

Vision System for the Navigation of a Mobile Robot

Griselda Saldaña González¹, Jorge Cerezo Sánchez¹, Mario Mauricio Bustillo Díaz², Apolonio Ata Pérez²

¹ Universidad Tecnológica de Puebla,
Ingeniería en Tecnologías para la automatización, Puebla,
Mexico

² Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación, Puebla,
Mexico

{griselda.saldana, jorge.cerezo}@utpuebla.edu.mx, bustillo1956@hotmail.com, apolonio.ata@gmail.com

Abstract. In this paper the development of an object detection system in a controlled two-dimensional space using computer vision techniques is presented. The detected objects have a rigid geometry and are exposed to real light; therefore, the system is robust to changes in lighting and shading. In order to handle the large amount of data to be processed in real time, a MyRIO device which contains an FPGA is used. This device allows communication with the LabVIEW software where the user interface resides. Using LabVIEW a tracking by color algorithm is implemented, in order to attend to a reactive agent, which uses an infrared sensor to detect the distance to an obstacle and perform the functions of foraging and storage. In order to improve performance, a supervisory system was implemented using a Kinect device that provides information relative to the position of the objects in the test area. This information allows eliminating occlusion problems.

Keywords. Tracking, LabVIEW, color detection, mobile robot.

1 Introduction

Currently, vision plays a central part in the robotic area for a series of different tasks such as, self-location, navigation, object recognition and manipulation, object tracking, social interaction between human and robot, imitation, among others [10]. Over the last years mobile robots have participated in more and more complex tasks which often require the collaboration among individuals who, generally, differ in relation to the abilities they possess and the way they perceive the

environment [12]. In order to carry out those tasks, mobile robots tend to be equipped with powerful vision systems in which real time operation becomes a challenge. Furthermore, when a system of multiple robots is considered, control and coordination are complex and demanding tasks. According to [16], vision systems for mobile agents, can be classified in six types, sensing and perception [31, 23, 9, 17, 7], mapping and self-localization [22, 2, 4], recognition and location [5, 28, 29, 30, 14], navigation and planning [13, 20], tracking [21, 19, 1] and visual servoing [25, 11, 27].

In this work a vision system mounted in a mobile robot, which is able to navigate in a controlled environment by using a camera, is developed. The system is inspired in the task division phenomena, which is carried out in some insect colonies for making complex tasks common for a group, such as the forage [8]. The robot task consists on observing its surroundings in order to identify an object of interest of a specific color, then navigate towards it and grab the object in order to transport it to a storage area. The agent controls a camera in an independent way, so that it can focus on different objects around it.

On the other side, the robot has its own computing resources which allow it to analyze the image by using a tracking algorithm based on color. To process the images in real time a MyRIO 1900 board is used; this board includes an FPGA which allows it to perform a parallel processing.

The robot also has the necessary resources for communicating with other agents by using a Wi-Fi network. The system was programmed using LabVIEW, a development environment for measurement and automation, developed by National Instruments (NI) [18, 6], which has a group of vision tools [24] and a set of graphic administrators and users interfaces.

The document is organized as follows. First, section 2 grants a brief introduction to agents concept, in section 3 a description of the system is presented, emphasizing the agent construction as well as its programming in LabVIEW, in section 4, the tests performed to the system are shown altogether with some of their results. Finally in section 5 conclusions and future work are presented.

2 Agents

Intelligent agents have been used in many scenarios, one of them is robotics. An agent, in hardware or software is capable of performing intelligent tasks autonomously and independently therefore it should own the attributes of being autonomous, perceive and act on its environment, furthermore it should have goals and motivation and being able to learn from experience and to adapt to new situations. Reactive agents represent a special category of agents that have no representation of the environment, but respond to it using a stimulus-response mechanism. The different characteristics of the environment where the agent is found are perceived as different stimuli that allow the elaboration of a direct response without needing an internal representation of such an environment. Its implementation is based on the philosophy "the best representation of the environment is the environment itself" [3].

