is checked by comparing the result of both processes. The testing is done to the frequency of the slowest processes. Currently the fast process is running to approximately 4 Hz and the slow is running to 0.1 Hz.

Robot visual navigation is done by using the proposed system. In order 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.

We illustrate this task with a experiment carried out with the mobile robot LAMA. Figure 24(a) shows the video image, (b) presents the 3-D image and (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 Fig. 24 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 Murrieta-Cid et al. (1998a) 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 Fig. 24 III presents the new landmark selection based on image segmentation and classification. The next sequence of tracking is shows on the line IV of Fig. 24 and the next landmark commutation is presents on line V. Finally on the line VI the robot continue navigation task.

7.1. Experiments of Simultaneous Localization and Modeling (SLAM)

We illustrate this task with an experiment carried out in the EDEN site at LAAS-CNRS. In this work SLAM task is based on landmark extraction. The strategy to select the landmarks is the one presented on Section 7. Left column of Fig. 25 shows 2-D images corresponding to left stereo-vision camera. On these images the rocks selected as target and the zone where the target is looking for are shown. The results obtained regarding environment modeling are shown on the second

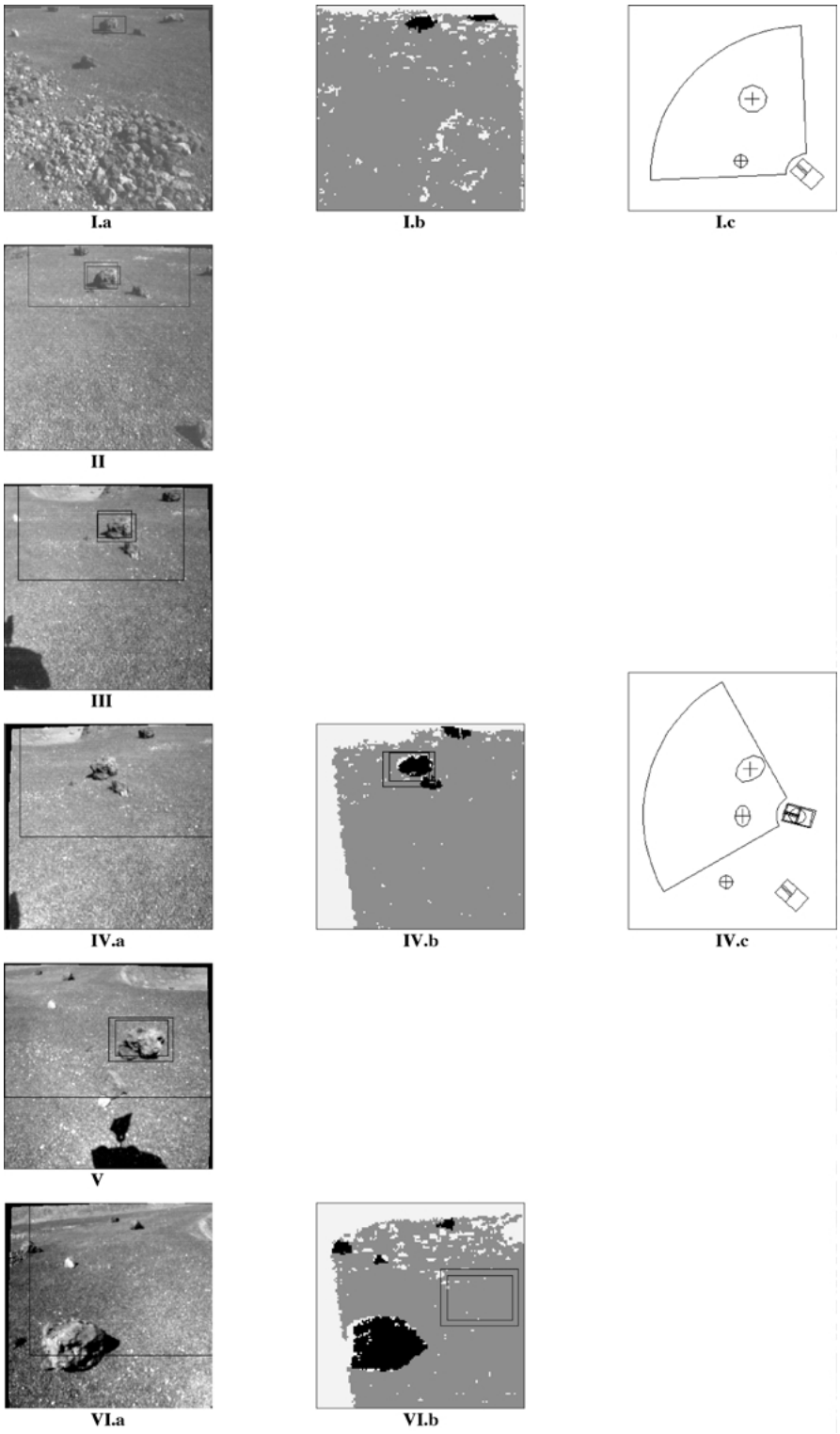


Figure 23. 3-D robot localization.

