X Congresso Brasileiro de Inteligência Computacional (CBIC'2011), 8 a 11 de Novembro de 2011, Fortaleza, Ceará
© Sociedade Brasileira de Inteligência Computacional (SBIC)

COMBINING ULTRASONIC SIGNALS AND MULTI-COLORED IMAGES TO PERFORM OBJECT TRACKING AND RECOGNITION IN LOW-COST ROBOTIC PLATFORMS

Danilo H. F. Menezes
Departamento de Computacao, Universidade Federal de Sergipe, Sao Cristovao, Brasil
danilohfm@dcomp.ufs.br

Thiago D. Mendonca, Wolney M. Neto
Department of Informatics, Vrije Universiteit Brussels, Brussels, Belgium
{tdantasd,wdemello}@vub.ac.be

Hendrik T. Macedo, Leonardo N. Matos
Departamento de Computacao, Universidade Federal de Sergipe, Sao Cristovao, Brasil
{hendrik,lnmatos}@ufs.br

Abstract – Robotics research focuses on a broad range of interdisciplinary aspects. Regarding robot-driven application development, different purposes and complexities may be considered. State-of-the-art platforms are usually adopted for developing non-trivial tasks, however, their high costs occasionally inhibit robotics application for education and research purposes. Some tasks, such as pattern recognition, are usually designed without considering low-cost requirements. In order to fully explore the capabilities of low-cost platforms, this article presents an empirical analysis of object tracking and recognition accuracy, non-trivial and essential tasks for Robotics. This task is performed by an autonomous robot equipped with camera and ultrasonic sensor. Three experimental scenarios are defined for further observation and comparison. Object tracking and representation acquirement are achieved in these scenarios only by camera, only by ultrasonic sensor and by combining both, respectively. 10-fold cross validation has been carried on a MLP neural network with different learning rates. Image-based recognition got an average f-measure above 0.9 and an area under ROC curve above 0.95, which proved to be better than ultrasonic-based recognition with a f-measure around 0.8 and area under ROC curve around 0.85. Experiments have also validated low-cost platforms adoption for object tracking and recognition.

Keywords – Low Cost Robotic Platforms, Object tracking, Object Recognition, Multi-layer Perceptron.

1 INTRODUCTION

Path-planning, self-localization, environment exploitation and mapping are typical robot-regarded investigation issues. In particular, recognizing its habitat is crucial to perform well any of such tasks. Object recognition, however, is a complex task and good accuracy level is usually achieved on high-cost robotic platforms, equipped with modern and advanced sensing devices [14] and complex software [7]. Unfortunately, many academic and research institutions around the world do not have enough financial support to be able to provide substantial contributions to the research field.

The use of low-priced platforms and sensors have recently been validated for educational and research purposes. The first has been validated in AI [9], [13], Programming Languages [3], Image Processing [12], and Embedded Systems [**?**] courses. The second can be exemplified by some complex issues such as real-time face detection [8], robot navigation [2], [11] and self-localization [14].

Some of these complex tasks are solved adopting computer vision techniques [2], [8], [11]. This approach considers retrieving data from a video camera to perform different purposed tasks. In addition to such way of acquiring environment representation, it is possible to consider other sensors capabilities, which can be more robust depending on the issue. Concerning robot navigation, the combination of an ultrasonic sensor and a video camera is a better approach [2]. Despite these implemented works, none of them has ever explored the combination of sensors for solving object recognition robotic vision problem. This approach could arise questions such as: whether combining camera and ultrasonic sensor is more efficient than considering just a camera; how could an ultrasonic sensor acquire an object shape representation; how efficient could a low sensor be for object shape recognition; and so forth.

