SERVIROB: A Mobile Robot for Restoration

L. ACOSTA, E.J. GONZALEZ, J.N. RODRIGUEZ, J.A. MENDEZ, A. HAMILTON, S. HERNÁNDEZ
Grupo de Computadoras y Control
University of La Laguna
Avenida Astrofísico Fco. Sánchez S/N, CP 38207, La Laguna
SPAIN

Abstract: - In this paper a highly specialized autonomous robot is presented. The task of the robot is the setting and clearing of tables in a restaurant. The environment in which the robot will operate will be controlled and well known. The first objective of the robot is to navigate around the room to find collectable sites, that is, tables. Once the robot has located a table, its taks consists of analyzing the scene and extracting all the necessary information to collect the distinct objects located on it. The application is designed to identify and collect the following objects: dishes, bottles, glasses, forks, spoons, and knives.

Key-Words: - Autonomous robotics.

1 Introduction

This work presents an application (SERVIROB) that consists of an autonomous robot for setting and clearing tables in a restaurant. To accomplish this task the robot is autonomously navigating in the room and has to fulfill various objectives.
- Navegation
- Identification of the work areas (tables)
- Identification of the objects to be collected.

In this study, the following objects have been considered:
-White dishes of two different sizes
-Metal forks, spoons and knives
-Clear glasses
-Clear bottles
- Manipulation of these objects.These objects must be collected and deposited in individual recipients. A specially designed manipulator is mounted on