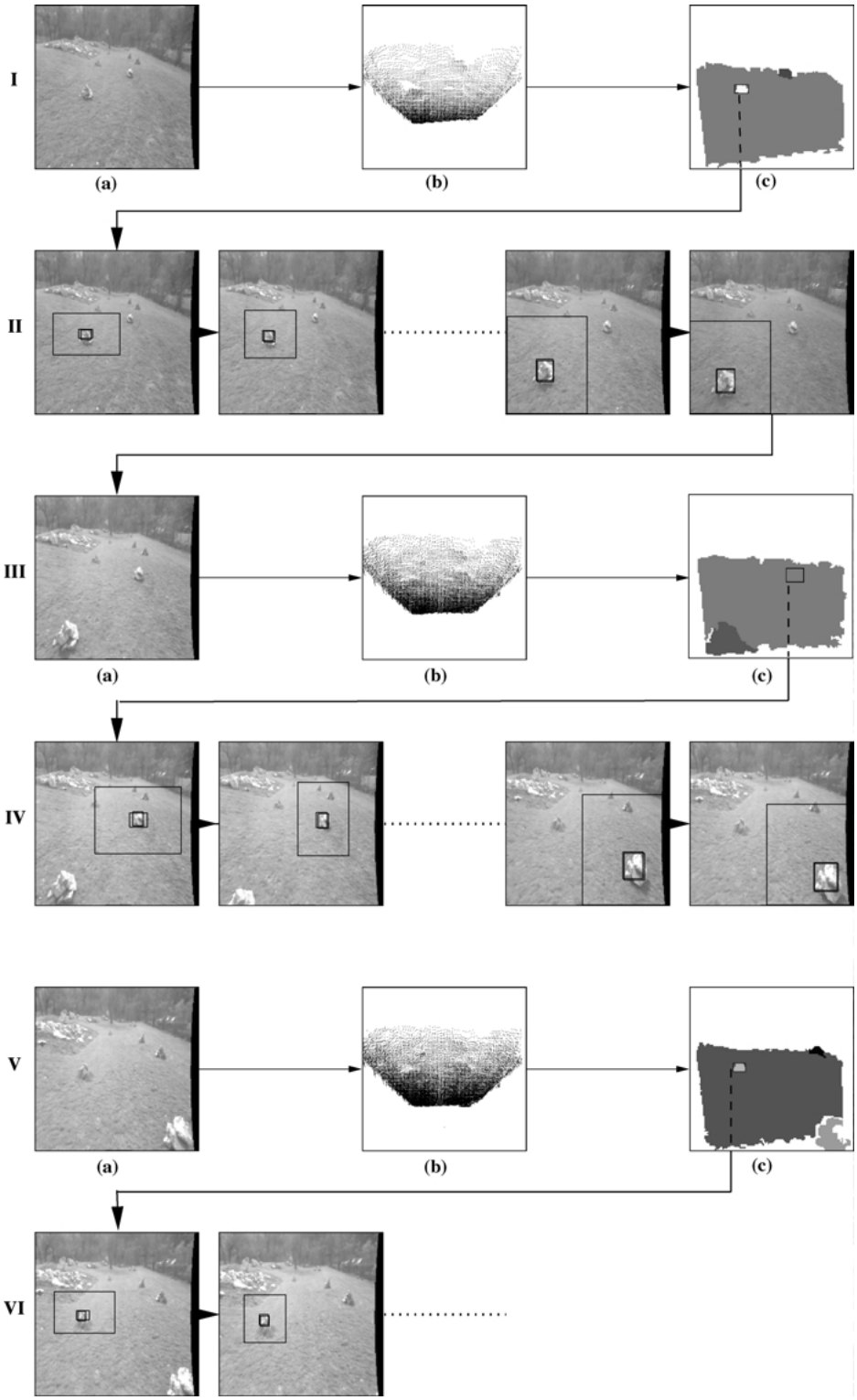


Figure 24. Visual robot navigation based on landmarks.