In order to show the capabilities of low cost platforms to solve not simple tasks as well as identifying what sensors best fit each particular issue, this paper proposes an analysis of efficiency for object shape recognition by an autonomous mobile robot based on vision and/or ultrasonic perception. The environment is comprised of the low-cost Lego Mindstorms NXT robot mounted with an off-the-shelf video camera and/or a simple ultrasonic sensor which comes with the NXT platform. Three scenarios have been configured: (i) using only the camera, (ii) using only the ultrasonic sensor and (iii) combining both. The second scenario suggests an original approach to acquire object shape representation. In each of these scenarios it is possible to observe different methods to object tracking and to acquire an object representation. The image processing workflow and neural network classification methods are the same for all scenarios.

In section 2 the state-of-the-art of low-cost platforms adoption is presented. Section 3 describes the robot, its configuration and environment customization. In section 4, the three sensing scenarios are defined. Section 5 describes the algorithms for object tracking, frame adjustment, image processing and classification. Section 6 presents experiments and discusses results. Finally, section 7 concludes on the feasibility of the proposed approach for recognizing objects.

2 LOW-COST ROBOTIC PLATFORM RESEARCH

Inclusion of robotics in education and research allows the conception and the development of a wide diversity of appliances. For educational purpose, robotics is employed at graduation courses. According to [6], "At least seven out of the 14 knowledge areas in the CC2001 Draft contain topics that could be motivated or enhanced through robotics-oriented projects." The author supports the importance of low-cost robots adoption for ACM Computing Curriculum 2001 beginning and advanced courses. Klassner also describes Lego Mindstorms as the best suitable platform to increase college students motivation considering its low cost, flexibility and appearance.

Sustaining Klassner's considerations, [4] proves that Mindstorms platform is sophisticated enough to demonstrate several fundamental concepts taught in standard signals and systems courses. Due to Mindstorms low price, students could perform experiments at home or in classroom.

An alternative of low-cost robotics is presented in [10]. According to them, GoGo Board is an open-source low-cost programmable brick mainly designed for developing countries. The authors use data from studies in several countries such as Brazil, Mexico and Thailand, stating that despite the educational benefits of programmable robots, their use has been limited to well-founded schools and organizations due to their prohibitive cost and limited availability. For this reason, the authors, who also cite NXT system, suggest the GoGo Board alternative.

Apart from educational applications, low-cost platforms can be inserted in more complex research projects. In [14], authors developed some case studies such as Monte Carlo localization method adopting the IRobot Create Roomba low-cost robot, that costs around US$130,00, one of the cheapest low-priced platform.

In addition to its educational use, the Lego Mindstorms NXT set has a great potential for more refined issues. In [11] a video camera is used to remotely control the robot to exit a maze while avoiding obstacles and searching for previously recognized objects. In such work, the authors implemented a non-trivial multi-agent system which envolves fuzzy logic and Simplified Memory-Bounded Algorithm (*SMA**) to take decision, and image processing techniques to better identify an object. In [8], a real-time face recognition system is implemented with Mindstorms NXT and wireless camera.

According to [2], a better approach to control robot's movement can be achieved by combining two low-priced sensors. They propose the combination of a sonar sensor and an off-the-shelf video camera to improve the movements of a Lego Mindstorms RCX kit, older than NXT version. This combination had never been suggested for the object recognition issue. In this work we combine low-cost ultrasonic sensor and video camera for object recognition in such a way that is possible to take some important insights on the different sensors combination concerning object tracking and recognition performance.

3 ROBOT AND ENVIRONMENT DESCRIPTION

3.1 THE ROBOT

An autonomous mobile robot has been customized using the Lego Mindstorms NXT robotics kit. This set is composed of light, sound, touch and ultrasonic sensors. The programable brick control all these sensors and three servo motors, allowing the development of various robotics applications. A brief description of the brick's most important properties is: 32-bit ARM7 microcontroller, 256 Kbytes FLASH/64 Kbytes RAM, Wireless Communications: Bluetooth class II V2.0, USB Port (12Mbit/s) and 6 AA batteries or Rechargeable Lithium battery.

NXT has been mounted as a mobile robot, moving around through two servo motors attached to rubber wheels. The ultrasonic sensor and an off-the-shelf camera had been also attached to the robot.