The three main characteristics of reactive agents are [15]:

- *Emergent functionality*: reactive agents are simple; they interact with other agents through basic mechanisms. The complex behavior model emerges when all interactions between agents are considered in a global way. It does

not exist a predetermined plan or specification for the behavior of reactive agents.

- *Task decomposition*: a reactive agent can be seen as a collection of modules acting independently taking responsibility for specific tasks. Communication between modules is minimal and usually at low level. There is no global model within the agent but the global behavior has to emerge.
- Reactive agents tend to be used at levels close to the physical level corresponding to the environment sensors.

Due to reactive agents respond directly to the sensor-stimuli, they present response times lower than those obtained by agents with prior processing and planning. The reactive agents are suitable for implementing the components of the reactive level in architectures for the control of mobile robots.

3 System Description

The system configuration is formed by a mobile robot type VEX Clawbot [3], which has a MyRIO board in its structure. This board communicates wirelessly with a central computer, in which the user interface resides. A high definition camera is connected to this board and located on a servomotor that allows it to move horizontally (panning).

The robot also has two servomotors for its transportation and one more for the gripper opening and closing movements. In the frontal part, there is an infrared sensor which allows the robot to determine the distance to the object of interest. Figure 1, shows the build prototype.

The MyRIO board has a wireless router that performs the Wi-Fi communication using global variables mapped to the network which allow the use of the available information. Furthermore, this board allows STAND-ALONE system operation, ever since the program developed runs within its own modules.

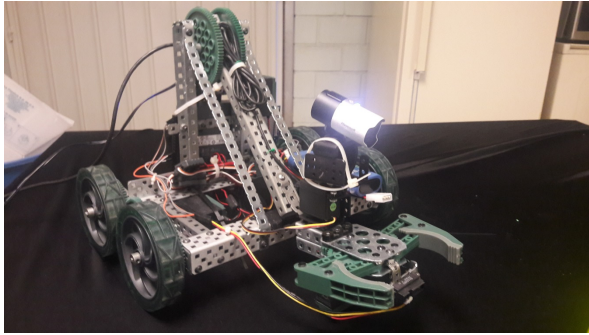


Fig. 1. Mobile Robot prototype

3.1 Color Based Tracking

To carry out the tracking of the detected object, the images obtained are converted to the HSI format and the scale for the filtering of the specified green color is determined. A section of the tracking algorithm developed in LabVIEW is shown in Figure 2.

Within the image limits, the vision system distinguishes the colors range that do not belong to the programmed filter (the background is eliminated) and only allows to see the specific color elements. Figure 3, shows two balls detected and their differentiation of the rest of the image.

3.2 User Graphic Interface

The interface that was developed is highly useful and practical. It has elements that allow the parameters configuration for the vision machine, to adjust the pulse width for the robot servomotors, it allows to modify the execution time, to select other colors' search and to visualize in real time the acquisition of the image obtained by the camera and the processed image. It is possible to observe the number of detected objects, as well as the coordinates (x, y), of their centers of mass.

It also has two indicators of the agent activity, one for the moment when the robot is aligned with the object of interest (in range) and another one for the moment when the gripper can take the object (pre-adjustment). Figure 4, shows the complete user interface.

3.3 Reactive Agent Design

A state machine was designed to allow the robot to interact with the environment. This machine is inspired in a reactive agent [26], who solves a local problem in a testing area of specific measurements, 1.68 x 2.24 mt., that is located in an environment with natural light; considering a vision range for the camera from 0 to 180 degrees. The system is designed in a modular way in order to add new actions, behavior patterns or additional agents. During the perception, the camera carries out a panning process which is interrupted when a determined object appears in the screen and when its center is aligned with the X axis of the camera.

Once the center of the object is located, the strategy programmed for the agent consists in determining in which of the five possible areas of vision in which the image was divided the object can be found, Figure 6. Afterward, the agent defines the control actions for the mobile to align the camera with the object to grab.