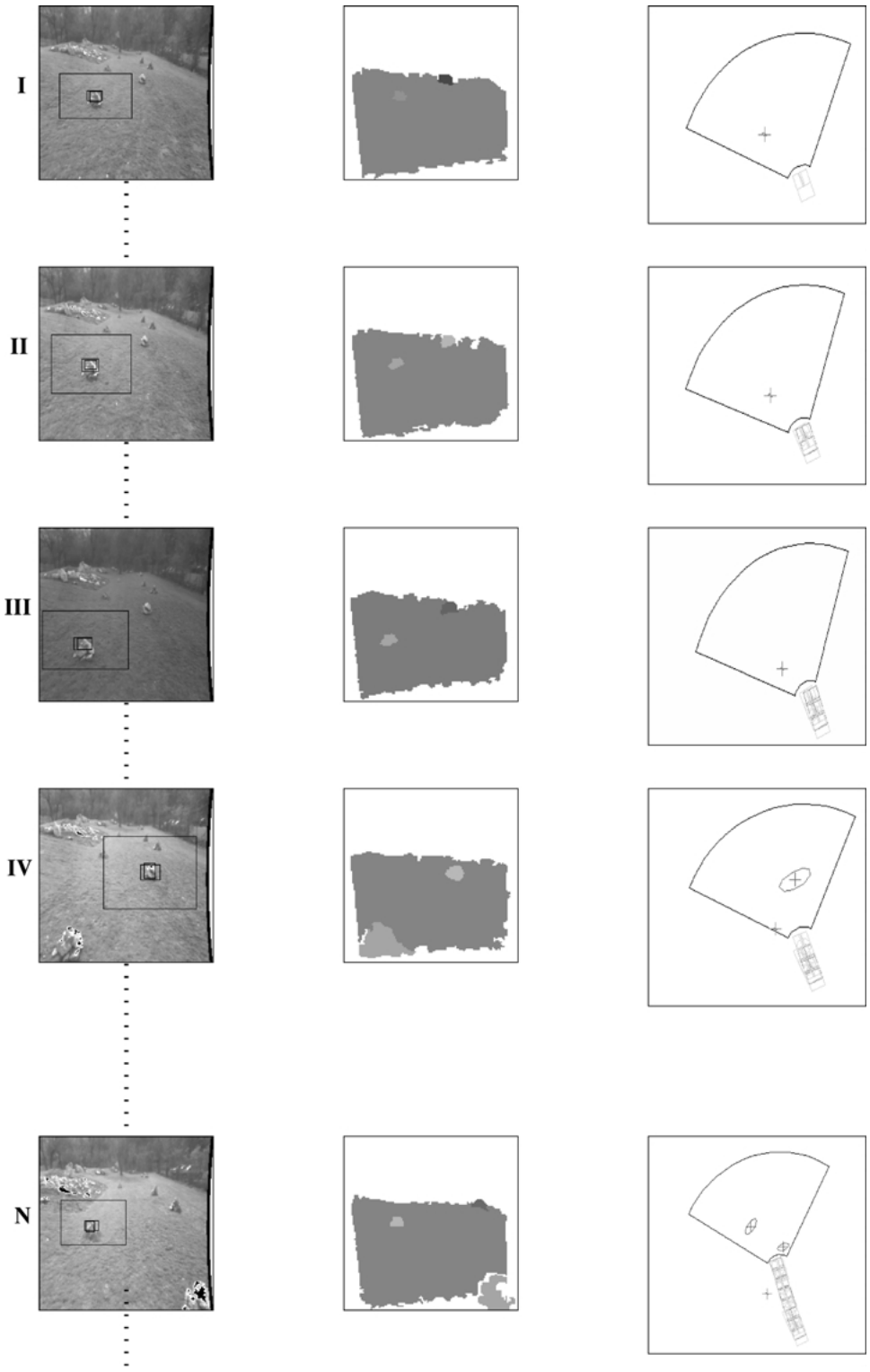


Figure 25. Simultaneous localization and modeling (SLAM) based on landmarks.

column. The maps of the environment and the localization of the robot are presented on the third column. On the row "I" the robot just takes one landmark as reference in order to localize itself. On the last row the robot uses 3 landmarks to perform localization task, the robot position estimation is shown by using rectangles. The most important result here is that the robot position uncertainty does not grow thanks to the usage of landmarks. The landmarks allow to stop the incremental growing of the robot position uncertainty.

8. Conclusion and Future Work

The work presented in this paper concerns the environment representation and the localization of a mobile robot which navigates in a planetary environment or terrestrial natural areas.

A local model of the environment is constructed in several phases:

- region extraction: firstly, the segmentation gives a synthetic representation of the environment.
- object characterization: each object of the scene is characterized by using 3-D features and its texture or/and its color. Having done the segmentation texture color and 3-D features can be used to characterize and to identify the objects. In this phase, visual attributes are taken into account to profit from its power of discrimination. The texture and color attributes are computed from regions issued from the segmentation, which commonly give more discriminant informations than the features obtained from an arbitrary division of the image.
- object identification: the nature of the elements (objects and ground) in the scene is obtained by comparing an attribute vector with a database composed by different classes, issued from a learning process.

The local model of the first scene is employed in order to select automatically an appropriate landmark. The matching problem of landmark's is solved by using a visual tracking process. The global model of the environment is updated at each perception and merged with the current local model. The current robot's situation and the numerical attributes of the landmark features are updated by using an Extended Kalman Filter (EKF).

Comparing the approach here proposed with our previous work, one important improvement is the current segmentation algorithm. Here we are using an unsupervised classification method in order to automatically generate classes in the attribute space. Thanks to this method our segmentation is more robust. In our system, the most difficult task to accomplish is segmentation, so if this step is robust, the whole system will be too.

Comparing our approach with other outdoor map building methods, the main contributions are: (1) The use of semantic labeling of objects and regions which allows to command the robot using semantic instead of numeric vectors. (2) The use of tracking of landmarks to aid matching perceived the local scene model with a global world model.

Based on intensive evaluation of our previous method we found out that the main problem to fuse local models into a global one is the matching of objects perceived in multiple views acquired during the robot motion. The tracking method allows to keep the correspondence between some of the landmarks during an image sequence simplifying the match among the remaining landmarks.