The ultrasonic sensor, which belongs to NXT set, consists of a transmitter that sends 40KHz sound signals, and a microphone that receives the sound back. The sensor has a range of 0cm to 255cm with an accuracy of +/-3cm. The adopted camera is Logitech QuickCam Express video camera, with resolution of 320x240 pixels. This CMOS video camera is wired connected to the PC via USB port.

Robot behavior configuration is done through firmware LeJOS NXJ, a tiny Java operating system which implements advanced control algorithms programming interface(*API*). The version used in this work is LeJOS NXJ 0.85, which runs Java applications uploaded to the robot. LeJOS NXJ was chosen considering that it provides well-implemented methods to control the motors, to receive sensor data and other functionalities already implemented in its API. It is also possible to establish wireless communication over Bluetooth protocol, supported by Mindstorms NXT set.

3.2 THE ENVIRONMENT

The environment provides different shaped and sized objects, aiming to validate the recognition of a broad variety of objects and shapes. Therefore, the experiment becomes less specific as possible. The experimental controlled environment consists of a 9 squared meters area composed of three-dimensional polygons arranged randomly, enabling the robot to approach from different angles, as Fig. 1 shows.

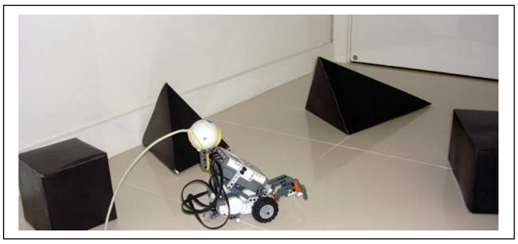


Figure 1: Experimental environment

4 SENSING SCENARIOS

Three different sensing scenarios have been defined through distinct methods. The first scenario proposes a 'blind' sensing that uses not more than ultrasonic sensor potential. In the second, all tasks are performed based on camera input data. Finally, in the third scenario, camera and ultrasonic sensor cooperate to perform the task.

In each scenario, two particular issues arise: object detection in the scene and subsequent robot moving correction to reach it. Besides different sensing approaches, a pre-processing and segmentation pipeline for images acquired by the camera is defined in order to obtain a good representation for subsequent feature extraction.

Next, we describe the scenarios introduced so far in respect to (i) stopping criteria, (ii) method for obtaining representation, and (iii) acquired representation.

4.1 R$\underline{u}$ SCENARIO

- *Stopping Criteria:*

  While performing a random trajectory, the robot stops whenever it reaches a minimum distance of 20cm from the object[1]. This value has been determined taking into account results obtained in empirical tests. After stopping in front of the object, the robot begins to acquire a representation of the object shape.

- *Method for Obtaining Representation:*

  The ultrasonic scanning approach, illustrated in Fig. 2a, consists in capturing just the object outline. As a consequence, robot performs a quick scan given it does not need to measure as many points as its matrix of representation has. Since measurement considers few points, minor accumulated error is observed in motor rotation.

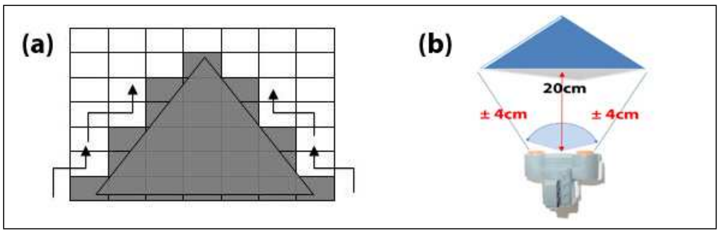


Figure 2: Methods for object scanning(a) and for discriminating object to background(b)

  Scanning starts by measuring the extreme bottom-left point of the object. Afterward, the sensor moves upward until it begins to see background, what is realized through high values from sensor output that exceed the first measure in 20%. In order to find the object once more, the sensor moves horizontally towards the object center. This procedure is repeated from the right to the left side of the object.

  The acceptance margin, given by twenty percentage points of initial distance, considers the variation in sensor reading given by motor rotation during scanning process, as Fig. 2b shows.

