

Algoritmo para la generación y ejecución de trayectoria de un objeto electromecánico móvil utilizando visión por computadora

Algorithm for the generation and trajectory implementation of an electro-mechanical moving object using computer vision

Adriana Ivett de la Roca Chiapas

Instituto Tecnológico de Zacatepec, México

delarocac@itzacatepec.edu.mx

Leticia Santa Olalla Ocampo

Instituto Tecnológico de Zacatepec, México

Leticia_soo@yahoo.com.mx

Roberto Ortiz Delgadillo

Instituto Tecnológico de Zacatepec, México

r_ortiz_d@hotmail.com

Resumen

En el presente trabajo se desarrolló un algoritmo computacional en Matlab para un objeto móvil con la finalidad de obtener la exploración visual dentro de un escenario, evadiendo obstáculos dentro de un área definida y utilizando técnicas de visión por computadora. El sistema computacional adquirió una imagen en línea, donde al mismo tiempo se realizaron σ exploraciones de m muestras dentro del escenario, lo que permitió identificar y generar la orientación del objeto móvil hacia la meta. El algoritmo se ejecuta n veces hasta que el móvil llegue a la meta.

Palabras clave: visión por computadora, algoritmo, Matlab, segmentación.

Abstract

In the present work was developed a computational algorithm in Matlab for a moving object with the purpose of obtaining visual scanning within a scenario, evading obstacles within a defined area and using computer vision techniques. The computational system acquired an image online, where at the same time were scans of samples in the scenario, which helped to identify and generate the orientation of the moving object towards the goal. The algorithm is executed n times until the mobile to reach the goal.

Key words: computer vision, algorithm, Matlab, segmentation.

Fecha recepción: Enero 2015

Fecha aceptación: Mayo 2015

Introduction

There is a growing interest in the automation of processes based on the visual perception of the real world, which requires the acquisition and processing of images. This is related to the area of interest, computer vision. This article presents a computational algorithm that uses the techniques of trajectory tracking and recognition of colors to get a moving object to the goal.

Color

An interesting option is to use color features for visual tracking, due to its simplicity and robustness to partial occlusions and changes of scale.

However, there are several problems to be discussed, notably the constancy of color¹ as they affect you shadows, changes in lighting and camera features. In this project different colors are used to identify the area of movement, the goal, obstacles and moving object. A robust and effective color objects presentation is needed for processing is accurate and fast. The color model plays an important role in the computational precision and robustness of the system. Some of the color models are RGB, YUV, HSL, using RGB for this project.

Trayectory

The ability of a mobile robot in the determination of its location in space is an essential task to navigate completely autonomously.² The knowledge of one's position, as well as other objects or features of interest in the surroundings of the robot, are the basic foundations that support the operations of higher level navigation. For example, the location which makes it possible to plan paths to reach a particular destination while avoiding obstacles, and is fundamental to more complex tasks like building maps of unknown environments. Without the localization capacity, the robots would be doomed to interact with the environment through reactive behaviors, and would be unable to plan actions beyond its field of local perception.³ Deviation function or trajectory function is used to evaluate the properties of the movement in a sequence. Its entrance is a journey and the return value must be inversely proportional to the softness of the trajectory (or directly proportional to the degree of deviation from the road).⁴

Development

The element involved in the work setting is a digital camera, while the communication between the computer and the camera is done via the USB port on the computer; the black background is the work area in which the moving object can move, and the goal is green. The moving object in front of blue color, red obstacles and interface for mobile, allowing communication between the computer and mobile object.

Interfaz PC-Objeto Móvil

In the block diagram of the interface used for this research, that is, between the computer and mobile object (Figure 1) shows that the interface is built for 4 cups, 4 sensors darkness, 2 H-bridges, two engines and the moving object. Is discussed in more detail each of these elements, stressing that as each of these are repeated, only the operation and design of 1 suction cup, 1 sensor, 1 H bridge and a motor will be explained, with the same explanation for other elements of the same type.