Some possible extensions to this system are going on: firstly, we plan to study image preprocessors that would enhance the extraction of those image features that are appropriate to the tracking method. Secondly, we plan to include new classes (e.g., rocky soil and ground depressions) to improve the semantic description of the environment.

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 natural areas 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 (Rubner et al., 1998). 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 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 enhancement of our method (Murrieta-Cid et al., 2002).

We are also working in a more complete topological representation of the environment in order to move the robot along very large paths where the environment can change significantly. Finally, this approach is being modified to detect new specific entities such as country roads. We also have the intention to apply this work in agricultural tasks.

Acknowledgments

The authors thank Nicolas Vandapel, Maurice Briot, Victor Ayala, Raja Chatila and Simon Lacroix for their contributions to the development of the ideas presented in this paper and to the implementation of some of the software. This work was funded by CONACyT México (PhD scholarship and project J34670-A), by COLCIENCIAS Colombia, France Foreign Office (PhD scholarship) and ITESM Campus Ciudad de México.

References

Asada, M. 1988. Building a 3D world model for a mobile robot from sensory data. In *Proc. International Conference on Robotics and Automation (ICRA)*, Philadelphia, USA.

Ayala, V. and Devy, M. 2000. Active selection and tracking of multiple landmarks for visual navigation. In *Proc. 2th International Symposium on Robotics and Automation (ISRA)*, Monterrey, México.

Ayala, V., Parra, C., and Devy, M. 2000. Active tracking based on Haussdorf matching. In *Proc. IEEE 15th International Conference on Pattern Recognition (ICPR)*, Barcelona, Spain.

Becker, C., González, H., Latombe, J.-L., and Tomasi, C. 1995. An intelligent observer. In *Proc. International Symposium on Experimental Robotics (ISER)*.

Betg-Brezetz, S., Chatila, R., and Devy, M. 1994. Natural scene understanding for mobile robot navigation. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, San Diego, USA.

Betg-Brezetz, S., Hébert, P., Chatila, R., and Devy, M. 1996. Uncertain map making in natural environments. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Minneapolis, USA.

Betgé-Brezetz, S., Chatila, R., and Devy, M. 1995. Object-based modelling and localization in natural environments. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Osaka, Japan.

Bulata, H. and Devy, M. 1996. Incremental construction of a landmark-based and topological model of indoor environments by a mobile robot. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Minneapolis, USA.

Canny, J. 1986. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6).

Chatila, R. and Laumond, J.-P. 1985. Position referencing and consistent world modeling for mobile robots. In *Proc IEEE Int. Conf. on Robotics and Automation (ICRA)*.

Dedeoglu, G., Mataric, M., and Sukhatme, G. 1999. Incremental, on-line topological map building with a mobile robot. In *Proc. Mobile Robots XIV SPIE99*, Boston, USA, pp. 129–139.

Delagnes, P., Benois, J., and Barba, D. 1994. Adjustable polygons: A novel active contour model for objects tracking on complex background. *Journal on Communications*, 45:83–85.

Devy, M. and Parra, C. 1998. 3D scene modelling and curve-based localization in natural environments. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Leuven, Belgium.

Dubuisson, M. and Jain, A. 1997. 2D matching of 3D moving objects in color outdoors scenes. In *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*.

Duda, R.O. and Hart, P.E. 1973. *Pattern Classification and Scene Analysis*, Wiley & Sons.

Dumaine, M., Perrinot, O., Guennon, R., Hurteau, R., and DeSantis, R.M. 2001. Odometry and DGPS integrated navigatation system for ground vehicles. In *Proc. International Conference on Field and Service Robotics (FSR)*, Helsinki, Finland.

Fillatreau, P., Devy, M., and Prajoux, R. 1993. Modelling of unstructured terrain and feature extraction using B-spline surfaces. In *Proc. International Conference on Advanced Robotics (ICAR)*, Tokyo, Japan.