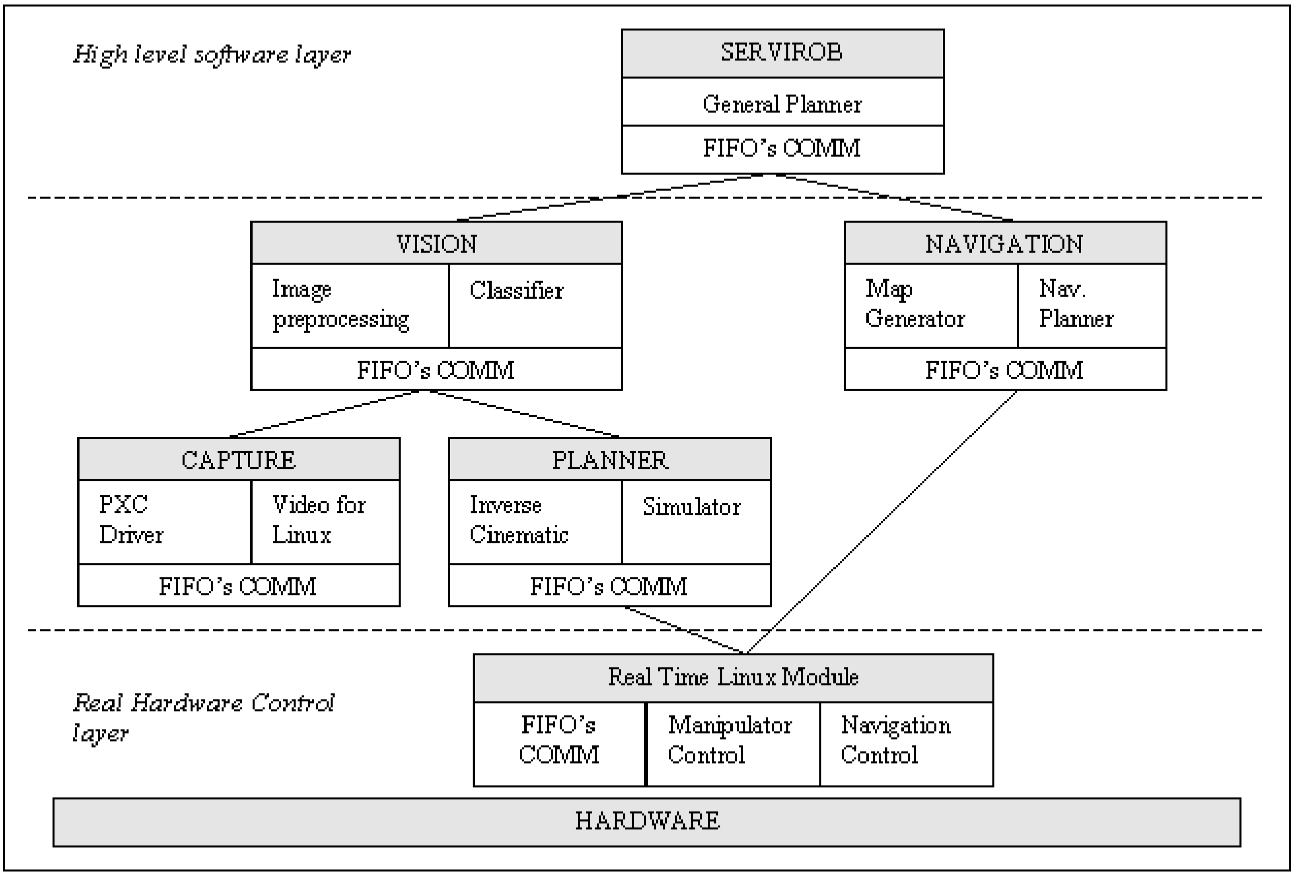

Fig. 1 Software Structure

the robot to accomplish this task.
Figure 1 shows the set of layers that compound the software structure. The fact of working with a layer structure provides the necessary characteristics in relationship with real time requirements ,module comunications and system scalability. The module comunication is carried out through a set of FIFO (first in first out) files provided by the OS (Operative System). The OS kernel has been modified to change all the interruption requests by a set of macros that allow us to setup a real time layer between the kernel and the robot hardware. In this way the system has a real time response and allow us to setup parameters like priority proccesses and the maximun time for task execution. Over this layer a set of modules has been developed (manipulator Planner, mobile base Planner and Image Capture modules) that provide the interface between real time hardware control layer and high level software like vision module or navigation module. The last layer situated on top of the software structure is the general planner. This layer defines the general robot behavior in order to resolve all posible situations that can be found surrounding the problem that the robot tries to solve.

2 System mechanics, Control and Power Supply electronics

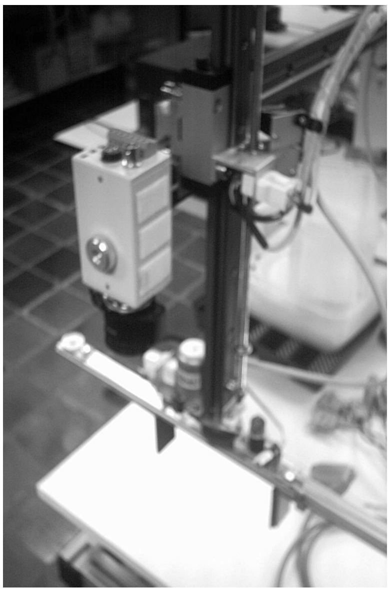

Figure 2: Detail of the gripper

In this section, the authors will describe briefly the system mechanics, control and power supply.
System mechanics tries to minimize the effect of possible accidental collisions. The prototype has been provided with a special grabber (Figure 2) for catching objects until 40 cm with the following features:
- Elastic material. This way, the task of catching the objects is easier.
- Pressure detector.
- A sensor that detects the position where the objects are in the table surface.

For the control of the engines, the authors have designed and developed ad hoc an electronic system. This system can be adapted to any prototype with DC engines and two-channel encoders. This system consists of a rack where the following boards are conected:
- PWM generation board
- Alarms generation board
- Power supply and encoder counters board

3 Control System

The authors have developed a tool that allows the robotic arm and the mobile structure to carry on those ordinary tasks necessary for a control closed-loop. The control scheme, considering the model of the robot, computates the non-linear component, and cancels it through a feedforward scheme, obtaining a linear model of the system this way. Although initially the authors had initially applied a PD controller to this linear system, finally they have adopted a linear predictive controller.