Once the camera located in the robot is in the central area (region 3), a slight adjustment movement is performed in order to center the object with the gripper and to determine how far the mobile is from it, by using the infrared sensor to finally move on and grab it.

It is important to emphasize that if the mobile and the ball misalign because of the effects of the locomotion mechanics of the robot or the movement of the object of interest, the perception system of the agent allows to correct the position of the robot in execution time and to carry out a new panning and aligning actions according to the changes in the environment. Figure 5, shows the implementation of the agent in a state machine developed in labVIEW.

This behavior, carried out by the agent, corresponds to the forage function. Once the object of interest is grabbed, the robot returns to the original position, where it collects the objects for another agent to carry out the storage later.

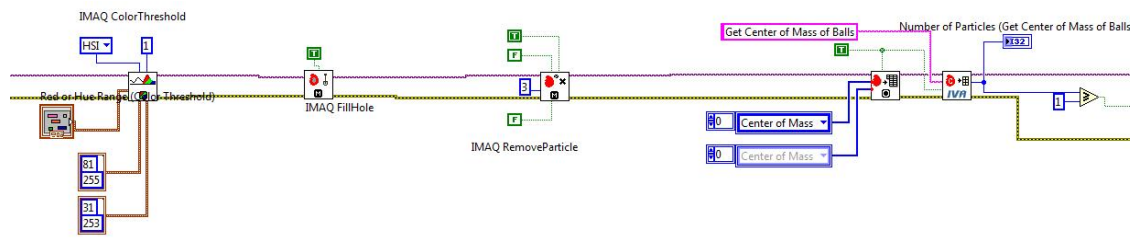


Fig. 2. Image processing for the color tracking

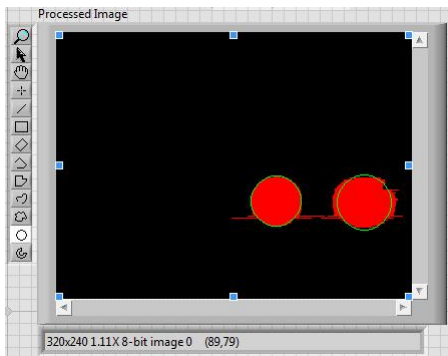


Fig. 3. Green objects tracking, the second that is moving

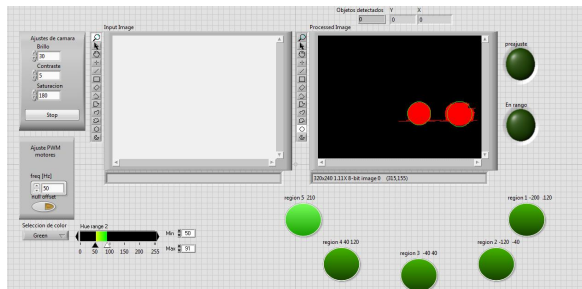


Fig. 4. User interface

4 Tests and Results

Multiple experiments were carried out considering the space of the testing area, Figure 7.

The robot and a green ball were located in the area and it was verified that the robot could identify the ball, align to it, grab it and leave it in a specific position. The system was tested under artificial light and it was proved that the robot's behavior is robust to different noise conditions and lighting and shadowing changes. An important step for

the correct functioning of the system was color calibration since, depending on the weight given to the saturation and the tonality (HSI), discrimination can or cannot improve.

The value of brightness makes the system more or less immune to the light changes of the environment. Previous to the implementation of this algorithm, Traditional mean shift and Shape adapted mean were tested, in both cases it was attempted to detect 4 green balls taking as parameters Brightness: 30, Contrast: 5. Saturation: 180 and the results obtained are shown in Table 1.