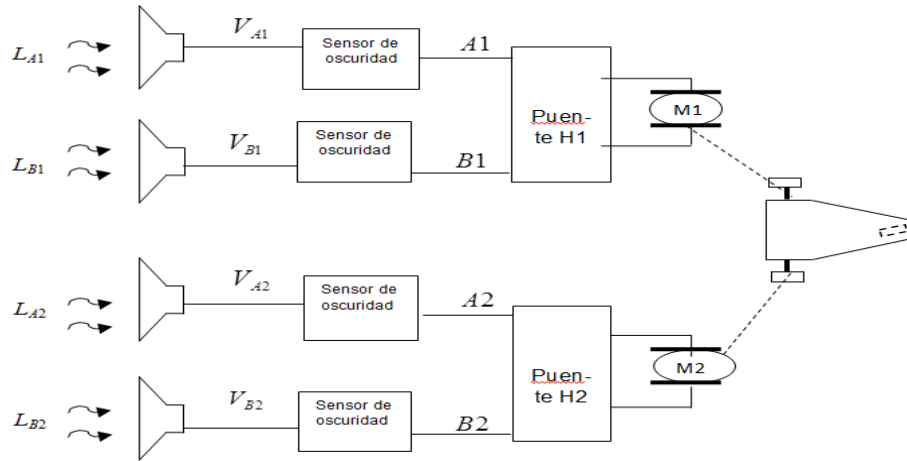


Figure 1. Block diagram between PC and mobile object

Figura 1. $L_{A1}, L_{B1}, L_{A2}, L_{B2}$ light signals are sent by the computer monitor and received by the sensors of darkness through electrical signals $V_{A1}, V_{B1}, V_{A2}, V_{B2}$, where each sensor sends an electric signal to the Bridge respective H. If the L input is white, the electrical signal is 0 and if L It is black, the electrical signal is +5 volts.

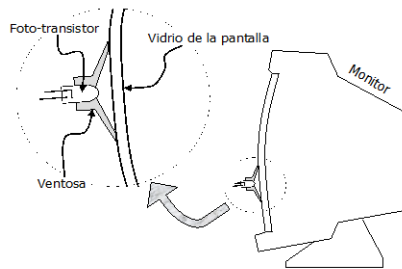


Figure 2. Interface to the monitor, the monitor light illuminates the photoresist

Electrical interface Output Used for Machine Vision Systems Based on MATLAB where you can manipulate the images delivered by the monitor, which ultimately is issued and displayed by the user light. Only need to take this light to build electrical signals, as shown in Figure 2.

Figure 3 shows the image acquired with controlled light conditions, proper placement of the camera and looks ideal material.

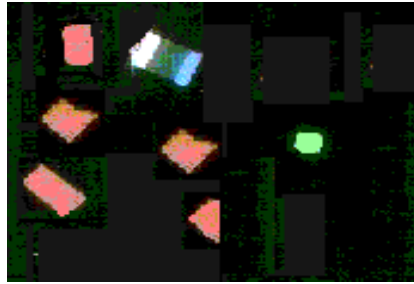


Figure 3 Acquisition of Image "A"

I preprocessing

Figure 4 shows the steps to be followed to improve the acquired image.

1. double instruction to the image acquired "A", so that your data take values of 4 bytes long and thus can be applied any operation, $A = \text{double}(A)$ applies

2. The picture "A" has a three-dimensional matrix that contains information of red, green and blue, which are like is formed the image "A" initially.

$$A_{color} \in \{0,1,\dots,255\}^{n \times m \times 3} \quad (1) \quad color = \begin{cases} rojo \\ verde \\ azul \end{cases}$$

Subsequently a second image "B" in gray levels, applying each image matrix (red, green, blue) weighing as perceived by the human eye color is generated:

$$B = \text{round}(0.299A^{rojo} + 0.586 A^{verde} + 0.114 A^{azul}) \quad (2)$$

Where

$B_{ij} \in \{0,1,2,\dots,255\}$ is the pixel at row column of image gray levels. The image of Figure 4 has its dimensions $i=1,2,\dots,177$ y $j=1,2,\dots,230$.

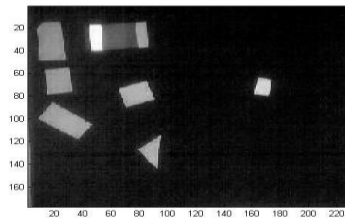


Figure 4. Figure A (see Figure 3) gray levels B

3. A third variable called map, which contain values of zero (0) to one (1) in increments of $1/255$, color is between black and white (Table 1) is generated.

mapa =

0	0	0
1/255	1/255	1/255
2/255	2/255	2/255
.	.	.
.	.	.
254/255	254/255	254/255
1	1	1

Tabla 1. Mapa

This change color map is to adapt the visual interface to gray levels, otherwise deliver a map interface by default, which is not in shades of gray.