On the other hand, the authors have decided to implement a closed-loop control (taking the speed as variable to control) for each motive wheels of the system. With these controllers, the mobile robot is able to cover exactly the desired distance..

4 Vision System

This is one of the most important points of the described work. The vision module of the complete application analyses the scene environment when SERVIROB is in a position to collect objects. The objective of this module is to provide the necessary information to the main module of SERVIROB so that it can perform pick and place tasks in the table. Although the environment is controlled and the objects to be collected are well defined, working in a real scene implies that objects can be overlapped or partially hidden. Also rubbish of the table is a noisy element in the scene that makes difficult the identification. In view of this, the vision system must be capable of extracting the necessary information to identify the objects in the scene even under adverse conditions.

a) DEFINITION OF COLLECTABLE OBJECTS

The objects that can be found in the scene will be described by a set of geometric primitives, more precisely, circumferences and straight line segments. These primitives will constitute the basis of a language that allows the definition of all the objects that can be identified. Combining the description of all the possible primitives in a object together with the relationships amongst them, a vector of characteristics for each object is obtained. This vector is easily obtained from the images of the working scene, even when working with non-ideal conditions. As an example, in Figure 3 the definition of a spoon is shown. The relationships that these primitives must hold in order to define a spoon are imposed by the relative spatial location between them.

Doing the same with all the objects that the robot must recognise, a knowledge base is built. This knowledge base will be used by the vision system in the processing of the acquired images.

This method has two main advantages. First and most important, because in the procedure isolated characteristics with a given relationship are used to define the objects, the presence of all these characteristics is not necessary in order to obtain a positive detection. In this way, overlapped or partially hidden objects can be identified.

The second advantage is that the use of a visual language to define the objects allows to create a user interface where the number of recognizable objects can be increased easily. This property is important as it allows the application of the system to recognise different types of utensils.

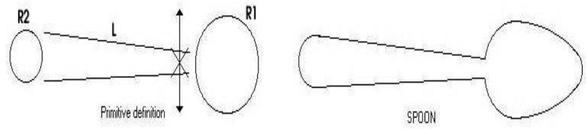

Figure 3: Definition of a spoon by the vision system

The analysis is carried out by means of an adaptive iterative process. The general algorithm repeats the process shown in Figure 4 for each class of predefined object. The object order for detection is also prespecified and is imposed by a set of rules defined by the user depending on the specific problem to be solved. For the application presented in this paper these rules are:

a) Objects whose altitude may affect the manipulator trajectories in the collection task are collected first (bottles and glasses).

b) Once the bottles and glasses (in this order) are collected, forks, spoons and knives are set as the next target. The order is defined by the number of primitives that identify each object. The risk or error is reduced applying first the more restrictive classifiers (with more primitives and relationships to hold).

c) It has to be taken into account that some objects (forks, knives and spoons) can be on the