For each algorithm 50 tests were carried out, the tests were done under controlled light conditions and with different distances of the robot, obtaining a 97% of detection for the proposed algorithm. 50 tests for each algorithm were also carried out for the cases with a variability of light, and in these cases 93% of the detection with the proposed algorithm was obtained, which surpasses the ones tested before, for which the percentage varies between 86% and 90% of detection. To make the robot align with the object of interest, in order to grab it, it was decided to split the field of vision of the panning camera in 5 zones, and then a fine adjustment of the gripper was made.

This was a result of multiple movement tests of the robot that were made in order to make it react to changes in the environment. A fine adjustment to the gripper was made in order to make the robot grab the object of interest. Because of this, it was necessary to regulate the advance speed of the robot by manipulating the width of the pulse for the motors. The rangefinder (infrared), was calibrated with the same purpose, to adjust it to a measurement in units of the real world. Even

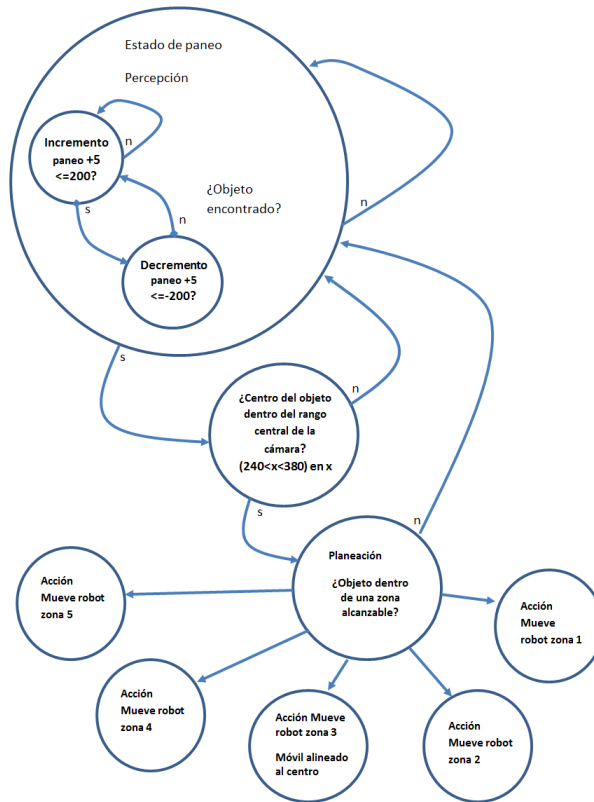


Fig. 5. Design of a reactive agent for the vision systems

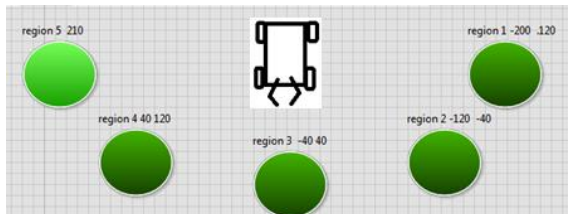


Fig. 6. Regions of the field of view for the robot, in degrees



Fig. 7. Testing area

though the system is robust, there is no solution for the case when objects' occlusion exists.

Each ball within the vision system receives a tag. If two balls located in the robot vision area intersect, the tags may invert. If this happens, the robot will still be able to grab any of the balls and for its expected behavior it is not relevant which of the balls is taken first. On the other side, if two balls

are located next to each other, the vision system can recognize them both as one object, which does not affect the aligning of the robot but it does affect when grabbing the object. If any object is located behind the robot, the vision system would indefinitely continue in the panning state.

Table 1. Objects detected by the algorithm

Algorithm	With controlled light	Under variable light conditions
Traditional mean shift	3 to 4 balls	0 to 3 balls
Shape adapted mean	3 to 4 balls	0 to 3 balls
Based in color calibration	4 balls	4 balls

4.1 Supervisory System

In order to solve the occlusion problem presented with the vision system, a kinect vision sensor was placed in parallel to the test area at 2.8 meters high to provide information about the position of the objects of interest related to the robot.