- *Acquired Representation*

  Distances measured by ultrasonic sensor are represented in matrix format (Fig. 3a). Such original matrix is then transformed into a matrix consisting of 0's and 1's, where the value 1 represents the object and 0 the background (Fig. 3b).

[1]Limitations on how an object is distinguished from the background in this scenario are discussed in the results section.

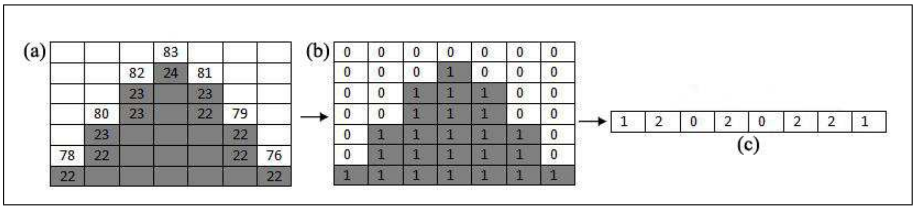


Figure 3: Measurement representation since sensor reading to feature representative vector

The feature vector (Fig. 3c) is obtained from the matrix on the following criteria:

- Attributes 1-6: difference between the number of 1's in line *L* and line *L-1*, from top to bottom;
- Attribute 7: number of 0's in the six extracted features (A squared object should have, theoretically, more 0's than 1's);
- Attribute 8: number of 1's in the six extracted features (A triangular object should have, theoretically, more 1's than 0's).

4.2 Rc SCENARIO

- *Stopping Criteria:*

  Robot's stopping criteria is considered during object tracking. The robot stops moving when a positive pattern is detected into the sight of the camera and whether more than 5 positive patterns are detected simultaneously. This number has been observed empirically when the robot is near to the object, as a peculiarity of the classifier behavior. The positive sample is recognized by a classifier to be described in the object tracking section.

- *Method for Obtaining Representation:*

  The implemented algorithm for object tracking provides its position into the real-time captured frame. This information is considered to correct robot route in order to get a better representation. When stopping criteria are reached, an object picture is taken.

- *Acquired Representation*

  From a multicolored picture of 320x240 pixels, some pre-processing and segmentation steps are taken: (1) Grayscale Transformation, (2) Segmentation, (3) Threshold Calculation and Binarization and (4) Dilation.

  After completion of these steps, an 320×240 pixel binary image is achieved with two defined regions, the object and the background. This image is downscaled to a 16x12 matrix, followed by feature extraction using a method similar to the one adopted in the scenario mentioned before. In this case, the vector of attributes is comprised of:

  - Attributes 1-11: difference between the number of 1's in line *L* and line *L-1*, from top to bottom;
  - Attribute 12: number of 0's in the eleven extracted features;
  - Attribute 13: number of 1's in the eleven extracted features.

4.3 Ruc SCENARIO

- *Stopping Criteria:*

  Only ultrasonic sensor is considered for deciding upon when the robot stops in front of the object. According to results observed during the development of this scenario, ultrasonic sensor is more appropriate because the camera can not often estimate whether the object is near or far from the robot. Stopping is performed when the sensor scans a distance of 20cm from the object to be captured by the camera.

- *Method for Obtaining Representation:*

  Starting with navigation performed by keeping the detected object in center of camera field of view, the robot moves toward the object to be recognized. At that time, the ultrasonic sensor decides for stopping the robot and a camera frame is captured as the object representation.

- *Acquired Representation*

  The frame captured when the robot stops will be pre-processed and segmented in the same way of Rc Scenario. At the end, the image is downscaled to a 16x12 matrix, followed by feature extraction method present in the previous scenario. The vector of attributes is comprised of the same 13 attributes.