Haddad, H., Khatib, M., Lacroix, S., and Chatila, R. 1998. Reactive navigation in outdoor environments using potential fields. In *Proc. International Conference on Robotics and Automation (ICRA)*, Leuven, Belgium.

Hebert, M., Caillas, C., Krotkov, E., Kweon, I., and Kanade, T. 1989. Terrain mapping for a roving planetary explorer. In *Proc. International Conference on Robotics and Automation (ICRA)*, vol. 2.

Huttenlocher, D.P., Klanderman, A., and Rucklidge, J. 1993a. Comparing images using the hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863.

Huttenlocher, D.P., Rucklidge, W.J., and Noh, J.J. 1993b. Tracking non-rigid objects in complex scenes. In *Proc. Fourth International Conference on Computer Vision (ICCV)*.

Jiansho, S. and Tomasi, C. 1994. Good features to track. In *Proc. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*.

Krotkov, E., Caillas, C., Hebert, M., Kweon, I., and Kanade, T. 1989. First results in terrain mapping for a roving planetary explorer. In *Proc. NASA Conference on Space Telerobotics*.

Kweon, I.S. and Kanade, T. 1991. Extracting topological features for outdoor mobile robots. In *Proc. International Conference on Robotics and Automation (ICRA)*, Sacramento, USA.

Lacroix, S., Chatila, R., Fleury, S., Herrb, M., and Simeon, T. 1994. Autonomous navigation in outdoor environment: Adaptive approach and experiment. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, San Diego, USA.

Mallet, A., Lacroix, S., and Gallo, G. 2001. Position estimation in outdoor environments using a pixel tracking and stereovision. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, USA.

McKerrow, P.J. and Ratner, D. 2001. Navigating an outdoor robot with simple discontinuous landmarks. In *Proc. Fourth European Workshop on Advanced Mobile Robots (EUROBOT)*, Lund, Sweeden.

Metea, M.B. and Tsai, J. 1987. Route planning for intelligent autonomous land vehicles using hierarchical terrain representation. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Raleigh, USA.

Murrieta-Cid, R. 1997. Target tracking method based on a comparation between an image and a model. Technical Report Num. 97023, LAAS CNRS, written during a stay at Stanford University.

Murrieta-Cid, R. 1998. *Contribution au développement d'un système de Vision pour robot mobile d'extérieur*. PhD Thesis (in French), INPT, LAAS CNRS, Toulouse, France.

Murrieta-Cid, R., Briot, M., and Vandapel, N. 1998a. Landmark identification and tracking in natural environment. In *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Victoria, Canada.

Murrieta-Cid, R., Parra, C., Devy, M., and Briot, M. 1998b. Contribution on vision and modelling. In *Proc. International Symposium on Robotics and Automation (ISRA)*, Saltillo, México.

Murrieta-Cid, R., Parra, C., Devy, M., and Briot, M. 2001. Scene modelling from 2D and 3D sensory data acquired from natural environments. In *Proc. IEEE International Conference on Advanced Robotics (ICAR)*, Budapest, Hungary.

Murrieta-Cid, R., Parra, C., Devy, M., Tovar, B., and Esteves, C. 2002. Building multi-level models: From landscapes to landmarks. In *Proc. IEEE International Conference on Robotics and Automation (ICRA)*, Washington, USA.

Ohta, Y. 1985. *Knowledge-Based Interpretation of Outdoor Natural Color Scenes*, Morgan Kaufman: Palo Alto, CA.

Otsu, N. 1979. A threshold selection method from gray-level histrograms. *IEEE Transaction on Systems, Man and Cybernetics*, 9(1):62–66.

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

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

Rosenblum, M. and Gothard, B. 2000. A high fidelity multi-sensor scene understanding system for autonomous navigation. In *Proc. IEEE Intelligent Vehicles Symposium (IV)*, Detroit, USA.

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

Rucklidge, W. 1997. Efficiently locating objects using the hausdorff distance. *International Journal of Computer Vision*, 24(3):251–270.