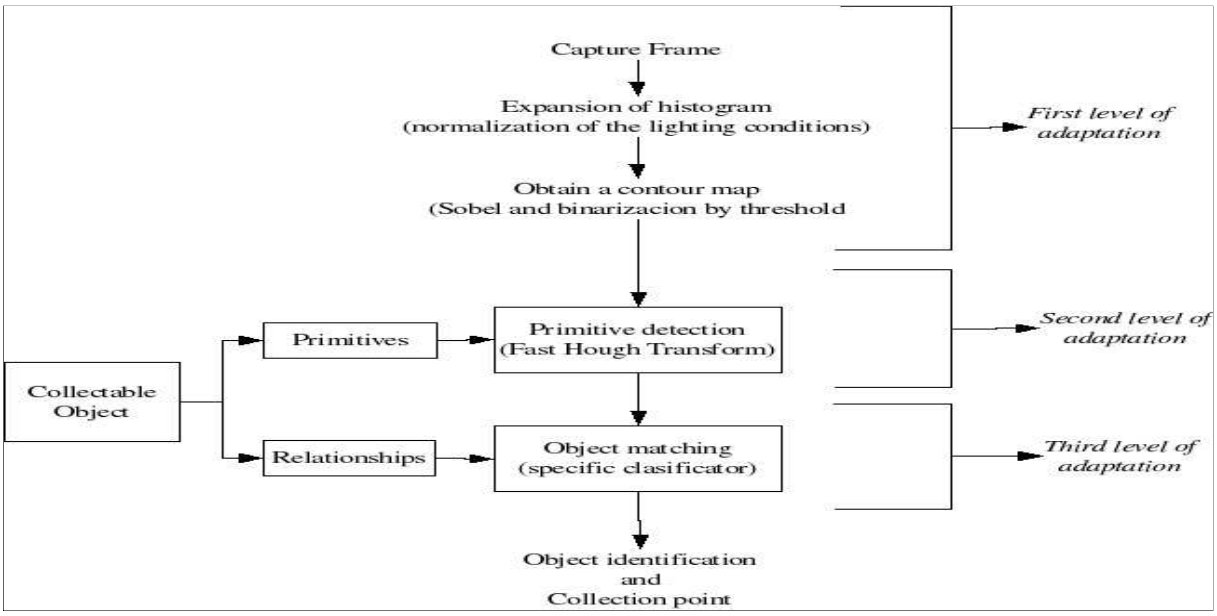

Figure 4: Designed algorithm for the SERVIROB Vision system

dishes. So these objects are collected before the dishes.

b) ANALYSIS PROCESS

The first step is to capture the work scene. The capture module has been designed to work with two video cameras in order to obtain stereo vision.

Once a scene frame is captured, the **preprocessing module** is started. It has to be taken into account that, if a positive detection is obtained in a method iteration, the detected object is collected. This implies a change in the working scene for the next iteration. That is why after the frame capture an adaptation process is carried out. This process is called *first adaptation process* and consists of a normalization of the lighting conditions. This is done because when an object is removed from the scene, the brightness and contrast conditions change significantly. Even the object distribution changes if overlapping existed before the collection.

The preprocessing module allows the application of several operators to achieve normalization of the brightness and contrast conditions in the image. Thus, the primitive detection efficiency is improved. The relief operators implemented are histogram matching, linear piecewise transformations, equalization and non linear parameterized operations.

The system implemented allows captures to try distinct strategies using the operators library in order to minimize the noise in the image. Once the best sequence for the specific conditions is defined, that sequence will be applied automatically on each on line capture.

After the normalization process is finished, the system begins the identification of collectable objects. This task is performed by the **classifier module**. This module first tries to locate primitives in the scene, then verifies the relationships for the specific object and finally generates the votes in the parameter space, that allows to find the object detected with a higher probability of success. Also, at this stage the adaptation parameters are evaluated to check its goodness. The first step in this module is to obtain a contour map of the scene. This map can be obtained in different ways because different operators were implemented to create it. After several experiments it was found that a first order operator, like Sobel operator, together with a binarization operator is the best choice. When Sobel operator is applied, a thick contour map is obtained. Later it will be shown that this map is essential for the construction of the vote map. Next, the specific primitives for the object are identified.

A. Primitive detection

The basic primitives defined in this work are circumferences and straight line segments. The system generates a list of primitives, grouped according to their properties. The circumferences will be grouped by its radio length. The segments will be grouped according to their length. One Hough operator has been implemented for the detection of circumferences and a different Hough operator for the detection of straight lines. Both operators are highly parameterized [1,2]. These parameters are manipulated by an adaptation process called *second level of adaptation* process. In this way, the tolerance of the operators are tuned. More precisely the Hough operator for circumference detection has five parameters to be adjusted in this level (Rmin, Rmax, threshold, round, fit) as shown in table 1.