5 THE OBJECT RECOGNITION TASK

5.1 OBJECT TRACKING

The Rc scenario makes use of a camera to perform object tracking. While moving randomly, camera detects an object and the robot start tracking it. It adjusts its trajectory considering object position in camera view. The robot must have a prior knowledge of object shapes to be recognized. HaarTraining classification function, which is available in OpenCV library [1], is used to train an object detector. It was originally proposed in [15] and broadly adopted for real-time scenarios.

The recognizer consists of a boosted classifier which is trained to consider some critical features extracted from image and then detect patterns in images extremely quickly. The algorithm implemented in OpenCV considers an improvement to Viola-Jones implementation [15].

Such object detector classifies the images as object or background. Classification concerning object shape, triangular and rectangular, is performed later by neural network in order to compare the efficiency of all scenarios, even ultrasonic based one.

5.1.1 OBJECT DETECTOR

An object detector is trained to look for two classes of object shapes: rectangular or triangular. To achieve this purpose, 340 positive samples of rectangular shapes and the same number of triangular objects were collected and then reflected, totaling 1360 samples. Training database is composed of samples with different lighting and perspective. 300 negative images are also provided to train the detector, which are taken from different environment backgrounds.

All samples are transformed to grayscale and submitted to histogram equalizarion. Subsequently, they are downscaled from 320x240 to 32x24 pixels. Such scaling provides a reduction in the number of features to train the detector.

5.2 STEPS FOR IMAGE PROCESSING

Captured image of the object should be pre-processed after being submitted to segmentation. Afterwards, a segmentation algorithm is combined with other techniques of image processing to separate the main subject (the object) of the rest of the image. In order to differentiate two shapes, features of segmented image are extracted and given to the classifier. This pipeline is composed of 5 stages:

1. **Transformation to Grayscale** In order to make object recognition invariant to color, we transform the colored captured image from RGB format (Fig. 4a) to grayscale (Fig. 4b).

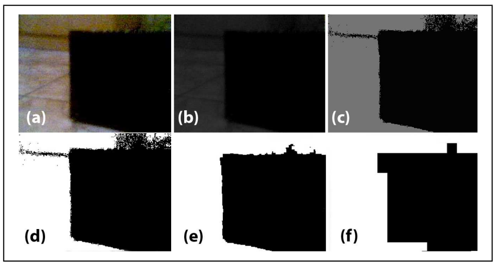


Figure 4: Example of image manipulation stages

2. **Segmentation** K-means algorithm is used to perform image segmentation Fig. 4c. K-means algorithm parameters are defined as follows: k = 2, Max. iterations = $10,000$, $\epsilon = 0.2$, and random centroid initialization.

3. **Threshold Calculation and Binarization** Some global thresholding techniques were tested and the best results were obtained with the Maximum Entropy Threshold technique. The threshold calculated for Fig. 4c was 4.5, which produced the binary image on Fig. 4d.

4. **Dilation** We used a digital filter for noise reduction, Fig. 4e, based on mathematical morphology. It was used a dilation operation with a 9×9 squared mask.

5. **Downscaling** Finally, the imagem is scaled in order to make its recognition feasible. The image is downscaled from 320×240 to 16×12 pixels (Fig. 4f)

5.3 CLASSIFICATION

A MLP Neural Network trained with error backpropagation technique is used to classify objects along two classes: rectangular or triangular. Two different architectures of neural networks have been defined.

One classifies data from the ultrasonic sensor and another receives the attributes obtained through the camera. The first is composed of 8 neurons in input layer, 5 in hidden-layer and 2 neurons in output layer. The second architecture is comprised of 13 neurons for input layer, 7 in hidden-layer and 2 neurons in output layer. Those networks have simple architectures which favor embedding classifier code in the robot in future works.

The number of neurons in the input layer corresponds to the size of attributes vector obtained from the ultrasonic sensor and camera. In the first case, only eight features are considered due to the error accumulated during sensor rotation, which would be worse if the number of measurements is increased. In the second case, the number of features is not closer to the first because, if reduced, image loses object shape representation.

The following parameters have been defined to train both Backpropagation MLP Networks: learning rate (μ) = 0.9, momentum = 0.2, max. of iterations = 25.

