Faster R-CNN for object location in a Virtual Environment for sorting task

https://doi.org/10.3991/ijoe.v14i07.8465

Javier O. Pinzón Arenas, Robinson Jiménez M(✉), Paula C. Useche Murillo
Mechatronics Engineering Program, Faculty of Engineering, Nueva Granada Military University, Bogotá, Colombia
robinson.jimenez@unimilitar.edu.co

Abstract—This paper presents the implementation of a mobile robotic arm simulation whose task is to order different objects randomly distributed in a workspace. To develop this task, it is used a Faster R-CNN which is going to identify and locate the disordered elements, reaching 99% accuracy in validation tests and 100% in real-time tests, i.e. the robot was able to collect and locate all the objects to be ordered, taking into account that the virtual environment is controlled and the size of the input image obtained from the workspace to be entered to the network should be 700x525 px.

Keywords—Faster R-CNN, Object Recognition, Virtual Environment, Autonomous Mobile Agent, Region of Interest.

1 Introduction

The applications of robotics in different areas have had an exponential growth in recent years, such as in medicine with surgical assistants [1,2], in the agricultural sector [3], in the industry for high-risk tasks [4] or even in tasks of daily life with assistance robots [5,6] or service in the cleaning area [7]. For the control of the robots, a great variety of techniques have been implemented, among which are those related to Machine Learning. For example, in [8] an adaptive model based on neural networks is used for the planning of trajectories and evasion of obstacles, or in other cases, Deep Learning techniques are used, such as the Convolutional Neural Networks (CNN) [9], where through these it is possible to perform the remote control of a mobile robot by means of commands made by a user with different hand gestures, in such a way as that executes certain action [10].

Mainly, the CNN are used for image recognition applications, managing to discriminate up to 1000 different categories, as demonstrated by the AlexNet network [11], it has even improved its operation by making them increasingly deeper, involving 19 convolution layers, which allows it to learn more details in large-scale images [12]. On the other hand, thanks to the great performance that CNN has had in multiple applications, mainly in recognition of patterns in images, it have been begun to create variants or improvements to this type of network, which strengthen and diversify the applications of this one. For example, CNNs have been combined with localization techniques, so that, apart from recognizing the object, it locates it through a Region of Interest (RoI), creating a so-called Region-Based CNN (or R-CNN) [13], which consists of using region proposal algorithms in conjunction with CNN, however, they have a long execution time, making their use in real-time applications inefficient. Due to this, in recent years the CNN architectures have been improved, through the implementation of new combinations to reduce processing times, as shown in [14], where Region Proposal Networks are used, obtaining a network called Faster R-CNN, which is not only faster, but also improves location accuracy.

The novelty of this work is in the application of the Faster R-CNN in the control of a mobile robotic arm in a virtual environment, which has as its task to collect disordered elements in a work area and place them in boxes. The Faster R-CNN will be responsible for identifying and locating the objects within the work environment, in such a way that the robot is able to move to the nearest one to perform the collection. In this way, it can be demonstrated the capacity of this neural network in tasks that require locating and grouping elements, initially validated in virtual environments, which can be migrated to real environments, in different types of applications, such as harvesting in crops, treatment of goods in warehouses or in search and rescue schemes.

The paper is divided into 4 parts, where section 2 describes the virtual environment implemented and the proposed Faster R-CNN architecture along with its training and validation. Section 3 presents the results obtained with real-time tests within the environment. Finally, section 4 shows the conclusions reached.

2 Methods and Materials

The implementation of this work is divided into two stages, where, in the first place, the virtual environment in which it is going to work is created, within which the task of sorting the objects will be performed. Following this, the training of an R-CNN is performed in such a way that, simulating a shot of an elevated camera, the objects to be ordered will be recognized and located. Said stages are described below.

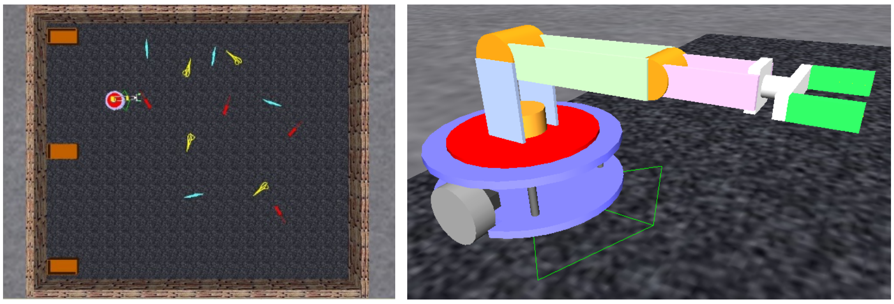
2.1 Virtual Environment

The work environment consists of a virtual 3D environment, whose vertical axis is the Y axis, inside which there are 3 types of objects distributed on the ground (see Figure 1). These objects are randomly placed within the limits indicated by the walls, which indicate the range of vision of the upper cam, when starting each test. The elements used are scissors (yellow), scalpels (cyan) and screwdrivers (red), which have previously been marked with a specific color, this is done as a learning support for the neural network that locates them, since at the distance the camera is located, some are not clearly distinguishable and can cause a high level of confusion in their recognition.