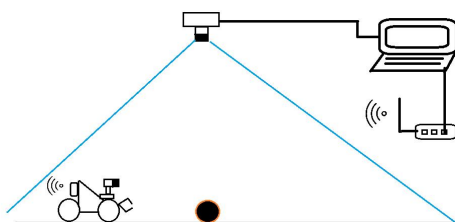
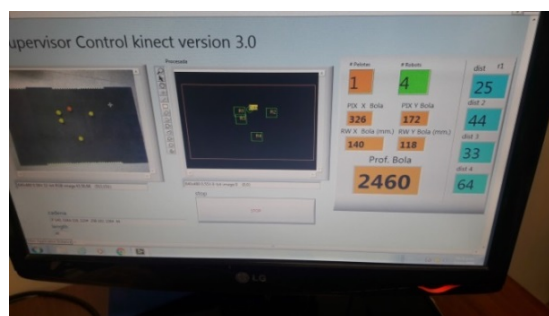
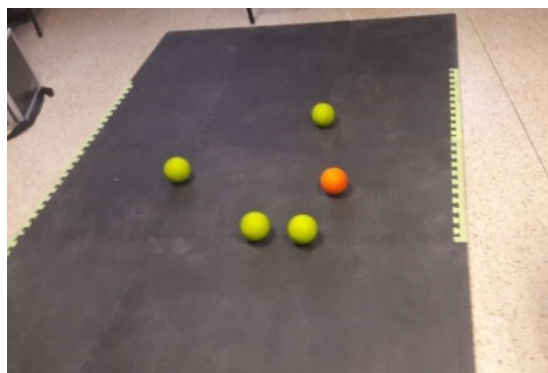
**Fig. 8.** Architecture of the supervisory system

Figure 8, shows the architecture of the supervisory system implemented.

The Figure 9, shows a case where the indicators in orange show the position in x and y of the ball in millimeters, from the origin of the plane. Additionally the green indicator shows the number of robots detected. The system also shows the depth of the objects in millimeters, this information will be used in the future in order to perform more complex actions.

A communication system was developed by creating a local network between a computer containing the object detection algorithm and the MyRIO unit via Wi-Fi. The information of the detected elements is structured in a string of 40 characters in length, so it can be shared through a global variable created in LabVIEW. The content of this string is structured as shown in Figure 10. The data is: "P: Ball, X real world, Y realworld, A:robot1 X real world, Y real world #distance to ball..."

This information available on the network will be used in the future to communicate a group of mobile robots in the testing area, so they can

**(a)** User interface showing detection of multiple objects at runtime**(b)** Objects disposition in the test area**(c)** Real world measurement**Fig. 9.** User interface with measurements of real world

coordinate their actions in order to grab objects and accomplish the foraging task.

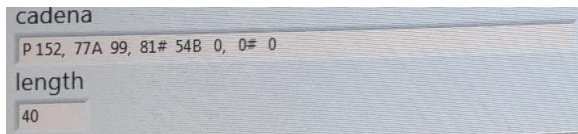


Fig. 10. Character string sent using a global variable

5 Conclusions and Future Work

The possibility of using a robot to perform a task common to a group, such as forage, is demonstrated in this work. The use of LabVIEW allowed to reduce the implementation time and to carry out the analysis of the vision system and the agent in real time.

The implemented agent executes the control actions of the robot and is able to perform corrections during the execution time. It presents the advantage of allowing the development of actions independently and to incorporate new behaviors quickly and easily.

The implemented system is modular and configurable. Finally, in contrast to other similar works, this system has the advantage of allowing communication between agents or with other devices through the use of Wi-Fi, thanks to the use of the MyRIO device.

A second robot in charge of the storage function is proposed as future work, altogether with communication tests between the agents. On the other side, an algorithm of obstacles evasion will be implemented. This will allow the robot to carry out more complex tasks common to the agents group.

Finally, the inclusion of a Kinect device has eliminated the occlusion problem; this has provide a superior view of the testing area in order to identify all the time the position of the objects of interest, even when these objects are located behind the robot.