6 EXPERIMENTS AND RESULTS

In this section we describe a series of experiments conducted with the purpose of measure the classification performance of the neural network and discuss the overall performance of the above described scenarios.

In order to validate the neural network performance, a 10-fold cross validation has been used and paired t-tests have been accomplished afterwards. We have analyzed different values for learning rate and maximum number of iterations. Ultrasonic classification and the image classification have been compared by ROC curve analysis.

We have considered two datasets:

- Ultrasonic dataset - 60 samples were taken for the 10-fold cross-validation.
- Image dataset - 222 samples were taken for the 10-fold cross-validation. The set is comprised of 111 triangular and 111 rectangular samples.

For the experiments settings, two different values of learning rate have been chosen, 0.9 and 0.3. A higher learning rate converges faster than lower ones. On the other hand, it's more likely to reach a local minimum. We have also varied the maximum number of iterations along the values of 25, 50, 100, 500, 10000 and 200000.

The procedures were similar for both datasets. The 10-fold cross validation has been held for each experiment configuration and repeated ten times in order to acquire a better statistical significance for the success rate. The samples to each fold were randomly chosen, but the same folds were used for each model. Tables 1 and 2 show the average success rate and their respective standard deviation for both image and ultrasonic datasets.

Table 1: Image dataset with success rates for different learning rates (LR).

Maximum Number of Iterations	25	50	100	500	10000	200000
Success Rate (%) [LR = 0.3]	90.92	91.41	92.09	92.66	92.44	92.67
Standard Deviation (%) [LR = 0.3]	5.95	5.66	5.48	4.73	5.03	5.23
Success Rate (%) [LR = 0.9]	91.10	92.17	92.63	92.68	92.26	92.22
Standard Deviation (%) [LR = 0.9]	6.34	5.72	5.51	5.26	5.04	4.90

Table 2: Ultrasonic dataset with success rates for different learning rates (LR).

Maximum Number of Iterations	25	50	100	500	10000	200000
Success Rate (%) [LR = 0.3]	81.67	81.17	80.50	78.67	79.17	79.00
Standard Deviation (%) [LR = 0.3]	15.98	15.65	15.54	16.25	15.96	15.28
Success Rate (%) [LR = 0.9]	80.67	80.00	81.17	80.50	80.50	80.50
Standard Deviation (%) [LR = 0.9]	16.02	16.07	14.92	15.36	14.80	15.54

Results show that image-based recognition performed better than ultrasonic ones. In addition, it is possible to notice that increasing the value of maximum iterations does not necessarily improve the performance of the model.

Tables 3, 4, 5 and 6 show some important statistical measures for a 10-fold cross validation run. Results show an average *f-measure* above 0.9 for the image-based recognition, which means a proper commitment between *precision* and *recall*. In other words, this shows that the classifier performs well in detecting true positive patterns and false positive ones. The area under ROC curve also has high values (above 0.95), which confirms the good performance of this kind of classifier. Ultrasonic-based recognition has shown lower *f-measure* values (around 0.8) and area under ROC curve (around 0.85). The ROC curves for both classes, rectangular shapes and triangular shapes, can be seen in figures 5(a) and 5(b), respectively.

According to the paired t-test performed, there is no evidence of a significant difference (at the 0.05 level) between the success rate of the classifier with learning rate of 0.9 and 25 maximum iterations and the other ones. Since it is the one with higher convergence rate and thus, lower computational burden, it should be chosen.

Table 3: Statistical results for a 10-fold cross validation run, using the image dataset with learning rate of 0.3.

Maximum Number of Iterations	25	50	100	500	10000	200000
F-Measure	0.923	0.932	0.928	0.914	0.919	0.919
Area under ROC curve	0.972	0.968	0.969	0.977	0.974	0.973
Mean Absolute Error	0.110	0.101	0.092	0.077	0.084	0.085
Relative Absolute Error (%)	22.1	20.1	18.3	15.3	16.9	17.0

Table 4: Statistical results for a 10-fold cross validation run, using the image dataset with learning rate of 0.9.