On the other hand, there are 3 boxes located on the left side (see Figure 1a) where each of the tools will be carried, with the upper box for the scalpels, the middle for the scissors and the bottom for the screwdrivers.

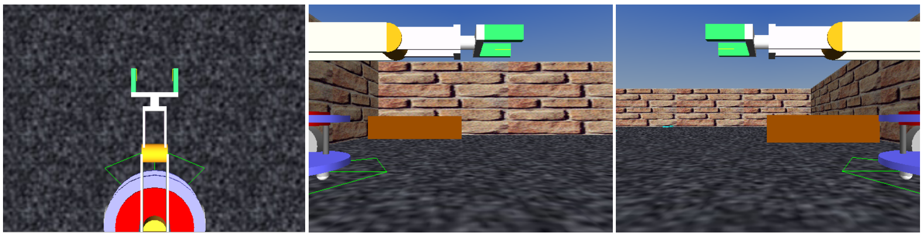
Finally there is the robot (see Figure 1b), that is composed of a mobile part and a manipulator, so that it can move to the recognized element, achieve the grip and then leave it in the place that corresponds to the type of tool. Additionally, the mobile part has a proximity sensor in its front to avoid collision with the objects and also reach a suitable distance to be able to make the grip of the element correctly.

For purposes of better visualization of the workspace and the robot, within the virtual environment, it can be chosen different views apart from the global one taken by the upper camera of Figure 1a. These views are the "Top of the robot" (Figure 2a) that focuses mainly on the gripping area of the end effector, the "Right side of the robot" view (Figure 2b) and the "Left side of the robot" view (Figure 2c). Each of the auxiliary views follows the robot from the perspective shown.



(a) (b)

Fig. 1. Virtual environment seen from the top camera and Mobile Robotic Arm



(a) (b) (c)

Fig. 2. Auxiliary Views

2.2 R-CNN Architecture

Previous developments using R-CNN, such as the one presented in [15] for the recognition of two hand gestures, have allowed to identify that the execution time of this network tends to increase depending on the number of contours in the image and the size of it, due to the RoI algorithm used as a previous step to the use of CNN. Taking into account the above, due to the amount of contours that are found in the environment by the objects and the type of ground of the proposed development, the use of a contour algorithm for the RoI would greatly delay the process of identifying the elements, especially considering the size of the image to be used (700x525 pixels). For this reason, the recognition of each of the objects distributed on the floor of the work environment is done through a Faster R-CNN, as described in [14], where, in addition to the Fast R-CNN identification, an RPN or Region Proposal Network is used, which drastically improves the speed of recognition, since it does not analyze each edge or change in the image, but basically acts as a Fully-convolutional Network [16], making part of the global prediction of CNN to train, having a high performance in obtaining the RoI and in the global execution time of the algorithm.

With this, the architecture shown in Figure 3 is proposed, where the network has an RGB image input of 32x32 pixels, i.e. the input of the proposed regions in the database cannot be smaller than the size of the proposed input. The advantage with this type of neural network is that it does not require a fixed size of the input images.

To train the Faster R-CNN, a database of 1000 images is built with a single object per category to be identified, where each one is labeled manually, obtaining images like those shown in Figure 4. Of these, 90% are taken for training 10% to validate the network once it has been trained.