To this point only it has made a transformation corresponding to the gray levels of the original image A, to perform the next step is image segmentation B.

Stage Segmentation

Segmentation is the division or partition of the image acquired on objects that compose it and the background. This research is required to identify the moving object, the obstacles, the target and the background.

The segmentation in this case, also refers to the concept of thresholding. To identify the thresholds used, it leaned analyzing histograms and various types of paper of different colors and fabrics are analyzed, helping to determine the most appropriate colors for segmented.

The equations used for thresholding were:

SUBMATRIZ ANALIZADA	ECUACIÓN
Obstáculos	$obstaculo_{ij} = \begin{cases} 255, si(A_{ij}^{rojo} \geq 185) \& (NO(móvil_{ij})) \& (NO(meta_{ij})) \\ 0, en cualquier otro caso \end{cases}$
Meta	$meta_{ij} = \begin{cases} 255, si(A_{ij}^{verde} \geq 212) \& (NO(móvil_{ij})) \\ 0, en cualquier otro caso \end{cases}$
Frente del móvil	$nariz_{ij} = \begin{cases} 255, si(A_{ij}^{azul} = 220) \& (NO(móvil_{ij})) \\ 0, en cualquier otro caso \end{cases}$
Móvil	$móvil_{ij} = \begin{cases} 255, si B_{ij} > 235 \\ 0, en cualquier otro caso \end{cases}$

Table 2. Ranges for thresholding stage

The matrix obstacles have all the elements of the "A" image that are light red in the range greater than 185; goal in the matrix have all the elements of the image "A" that are light green in the range greater than 212 in the matrix nose have all the elements of the image "A" light blue in the highest rank 220, and in Table 2 have all those elements of the image B that are light in white range greater than 235.

Stage Representation and Description

Thus the thresholds for each element used in the stage where they have identified each of the components involved are obtained. To calculate the centers of area that it has:

$$(bx,by)= centro(nariz)$$

Where :

$$bx = \frac{\sum_{i=1}^{177} \sum_{j=1}^{230} j * nariz_{ij}}{\sum_{i=1}^{177} \sum_{j=1}^{230} nariz_{ij}} \qquad by = \frac{\sum_{i=1}^{177} \sum_{j=1}^{230} i * nariz_{ij}}{\sum_{i=1}^{177} \sum_{j=1}^{230} nariz_{ij}}$$

And so for: $(cx,cy) = centro(móvil)$, $(wx,wy) = centro(cola)$ y $(Mox,Moy) = centro(meta)$

For this case, the dimensions of the image (Figure 6) go $i=1,2,\dots,177$ y $j=1,2,\dots,230$. The values obtained for the nose or front of the phone after the calculation are: $(bx=82, by=26)$; to the center

of the moving object ($cx=66, cy=27$); center for mobile tail ($wx=49, wy=28$) and the center of the target ($Mox=161, Moy=70$).

Trajectory calculation to the Goal

Calculating path to the goal is a joint operations center area of mobile (front, middle and tail) and the goal; In addition to calculating the angles involved between the moving object and goal. Figure 5 shows the analysis, where is observed:

(cx,cy) It is the center of the movable within the workspace.

(bx,by) It is the center of the nose moving within the workspace.

(wx,wy) It is the center of the tail moving within the workspace.

(Mox,Moy) It is the center of the goal within the workspace.

σ The angle of the mobile relative to the horizontal.

μ It is the angle of orientation of the target with respect to Mobile.

α It is the angle of the direction of the target with respect to the horizontal.

$\sigma + \mu + \alpha$ It is equal to an angle of 90° .