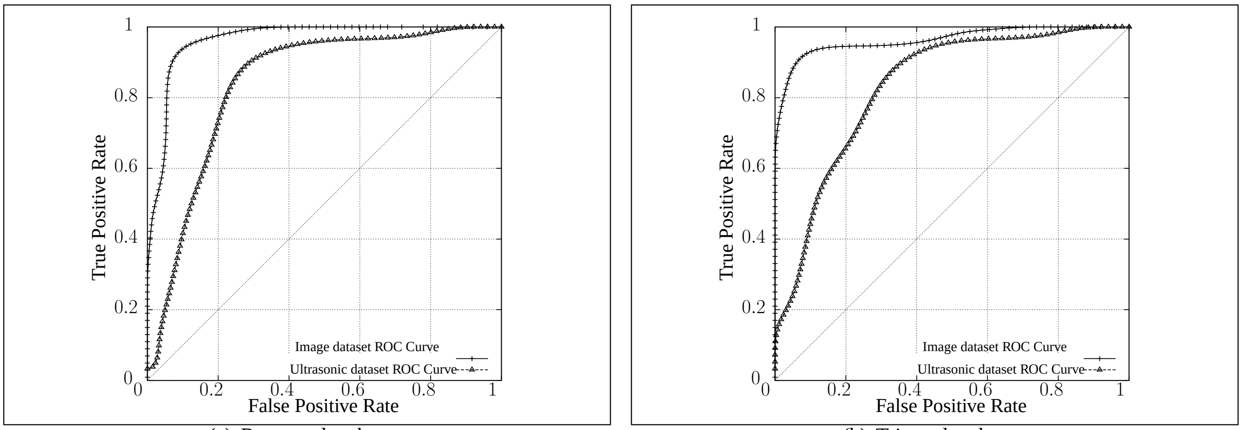
Maximum Number of Iterations	25	50	100	500	10000	200000
F-Measure	0.937	0.932	0.910	0.910	0.905	0.914
Area under ROC curve	0.968	0.968	0.965	0.969	0.972	0.974
Mean Absolute Error	0.100	0.096	0.090	0.093	0.094	0.090
Relative Absolute Error (%)	20.0	14.1	18.1	18.6	18.8	18.0

Table 5: Statistical results for a 10-fold cross validation run, using the ultrasonic dataset with learning rate of 0.3.

Maximum Number of Iterations	25	50	100	500	10000	200000
F-Measure	0.817	0.817	0.783	0.733	0.767	0.750
Area under ROC curve	0.858	0.856	0.846	0.859	0.846	0.838
Mean Absolute Error	0.296	0.263	0.248	0.256	0.261	0.264
Relative Absolute Error (%)	59.1	52.7	49.7	51.2	52.2	52.8

Table 6: Statistical results for a 10-fold cross validation run, using the ultrasonic dataset with learning rate of 0.9.

Maximum Number of Iterations	25	50	100	500	10000	200000
F-Measure	0.800	0.816	0.800	0.816	0.817	0.800
Area under ROC curve	0.846	0.843	0.870	0.866	0.877	0.873
Mean Absolute Error	0.248	0.253	0.238	0.219	0.188	0.186
Relative Absolute Error (%)	49.6	50.5	47.5	43.7	37.6	37.2



(a) Rectangular shapes (b) Triangular shapes

Figure 5: ROC curves on the classification of different shapes

7 CONCLUSION

Low-cost platforms and Lego Mindstorms NXT in particular are suitable for developing complex tasks such as object recognition. In this paper we have described experiments developed in three different sensing scenarios based on ultrasonic and camera sensors.

Ten-fold cross validation with paired t-test has been used to evaluate the classification models, neural networks, of each sensor and to measure which parameters were the best. The chosen parameters were 0.9 learning rate and 25 maximum iterations which leads to a low computational burden and success rates of 0.91 and 0.81 for the camera and ultrasonic recognition, respectively. This shows that the camera-based performance overcomes the ultrasonic one. The combination of camera-based recognition with ultrasonic-based tracking have shown a better performance if compared to scenarios consisting of just one kind of sensor.