R min	102 pixels
R max	107 pixels
threshold	0.6
round	2

Tab. 1. Hough Transform parameters

R min and R max are the radius range where the search is performed. The threshold parameter is used to select the set of admisible solutions. The round parameter refers to the length of the exploration cell in the image. Finally, the fit parameter indicate the minimun pixel percentage that must appear in the image to be considered as a valid primitive.

In this way, and increasing the tolerance gradually, the scene is analyzed to find first the primitives that are less affected by noise. These primitives define the objects that are best candidates to be collected (isolated and well defined). Later, the primitives that are more affected by noise are searched (overlapped and partially hidden). As commented above, the primitives are defined by a set of properties that allows their grouping. For circumferences, the system is asked to look for a given radio. For segments, the system looks for a given length. These are provided to the system by means of a specific range. For instance, for a circumference of radio R, a minimum and a maximum radio around R is provided. The system tries to find primitives into this range. In this way, there is an adaptation to small deviations of the distance between the camera and the table produced by vibrations of the structure or irregularities of the ground. These variations produce changes in the dimensions of the objects and, hence, noise in the scene.

B. Object matching

To understand this procedure, the detection of a spoon is considered. In this case, there are three list of primitives: circumferences with radius R1, circumferences with radio R2 and segments with length L1.

For a spoon, the circumference of radio R1 (referred as A) must be at a distance of 120 pixels from the circumference of radio R2 (B). The vector (A,B) provides a slope by means of which the candidate segments to form a spoon are found. Of all the found segments, only those that are located between the two circumferences and minimize the distance (A,B) are considered. Although the system starts from strict spatial relationships amongst primitives, when the verification of those relationships is done, a *third level of adaptation* is defined.

This level allows for small deviations around the original distances of the object definition. The tolerance degree increases or decreases depending on the quality of the detection.

It can be observed that the detection process provides a great number of primitives that can be part of the analyzed object. This occurs because of the implementation properties of the Hough operators and because of the edge detection.

As commented above, the operator used to generate the contour map produces a map with thick contours. This fact together with the capability of the primitive detection operator for adaptation to small variations [3,4] in the size of the objects produces redundant information. This redundancy is used to generate the votes in the parameter space that allows evaluating the quality of the detection. From this information, the object with less error probability is finally chosen. Each primitive holding the constraints imposed by the relationships that define the object is provided with a specific number of votes. More votes are assigned to those primitives that fit better the original model.

5 Ultrasound System and Navigation Module

This section will describe briefly how the mobile robot is able to locate the tables, that is, the areas where it can catch the dishes, glasses, forks, spoons and knives. SERVIROB is provided with 15 ultrasound sensors, located in two different heights (the scheme is shown in Figure 5). With this disposition, the robot is able to distinguish the walls and columns (the obstacle would be detected to the same distance) from the tables (the sensor located at the low level would detect the table, while the high level sensor would not). These ultrasound systems, connected through an I2C bus, are able to measure distances till 6 meters, although from a practical point of view those measures over approximately one meter are not excessively reliable. The devices are managed by a microcontroller ATMEL AT90S8515. This microcontroller sends the information obtained by the ultrasound devices to the navigation module when required.

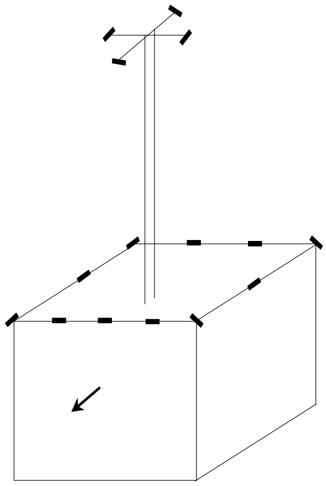

Figure 5: Ultrasound system

The distances measured by the ultrasound sensors are sent to a navigation module, together with the encoders and localization system data. This navigation module uses all those data, making a dynamic map of the environment. This map is based on cells and its design has followed the object-oriented programming techniques (OOP). In this sense, the implementation has been carried out employing the C++ programming language over a GNU/Linux system.

