Mobile System for Regular Polyhedron Recognition and Location

Mauricio Gabriel Orozco del Castillo

Unidad Profesional Interdisciplinaria en Ingeniería
y Tecnologías Avanzadas (UPIITA)
Instituto Politécnico Nacional (IPN)
México, D.F.
Instituto Mexicano del Petróleo (IMP)
mgabriel@imp.mx

Abstract. The project consists in the development of a mobile system capable of recognizing regular polyhedrons (platonic solids), and also their location inside a limited environment, that is a simplification of the real world, which is, by nature, highly complex. The recognition of these objects will take place through a webcam located on a mobile robot (both connected to a personal computer) that will move through the environment. The mobile robot will move through the environment in a straight line while it obtains information of the environment and of the platonic solids it encounters. After the robot finishes moving through a fixed path, the system creates a 3D virtual map with the location of the encountered polyhedrons, their type, size and orientation for a posterior analysis by the user or human operator through an interaction software that also was developed.

Keywords: Image Processing, 3D Computer Vision, Pattern Recognition, Mechatronics

1 Introduction

The ability to navigate in an autonomous way is fundamental for many animals and for every intelligent organism [2]. The upside of mobile robotics is that it focuses on problems related with the understanding of large-scale space, this is, regions of the space that are considerably larger than those that can be seen from a single point of view[3]. The behavior in a large scale environment not only implies dealing with an increased acquisition of knowledge, estimation of positional error, and the ability to recognize objects or places, but it also requires that all of the latter be exhibited simultaneously.

Usually, a single fixed camera is unable by itself to localize structures in three dimensions; it obtains the direction towards the structure, but not the distance to it. If only one camera is available, another mechanism is needed to recover the "missing" depth information. Many of the vision robotic systems are based on cameras that passively observe their environment. On the other hand, active vision systems are systems in which the observer is active (in motion) and not passive (still). In an active vision paradigm, the visual processing is dedicated to particular components in the image, and does not process the whole of it.

The creation of truly intelligent machines necessarily implies movement inside of our already dynamic world, and vision for a clear perception of it [10]. In this paper we propose a mobile vision system capable of recognizing and locating objects in an active way, using only a single camera. We choose regular polyhedrons as the objects to be analyzed because they are a well established group in geometry, have a moderate complexity and they represent in a good measure and in a general manner, the forms and characteristics of real world objects [11]. The goal is to show how a good knowledge of the environment and of certain objects can help us determine 3D information from 2D images without using any other tools, and to create a virtual 3D environment in which we can navigate using a computer.

2 Physical Construction

Due to the fact that the development of a moving vehicle for the robot is not the main goal of the project, we developed a simple one to achieve movement. A rail with a belt was used for a robot to move over it. A stepper motor was used and was adjusted to the rail's belt to convert circular motion into linear motion. The vehicle also has another stepper motor to control the direction in which the camera "sees". The motor rotates the structure on which the camera is located. This way we are able to move through the environment and to look around it in search for objects.

Our environment will have a black background, over which the figures will be located. Choosing a black background allows us to obtain a bigger contrast between it and the objects, facilitating their recognition.

Illumination is a problem that affects all vision systems. To solve this issue we attached two halogen bulbs to the moving vehicle so illumination remained constant as the camera moved. The bulbs were located symmetrically at the sides of the vehicle. It is important to note that the camera and the bulbs are inclined.

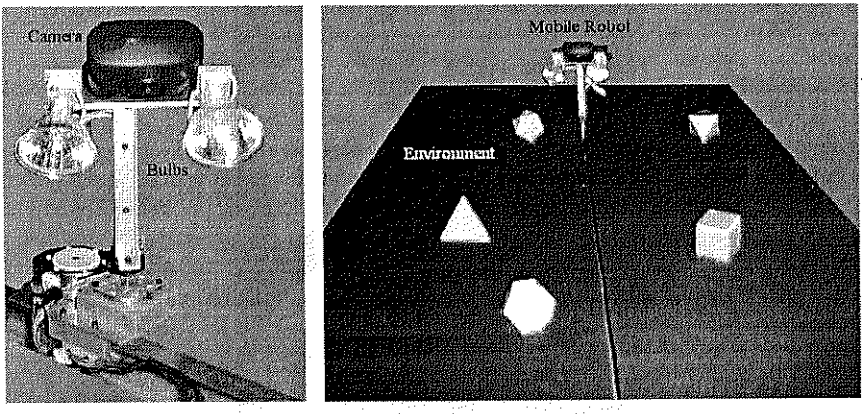


Fig. 1. Pictures of the mobile robot and of the mobile robot placed in the limited environment along with some of the polyhedrons to be recognized