The method presented in this paper is supported by algorithms that enable great object shape acquisition and consists of image processing techniques which enhance camera capabilities. R_{uc} successful scenario might be adopted to solve further robotic issues which depend on knowledge of environment. Future work can also test different low-priced sensors in order to compare its effectiveness as well as propose different methods for obtaining object shape representation.

REFERENCES

[1] Bradski, G.: The opencv library. Doctor Dobbs Journal 25(11), 120–126 (2000)

[2] Dinh, H., Inanc, T.: Low cost mobile robotics experiment with camera and sonar sensors. In: ACC'09: Proc. of the 2009 Conference on American Control. pp. 3793–3798. IEEE Press (2009)

[3] Eggert, D.: Using the Lego mindstorms NXT robot kit in an introduction to C programming class. Journal of Computing Sciences in Colleges 24(6), 8–10 (2009)

[4] Ferri, B., Ahmed, S., Michaels, J., Dean, E., Garyet, C., Shearman, S.: Signal processing experiments with the Lego Mindstorms NXT kit for use in signals and systems courses. In: Proc. of the 2009 Conf. on American Control. pp. 3787–3792. IEEE (2009)

[5] Gasperi, M., Hurbain, P., Hurbain, I.: Extreme NXT: Extending the LEGO Mindstorms NXT to the next level. Apress (2007)

[6] Klassner, F., Anderson, S.: Lego MindStorms: Not just for K-12 anymore. IEEE Robotics & Automation Magazine 10(2), 12–18 (2003)

[7] Kramer, J., Scheutz, M.: Development environments for autonomous mobile robots: A survey. Auton. Robots 22(2), 101–132 (2007)

[8] Lee, T.: Real-Time Face Detection and Recognition on LEGO Mindstorms NXT Robot. Advances in Biometrics 4642, 1006–1015 (2007)

[9] McNally, M., Klassner, F.: Demonstrating the Capabilities of MindStorms NXT for the AI Curriculum. In: American Association for Artificial Intelligence (2007)

[10] Sipitakiat, A., Blikstein, P.: Think globally, build locally: a technological platform for low-cost, open-source, locally-assembled programmable bricks for education. In: Proc. of the Fourth Int. Conf. on Tangible, Embedded, and Embodied Interaction. pp. 231–232. ACM (2010)

[11] Solano-Aragón, C., Alanis, A.: A Multi-agent Architecture for Controlling Autonomous Mobile Robots Using Fuzzy Logic and Obstacle Avoidance with Computer Vision. Bio-inspired Hybrid Intell. Systems for Image Analysis and Pattern Recognition pp. 215–246 (2009)

[12] Stevenson, D.E., Schwarzmeier, J.D.: Building an autonomous vehicle by integrating lego mindstorms and a web cam. In: SIGCSE '07: Proc. of the 38th SIGCSE technical symposium on Computer science education. pp. 165–169. ACM (2007)

[13] Talaga, P., Oh, J.: Combining AIMA and LEGO mindstorms in an artificial intelligence coursetobuild realworldrobots. Journal of Computing Sciences in Colleges 24(3), 56–64 (2009)

[14] Tribelhorn, B., Dodds, Z.: Evaluating the Roomba: A low-cost, ubiquitous platform for robotics research and education. In: 2007 IEEE Int. Conf. on Robotics and Automation. pp. 1393–1399 (2007)

[15] Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. Comp. Vision and Pattern Recognition, IEEE Computer Society Conf. on 1, 511 (2001)

[16] Wu, C.C., Tseng, I.C., Huang, S.L.: Visualization of program behaviors: Physical robots versus robot simulators. In: ISSEP '08: Proc. of the 3rd Int. Conf. on Informatics in Secondary Schools - Evolution and Perspectives. pp. 53–62. Springer-Verlag (2008)