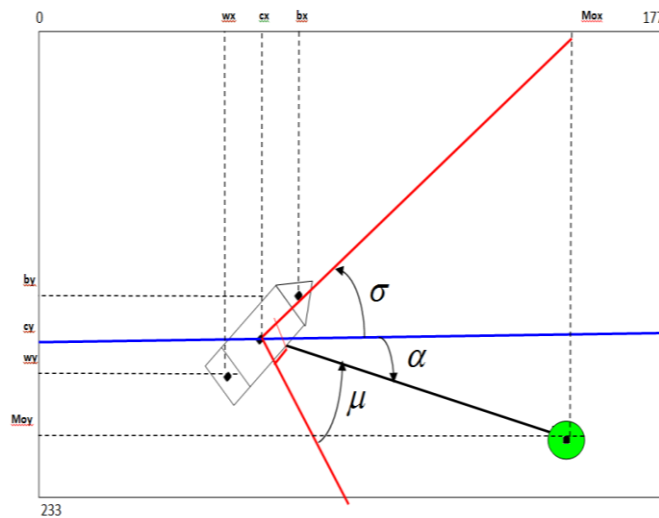


Figure 5. Calculation of the path toward the goal

To perform the calculation of σ, α y μ equations are required $\sigma = \arctan 2(wy - by, bx - wx)$, $\alpha = \arctan 2(Moy - cy, Mox - cx)$ y $\mu = \pi/2 - \alpha - \sigma$

After obtaining μ and must not be obstacles which direction the moving toward the goal, then the steps of the algorithm (Figure 6) are followed.

Results and Conclusion

Having supplemented the implementation of computational algorithm developed in Matlab, the result met the objectives. Explorations of the moving object is identified and the path was obtained executing the moving object towards the goal. They identified by technical obstacles color objects to be avoided by the moving object. The algorithm is able to work with other image format. Throughout the development of the project improvements or alternatives to the implemented solution is raised, for example, concentric scans, as would a radar, in addition to assessing how much time is needed for segmentation. Given the natural smooth movements of the images, it is not necessary to recalculate this value for each frame because there would be significant changes from one image to the next.

Matlab interface with differential robot, ie electromechanical mobile object, is of simple construction making it easy to reproduce. In addition, it is economical compared to commercial.

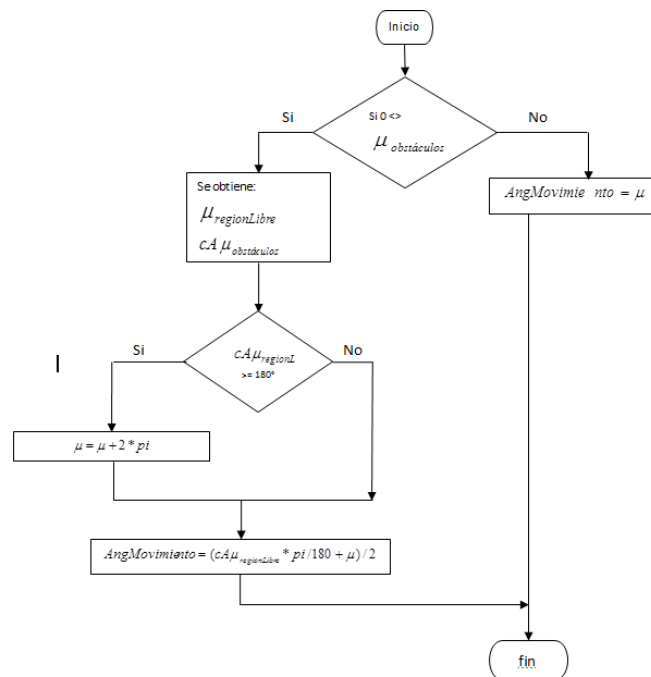


Figure 6. Flowchart of the algorithm for the generation and execution of the trajectory

Bibliography

Finlayson G. D. (2000) Computational color Constancy, in International Conference on Pattern Recognition, vol. 1, Barcelona, Spain, pp. 191-196.

Bailey, T. (2002). Mobile Robot Localisation and Mapping in Extensive Outdoor Environments. Tesis Doctoral, Australian Centre for Field Robotics, Department of Aerospace, Mechanical and Mechatronic Engineering, The University of Sydney (citado en pp. 9, 10, 128 y 129).

Fernández, C. (2012). Técnicas de Navegación para un Robot Móvil utilizando sistemas de razonamiento espacial. Salamanca, España, p. 9.

María Carina Roldán, Marcelo Naiouf, Armando de Giusti. Paralelización de Algoritmos de Seguidores de trayectorias laboratorio de investigación y desarrollo en Informática. Facultad de informática, UNLP.