3 The Regular Polyhedrons

As stated before, the objects to be recognized will be regular polyhedrons, also known as platonic solids or 3D regular politopes. There are only five of these: Tetrahedron, Hexahedron (cube), Octahedron, Dodecahedron and Icosahedron. A clear perception and a good understanding of these figures is needed not only to understand the objective of the system, but also to obtain the maximum number of tools to facilitate their recognition on latter stages. The regular polyhedrons are, by nature, 3D regular politopes. The word "politope" refers to closed volumes whose surface is composed of flat faces. When we talk about regular politopes, we talk about convex figures, with all of their faces being equal and with adjacent faces forming equal angles. We can summarize the properties of the objects of our study as: connected, convex, flat-faced, enclosed, with equal faces between them, and with equal angles between adjacent faces.

4 Image Processing

The image was filtered to minimize noise as in every image processing system, then the image was tresholded with an experimentally calculated value to separate the object from the background, and finally analyzed to discriminate the final pixels that formed part of the background instead of the object. Examples of this process are shown in Fig. 2. Once we have converted the original images into binary ones, it is a lot easier and faster to work with them.

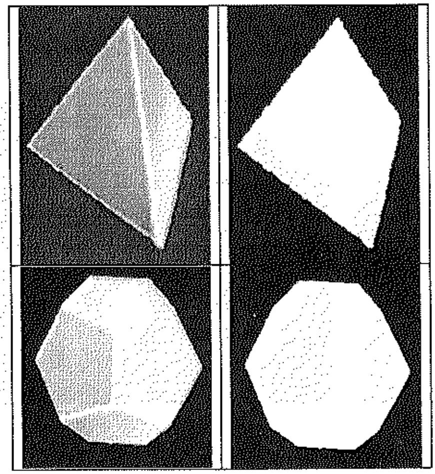


Fig. 2. Image processing on a tetrahedron and on a dodecahedron. We omit the information inside the objects and focus only on their convex hull

4.1 Locating the Vertices

The main objective of processing the images is to obtain the vertices of the objects. This is accomplished through an iterative method that we propose based on the digital chord property [8] of the figure.

The first step is to locate some initial vertices. This is easily done locating the extreme points of the figure: top, bottom, left and right. After this, the iterative method consists of measuring the perpendicular distance between all the border points of the image between two vertices and the line formed by joining them (Fig. 3). If the maximum distance measured is greater than an experimentally obtained treshold, then the point is a vertex. This threshold was obtained by trial and error using some test images of the five different objects. Once the first iteration is complete with vertices 1-2, 2-3, 3-4 and 4-1, if new vertices were encountered, these are re-numbered clock-wise and then the process repeats until no more vertices are found. This process is illustrated in Fig. 3.

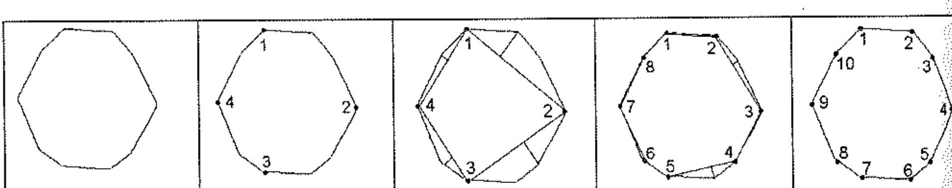


Fig. 3. The process of obtaining vertices on the bi-dimensional image of a dodecahedron

5 Obtaining the Location, Size and Orientation of the Objects

A single camera is unable to localize structures in three dimensions, so some other method must be used to recover the missing 3rd dimension. One of these methods is the Ground Plane Assumption [1]. We know that the objects are standing on the ground plane. This information, coupled with a calibrated camera are sufficient to localize the structure in 3D space. For this method to be valid, we must confirm that the points under study are laying on the ground. Due to the characteristics of the objects, it is evident that the two lowest points in the image representing object's vertices always lay on the ground, therefore, these will be the points used to determine the location, size and orientation of the objects.

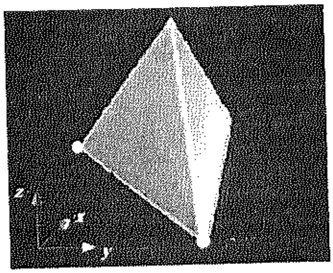


Fig. 4. Points used to determine the location, orientation and size of the objects

5.1 Geometrical Approach to the Ground Plane Assumption

To obtain the distance on which a point is located within the camera range we propose a geometrical approach, based on the knowledge of the properties shown in Fig. 5.