Acknowledgements

The authors thank PRODEP for the support for the execution of this work.

References

1. **Basso, F., Munaro, M., Michieletto, S., Pagello, E., & Menegatti, E. (2013).** Fast and robust multi-people tracking from RGB-D data for a mobile robot. In **Lee, S., Cho, H., Yoon, K.-J., & Lee, J.**, editors, *Intelligent Autonomous Systems*, volume 1, chapter 12. Springer, Berlin, Heidelberg, pp. 265–276.
2. **Biswas, J. & Veloso, M. M. (2012).** Depth camera based indoor mobile robot localization and navigation. *2012 IEEE International Conference on Robotics and Automation*, pp. 1697–1702.
3. **Brooks, R. A. (1986).** A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*, Vol. 2, No. 1, pp. 14–23.
4. **Çelik, K. & Somani, A. K. (2013).** Monocular vision slam for indoor aerial vehicles. *Journal of Electrical and Computer Engineering*, Vol. 2013, pp. 15.
5. **Cocias, T. T., Moldoveanu, F., & Grigorescu, S. M. (2013).** Generic fitted shapes (gfs): Volumetric object segmentation in service robotics. *Robotics and Autonomous Systems*, Vol. 61, No. 9, pp. 960–972.
6. **De Asmundis, R. (2011).** The importance of a deep knowledge of labview environment and techniques in order to develop effective applications. In **Silviu, F.**, editor, *Practical Applications and Solutions Using LabVIEW™*, chapter 21. InTech, pp. 1–37.
7. **Di-Paola, D., Naso, D., Milella, A., Cicirelli, G., & Distante, A. (2010).** Multi-sensor surveillance of indoor environments by an autonomous mobile robot. *International Journal of Intelligent Systems Technologies and Applications*, Vol. 8, No. 1-4, pp. 18–35.
8. **Dorigo, M., Birattari, M., & Stutzle, T. (2006).** Ant colony optimization: Artificial ants as a computational intelligence technique. *IEEE Computational Intelligence Magazine*, Vol. 1, No. 4, pp. 28–39.
9. **Fernández-Caballero, A., Castillo, J. C., Martínez-Cantos, J., & Martínez-Tomás, R. (2010).** Optical flow or image subtraction in human detection from infrared camera on mobile robot. *Robotics and Autonomous Systems*, Vol. 58, No. 12, pp. 1273–1281.
10. **Fiack, L., Cuperlier, N., & Miramond, B. (2015).** Embedded and real-time architecture for bio-inspired vision-based robot navigation. *Journal of Real-Time Image Process*, Vol. 10, No. 4, pp. 699–722.