6 Localization System

The mobile robot is provided with an innovative low-cost localization system. An indoor system to determine the position and orientation of the mobile robot has been designed. The system has an emitter and a receiving device. A general vision of the system is shown in Figure 6. The emitter (a laser pointer) placed on a wall at the contour carries out a sweeping at fixed speed. The receiving device located at the top of the robot has 32 photovoltaic

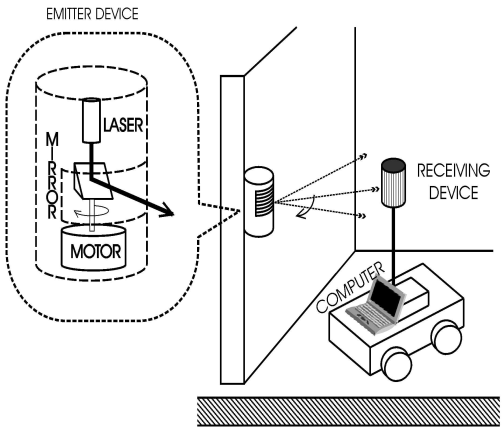

Figure 6: General view of the localization system

cells. The calculus of position and orientation is made by measuring the times of impact of the laser on the cells. The measure of the coordinates will be made based on the module and argument of a polar system, where the laser emitter is at the origin of a system of polar coordinates.

The authors have used the property that "the subtended angle for the cylinder with regards to the beacon is a function of the distance". Simply measuring the lapsed time since the laser light enters and leaves the system of photo-sensors, the distance can be obtained. The laser is always rotating at the same direction and speed. A synchronization system sends a radio pulse at the beginning of each laser sweeping. This synchronism pulse resets a counter. The value of the counter, when the laser impacts on the robot, is proportional to the angle [5].

7 Conclusions

This paper presents an autonomous robot for setting and clearing tables in a controlled environment that emulates a restaurant. For its implementation, several modules have been integrated (control, localization…).

With respect to specific modules, the mobile robot is provided with an innovative low-cost localization system. An indoor system to determine the position and orientation of the mobile robot has been designed. The system has an emitter and a receiving device. The emitter (a laser pointer) placed on a wall at the contour carries out a sweeping at fixed speed. On the other hand, the receiving device located at the top of the robot has 32 photovoltaic cells. The calculus of position and orientation is made measuring the times of impact of the laser on the cells.

Other important module to consider is a versatile vision system that has been implemented with a high degree of effectiveness. With the information provided by the vision system, the robot manipulator will be able to calculate the best trajectories to pick objects from the table. This was implemented using three level of adaptation in the recognition process. The experiments performed revealed a satisfactory performance of the system in the identification of objects, even when noisy images were processed. The mean time for the recognition of objects in complex test scenes is highly reasonable for those situations.

8 Acknowledgement

This work has been financed by the Spanish government, project # DPI2001-3681.

References:

[1] Leavers, V. F. *Shape detection in computer vision using the Hough transform* / V. F. Leavers: Springer, cop. 1992 ISBN3-540-19723-0

[2] A. S. Aguado, E. Montielb and M. S. Nixonc, *Invariant characterisation of the Hough transform for pose estimation of arbitrary shapes*, Pattern Recognition Volume 35, Issue 5, May 2002, pp 1083-1097.

[3] - A. A. Kassim, T. Tanb and K. H. Tana, *A comparative study of efficient generalised Hough transform techniques*, Image and Vision Computing Volume 17, Issue 10, August 1999, pp 737-748.

[4] - P.S.Naira, and A. T. Saunders Jr., *Hough transform based ellipse detection algorithm*, Pattern Recognition Letters Volume 17, Issue 7, 10 June 1996, pp 777-784.

[5] Sergio Hernández, Jesús M. Torres, Carlos A. Morales, Leopoldo Acosta *A New Low Cost System for Autonomous Robot Heading and Position Localization in a Closed Area*. Autonomous Robots. Volume 15, Number 2, September 2003, Autonomous Robots, pp. 99-110