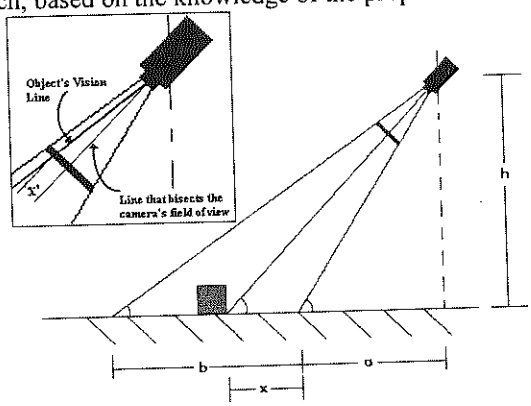


Fig. 5. Vision model showing the properties' values needed *a-priori* to determine the depth of an object

The objective of this procedure is to obtain x from the model variables: a, b, h and x'. Through a simple mathematical development it can be shown that the final value for x is given by equations 1 through 7.

$$\operatorname{tg}\theta = \frac{h}{a+b} \qquad (1)$$

$$\operatorname{tg}\phi = \frac{h}{a} \qquad (2)$$

$$\operatorname{tg}\alpha = \frac{h}{x+a} \qquad (3)$$

$$k = \operatorname{tg}\frac{\phi}{2} + \operatorname{tg}\frac{\theta}{2} \qquad (4)$$

$$m = 1 - \operatorname{tg}\frac{\phi}{2}\operatorname{tg}\frac{\theta}{2} \qquad (5)$$

$$f = \frac{m + \dfrac{kh}{a+b}}{2\left(k - \dfrac{hm}{a+b}\right)} \qquad (6)$$

$$x = \frac{x'(am+hk)+(hfm-afk)}{fk-x'm} \qquad (7)$$

Following the axes convention used in Fig. 2, and using a geometric approach similar to the one above, we can determine the y coordinate for the object using the model shown in Fig. 6. This procedure is a little longer and more complex than the one explained before, therefore it will not be shown in this paper.

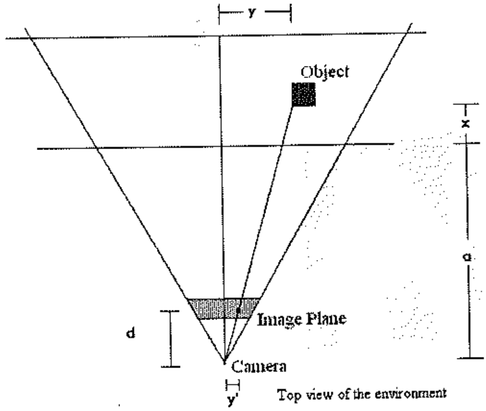


Fig. 6. Top view of the environment showing the model used to obtain the coordinate y. Knowing the camera range and the depth of the object we can know the value of y

5.2 Obtaining the Size of the Objects

Calculating the coordinates of the point closest to the camera can lead to determine the position of the object, but the size and the orientation of it are still missing. To determine these other characteristics we followed a similar approach.
Let's call $(x_1, y_1, 0)$ and $(x_2, y_2, 0)$ the coordinates of the vertices that are closest and second closest to the camera respectively (as done in section 4.1). Now, the length L of an edge of the figure would be the Euclidean distance between these two points as show in equation 8.

$$L = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \qquad \textbf{(8)}$$

Knowing the length of one edge, we can start applying the knowledge of the objects to then be able to rebuild the entire figure, since all the edges of it have the same length.

5.3 Obtaining the Orientation

To obtain the orientation we need to use again the coordinates of the two vertices given by $(x_1, y_1, 0)$ and $(x_2, y_2, 0)$. The angle θ, as shown in Fig. 7, that determines the orientation of the object can have two different values, depending on which side of the object contains the second vertex that is closest to the camera. For simplicity here we will just establish the equation for the general case, where

$$\theta = \text{arctg}\left(\frac{y_2 - y_1}{x_2 - x_1}\right). \tag{9}$$

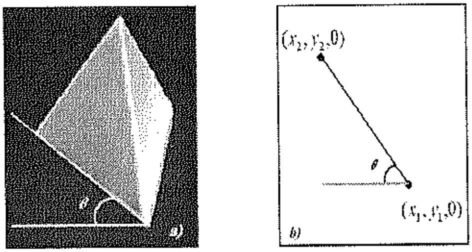


Fig. 7. The angle θ. a) Its physical representation and b) its analytical representation. This angle represents the orientation of the object in the environment