Saber, E., Tekalp, A.M., Eschbach, R., and Knox, K. 1996. Automatic image annotation using adaptative color classification. *Graphical Models and Image Processing*, 58(2):115–126.

Serra, J. 1982. *Image Analysis and Mathematical Morphology*, Academic Press: London.

Smith, R.C., Self, M., and Cheeseman, P. 1990. Estimating uncertain spatial relationships in robotics. *Autonomous Robot Vehicules*.

Sutherland, K.T. and Thompson, B. 1994. Localizing in unstructured environments: Dealing with the errors. *IEEE Transactions on Robotics and Automation*.

Tan, T.S.C. and Kittler, J. 1994. Colour texture analysis using colour histogram. In *Proc. IEE Vis. Image Signal Process.*

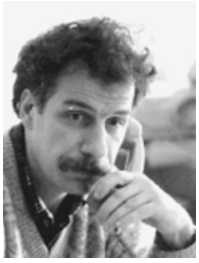
Unser, M. 1986. Sum and difference histograms for texture classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Vandapel, N., Moorehead, S., Whittaker, W., Chatila, R., and Murrieta-Cid, R. 1999. Preliminary results on the use of stereo color cameras and laser sensors in Antarctica. In *Proc. 6th International Symposium on Experimental Robotics (ISER)*, Sydney, Australia.

Du, Y. and Crisman, J. 1995. A color projection for fast generic target tracking. In *Proc. International Symposium on Intelligent Robots and Systems (SIRS)*, Pisa, Italy.

Rafael Murrieta-Cid received the B.Sc. degree in Industrial Physics Engineering from ITESM campus Monterrey (México) in 1990, the M.Sc. degree in Automatic Manufacturing Systems from ITESM campus Monterrey in 1993 and DEA "Diplome d'Etudes Approfondies" (French Master) in Signals and Images from INP of Toulouse in 1995. In 1996, he was visitor student at the Stanford Robotics Laboratory. In November 1998 he received the Ph.D. from INP of Toulouse France. His Ph.D research was done at the RIA group of the LAAS/CNRS, in Toulouse. In 1998–1999, he was Postdoctoral researcher in the Computer Science Department at Stanford University. Rafael Murrieta-Cid is currently Assistant Professor in the Electrical Engineering Department at ITESM campus México City and is working on Motion Planning, Perception and Experimental Mobile Robotics.

Carlos Parra got his B.Sc. degree in Electronic Engineering from Pontificia Universidad Javeriana of Bogotá-Colombia in 1992 and then a M.Sc. degree in Electrical Engineering from Universidad de los Andes of Bogotá-Colombia in 1994. He got a DEA in Control Science and Industrial Computer Science from Paul Sabatier University. In Mars 1999 he received the Ph.D. from Paul Sabatier University in Toulouse-France. His Ph.D. research was performed at the RIA group at the LAAS/CNRS, in Toulouse. In 1999, he did a Postdoctoral research in the LAAS/CNRS. Carlos Parra is currently Associate Professor in the Electronic Department at Pontificia Universidad Javeriana. His research interest include perception, mobile robots for natural environments, applications of computer vison and experimental mobile robotics.



Michel Devy got his degree in Computer Science Engineering in 1976 from IMAG, in Grenoble (France), and received is Ph.D. in 1980 in Toulouse (France). Since 1980, he has participated to the Robotics and Artificial Intelligence group of LAAS-CNRS; his research is devoted to the application of computer vision in Automation and Robotics. He has been involved in numerous national and international projects, about Manufacturing Applications, Mobile Robots for space exploration or for civil safety, Vision technology for Intelligent Vehicles, 3D modelling for body modelling. He is now Research Director at CNRS, head for the Perception Area in the Robotics Group of LAAS-CNRS and his main scientific topics concerns Perception for Mobile Robots in natural or indoor environments.