11. **Fomena, R. T., Yoon, H. U., Cherubini, A., Chaumette, F., & Hutchinson, S. (2009).** Coarsely calibrated visual servoing of a mobile robot using a catadioptric vision system. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, IEEE, St. Louis, MO, USA, pp. 5432–5437.
12. **Infantino, I., Cossentino, M., & Chella, A. (2002).** An agent based multilevel architecture for robotics vision systems. *Proceedings of the International Conference on Artificial Intelligence, IC-AI'02*, pp. 386–390.
13. **Kang, T.-K., Choi, I.-H., Park, G.-T., & Lim, M.-T. (2013).** Local environment recognition system using modified surf-based 3d panoramic environment map for obstacle avoidance of a humanoid robot. *International Journal of Advanced Robotic Systems*, Vol. 10, No. 6, pp. 1–13.
14. **Kwasue, K. & Komatsu, T. (2013).** Shape measurement of a sewer pipe using a mobile robot with computer vision. *International Journal of Advanced Robotic Systems*, Vol. 10, No. 1, pp. 52.
15. **Maes, P. (1991).** *Designing Autonomous Agents: Theory and Practice from Biology to Engineering and Back*. MIT Press.
16. **Martinez-Gomez, J., Fernandez-Caballero, A., Garcia-Varea, I., Rodriguez, L., & Romero-Gonzalez, C. (2014).** A taxonomy of vision systems for ground mobile robots. *International Journal of Advanced Robotic Systems*, Vol. 11, No. 7, pp. 111.
17. **Martins Ekmanis, A. N. (2016).** Mobile robot camera extrinsic parameters auto calibration by spiral motion. *Engineering For Rural Developmen*, volume 15, Jelgava, pp. 558–565.
18. **Moreno-Astorga, A., Moreno-Salinas, D., Chaos-García, D., & Aranda-Almansa, J. (2011).** Simulation benchmark for autonomous marine vehicles in labview. *IEEE OCEANS*, IEEE, Santander, Spain, pp. 1–6.
19. **Ostafew, C. J., Schoellig, A. P., & Barfoot, T. D. (2014).** Learning-based nonlinear model predictive control to improve vision-based mobile robot path-tracking in challenging outdoor environments. *International Conference on Robotics and Automation*, IEEE, Hong Kong, China, pp. 4029–4036.
20. **Papadakis, P. (2013).** Terrain traversability analysis methods for unmanned ground vehicles: A survey. *Engineering Applications of Artificial Intelligence*, Vol. 26, No. 4, pp. 1373–1385.
21. **Park, J., Hwang, W., Kwon, H., Kim, K., & Cho, D. (2013).** A novel line of sight control system for a robot vision tracking system, using vision feedback and motion-disturbance feedforward compensation. *Robotica*, Vol. 31, No. 1, pp. 99–112.
22. **Parvez, A., Zabir, A., Mohammad, I. J. R., Awal, M. A., & Choudhury, S. M. (2014).** Self-localization of a mobile robot using monocular vision of a chessboard pattern. *International Conference on Electrical and Computer Engineering (ICECE)*, IEEE, Dhaka Bangladesh, pp. 753–756.
23. **Peynot, T., Scheduling, S., & Terho, S. (2010).** The marulan data sets: Multi-sensor perception in a natural environment with challenging conditions. *The International Journal of Robotics Research*, Vol. 29, No. 13, pp. 1602–1607.
24. **Relf, C. G. (2004).** *Image Acquisition and Processing with LabVIEW*. CRC Press.
25. **Siradjuddin, I., Behera, L., McGinnity, T. M., & Coleman, S. (2013).** Image-based visual servoing of a 7-dof robot manipulator using an adaptive distributed fuzzy pd controller. *IEEE/ASME Transactions on Mechatronics*, Vol. 19, No. 2, pp. 512–523.
26. **VEX Robotics (2017).** <http://www.vexrobotics.com.mx/>.
27. **Wang, K., Liu, Y., & and, L. L. (2014).** Visual servoing trajectory tracking of nonholonomic mobile robots without direct position measurement. *IEEE Transactions on Robotics*, Vol. 30, No. 4, pp. 1026–1035.
28. **Xiao, J., Adler, B., Zhang, J., & Zhang, H. (2013).** Planar segment based three-dimensional point cloud registration in outdoor environments. *Journal of Field Robotics*, Vol. 30, No. 4, pp. 552–582.
29. **Yeom, S. & Woo, Y.-H. (2013).** Person-specific face detection in a scene with optimum composite filtering and colour-shape information. *International Journal of Advanced Robotic Systems*, Vol. 10, No. 1, pp. 70.
30. **Zhang, J., Zhang, J., & Chen, S. (2013).** Discover novel visual categories from dynamic hierarchies using multimodal attributes. *IEEE Transactions on Industrial Informatics*, Vol. 9, No. 3, pp. 1688–1696.
31. **Zhao, Z. & Weng, Y. (2013).** A flexible method combining camera calibration and hand—eye calibration. *Robotica*, Vol. 31, No. 5, pp. 747–756.

Article received on 08/08/2016; accepted on 18/10/2016.
Corresponding author is Griselda Saldaña González.