5.4 Adjusting the Location and Orientation

Since the position of the object is calculated assuming that the camera is at the origin of the environment, and since it usually won't because it is moving with the vehicle, we need to adjust the final coordinates of the object to a global origin, defined at the beginning of the path of the vehicle. So, let's call (x_p, y_p) the coordinates of the vertex defined by (x_1, y_1) but according to the global origin coordinates.

$$x_p = x_1 + x_c \tag{10}$$

$$y_p = y_1 \tag{11}$$

where x_c is the position of the vehicle when the image is obtained.

Now, since the angle by which the camera obtains the image depends of the movement of the motor controlling it, it must be taken into account. Let's call ϕ to the angle by which the camera observes the scene, described in Fig. 8.

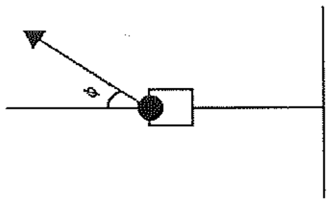


Fig. 8. The angle ϕ and its representation. This angle describes how the camera observes the scene, and it has obvious effects on the location of the objects

It can be seen that the global position and orientation of the object depend of this angle ϕ. The final coordinates of the global position would be then defined by:

$$x_p = x_c + (x_1)(\cos\phi) + (y_1)(\sin\phi) \tag{12}$$

$$y_p = (y_1)(\cos\phi) + (x_1)(\sin\phi) \tag{13}$$

and the global orientation θ_p by:

$$\theta_p = \theta + \phi \tag{14}$$

6 Recognizing the Objects

To be able to recognize the detected objects we followed a probabilistic approach. First we analyzed the objects' convex hull only, and not the object with all of its edges. Next we determined some characteristics of the convex hull. We have to keep in mind that these characteristics or features must remain non-dimensional, so they will be immune to the size of the objects. Let's define

$$d_E = \{d_{E1}, d_{E2}, \ldots, d_{En}\} \tag{15}$$

as the set of edge lengths of the figure,

$$d_c = \{d_{c1}, d_{c2}, \ldots, d_{cn}\} \tag{16}$$

as the set of distances from the center of mass of the figure to its vertices, and

$$d_N = \{d_{N1}, d_{N2}, \ldots, d_{Nn}\} \tag{17}$$

as the set of distances between all of the vertices of the figure. The characteristics taken into account were the ones shown in Table 1.

Table 1. Characteristics taken into account to differentiate the objects

Roundness	$mean(d_E)/max(d_E)$	$min(d_E)/mean(d_E)$	$mean(d_c)/max(d_c)$
$min(d_c)/mean(d_c)$	$mean(d_N)/max(d_N)$	$min(d_N)/mean(d_N)$	Number of vertices

It is important to note that the values of all of these characteristics are normalized to fall in the range $(0,1)$, so initially, all will have the same effect on the result.

Initially we acquired hundreds of test images, from which a data set was built, against which all new images could be compared. We then determine the probability that a new image belonged to the class (object) *i* by the equation

$$p(x \| Ci) = e^{-\left(\frac{1}{2}\right)\left(\frac{X-\mu}{\sigma}\right)^2} \quad \mathbf{(18)}$$

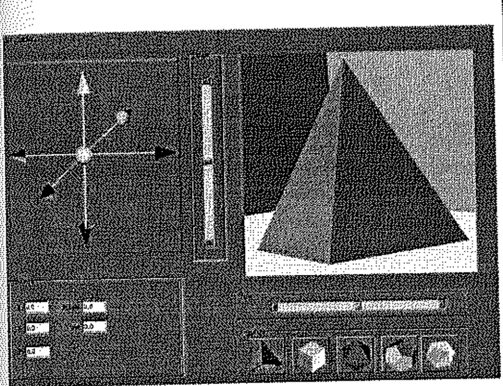
where X is the value obtained from any of the characteristics taken into account mentioned above, Ci represents the class i, μ the mean of the characteristic of the data set in study, and σ the standard deviation of the characteristic of the data set in study.

We take into account that there exists an *a-priori* and an *a-posteriori* probability that an object with determined characteristics belongs to a given class. The vertices will determine our *a-priori* knowledge, while the rest of the characteristics will determine the *a-posteriori* knowledge. The assumption for the *a-priori* knowledge would be, for example, to state: *"A figure with 4 vertices has greater probability of being an hexahedron than an octahedron"*, and so on. The *a-posteriori* knowledge is given by the normal density functions of the remaining characteristics, given by equation 18. So, for the general case, where *m* is the number of classes, and *n* the number of characteristics, we have

$$p(Ci | x_1, x_2, \ldots, x_n) = \frac{p(x_1, x_2, \ldots, x_n | Ci) P(Ci)}{\sum_{i=1}^{m} p(x_1, x_2, \ldots, x_n | Ci) P(Ci)}. \quad \mathbf{(19)}$$

7 Virtual Environment

The virtual environment was made using the OpenGL. This allowed us to create a very complete scenario without using very complex techniques. It was intended as much as possible to re-create the real scenario in the virtual environment, using all the characteristics obtained of the objects. In this environment we can freely move and see around it, examine the objects and the whole scene from any point of view. These lead to a better understanding of the real scene than if we only look at the 2D images resulting of the navigation through the real environment. Some examples of this transformation and a screenshot are shown in Fig 9.



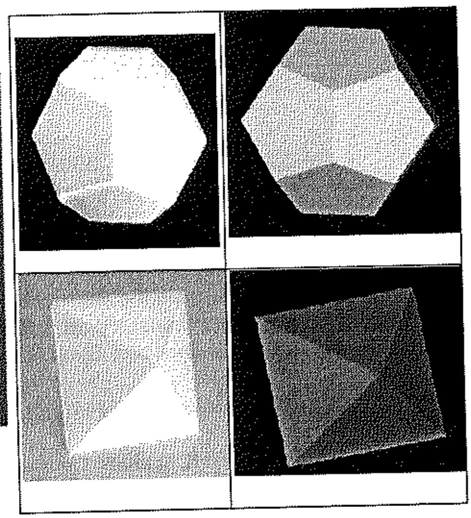


Fig. 9. a) Screenshot of the graphic interface for the virtual navigation and b) real and virtual images of a dodecahedron and of an octahedron showing the similarities between them

8 Results and Conclusions

- The knowledge of the environment is a strong help for the understanding and reconstruction of scenes
- Vision represents a powerful sensitive medium for artificial systems as agents. It can bring very detailed information of a scene, so its use appears fundamental for mobile robots' navigation.
- The image processing affects inevitably the precision of this and any other system. There is no image operation that warrants the preservation of the characteristics of the objects.

- A probabilistic classifier doesn't have a 100% accuracy. The classifier for this system achieved a 93.8% accuracy in terms of recognizing the objects as they really are.
- A virtual environment facilitates the understanding of a scene compared to a series of 2D images like the ones obtained by a camera.
- The geometrical method used to obtain the position, size and orientation of the objects, as detailed in section 5, showed the accuracy presented in Table 2.

Table 2. Result of errors in the 3D reconstruction for the four parameters involved in it after 20 tests

	X_p [cm]	Y_p [cm]	L [cm]	θ_p [°]
e_j	1.9324	1.14289	0.9323	15.8329

Referentes

1. A. De la Escalera. *Visión por Computador*. Prentice Hall, 2001.
2. G.Dudek y M.Jenkin. *Computational Principles of Mobile Robotics*. Cambridge, 2000.
3. O.Faugeras *Three-Dimensional Computer Vision: A Geometric Viewpoint*. The MIT Press, 1993.
4. R.González y R.Woods. *Tratamiento Digital de Imágenes*. Addison-Wesley/Díaz de Santos, 1996.
5. R.Klette, A.Koschan y K.Schlüns. *Computer Vision: Three-Dimensional Data from Images*. Springer, 1998.
6. P.Laplante y A.Stoyenko. *Real-Time Imaging*. IEEE Press, 1996.
7. D.Maravall y Gómez-Allenda. *Reconocimiento de Formas y Visión Artificial*. Addison-Wesley Iberoamericana, 1994.
8. S.Marchand-Maillet y Y.M.Sharaiha. *Binary Digital Image Processing*. Academic Press, 2000.
9. J.R. Parker. *Practical Computer Vision Using C*. Wiley, 1994.
10. S.Russell y P.Norvig. *Inteligencia Artificial: Un Enfoque Moderno*. Prentice Hall, 1996.
11. F.Ongay. *Máthema: El Arte del Conocimiento*. Fondo de Cultura Económica, 2000.