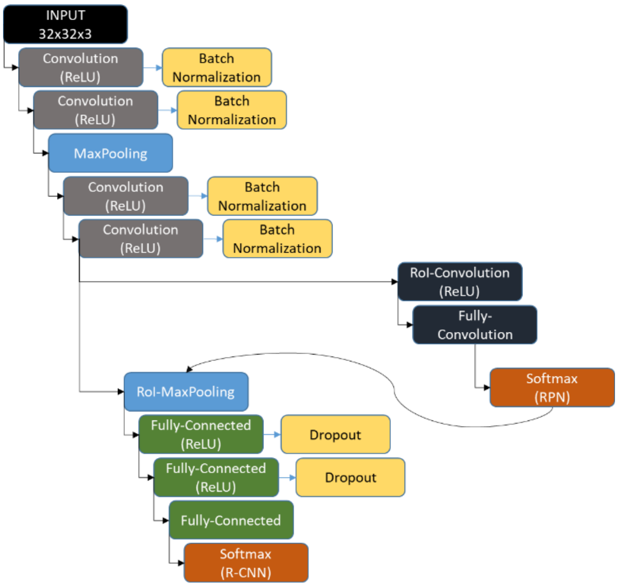


Fig. 3. Faster R-CNN Architecture

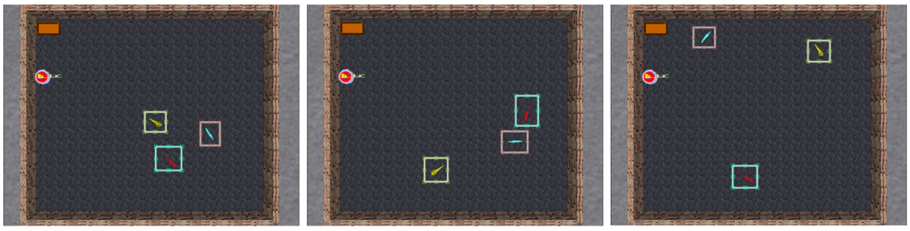
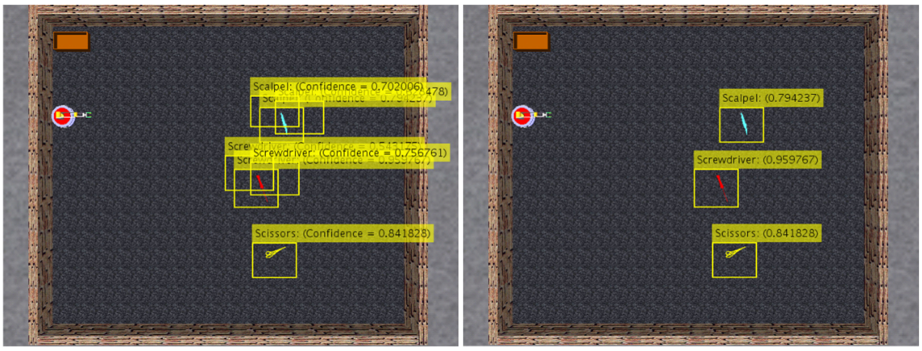


Fig. 4. Dataset Samples.

Once the network is trained, it is evaluated to verify its correct functioning in terms of an adequate positioning of the bounding box and that the prediction corresponds to the enclosed object. As shown in Figure 5a, several boxes are generated with the correct label on a single element, however, this may present a problem when used in conjunction with the mobile robot, for this reason, a post-processing phase of elimination of overlapped bounding boxes is done, so that only the most significant remains, as shown in Figure 5b. The elimination is done by means of the detection of label repetitions on the areas of the box detected in a sector, for example, if a box that recognized a scalpel has more with the same label, the one that has the highest degree of confidence is chosen and the others are eliminated. If one of the scissors is on the scalpel box, it is not eliminated, since its label is different. This is done with each element found.


Scalpel: (Confidence = 0.702006)
Screwdriver: (Confidence = 0.756761)
Scissors: (Confidence = 0.841828)
Scalpel: (0.794237)
Screwdriver: (0.959767)
Scissors: (0.841828)

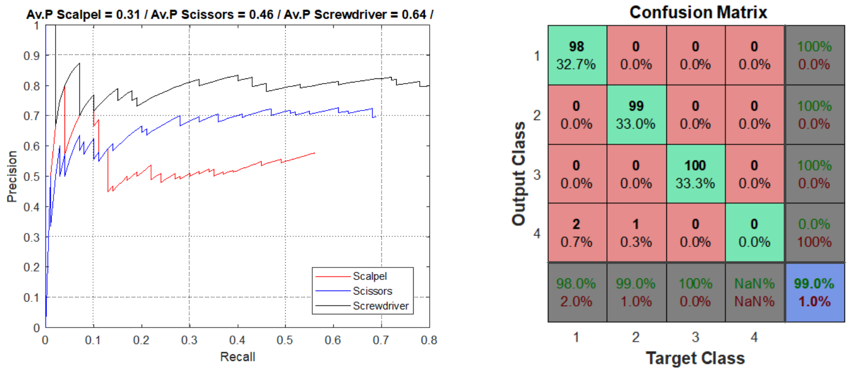
(a)

(b)

Fig. 5. a) Original recognized boundaries and b) after post-processing of overlapped RoIs

Once the post-processing phase is done, the accuracy of the network is evaluated by means of different levels of Recall and that each bounding box generated corresponds to the element it contains. These results are shown in Figure 6, obtaining a not very high accuracy in the recall (Figure 6a), in other words, the bounding box generated does not have an accurate overlapping over those presented in the dataset, especially in the scalpel, this is because, as shown in Figure 4, the boxes have varied sizes, while those generated in the network are more standard for each element, making the boxes larger for the scissors and the scalpel, while in the screwdriver, they are almost the same size. Regarding the prediction of the categories, the network obtained a 99% in the overall accuracy, as shown in Figure 6b, where no object was wrongly recognized, but 2 scalpels and scissors were not detected, probably because they were under another element, causing them not to be found.

These tests show that the Faster R-CNN has a high degree of accuracy in recognizing and locating objects, regardless of whether the size of the bounding box does not correspond exactly to the theoretical size of the dataset. In addition to this, the execution time of the network to recognize objects in the environment only takes 197.7 ms, which is a very fast response time taking into account the size of the image and compared with the previous work done [15].



(a)

Confusion Matrix

Output Class \ Target Class	1	2	3	4	
1	98 32.7%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
2	0 0.0%	99 33.0%	0 0.0%	0 0.0%	100% 0.0%
3	0 0.0%	0 0.0%	100 33.3%	0 0.0%	100% 0.0%
4	2 0.7%	1 0.3%	0 0.0%	0 0.0%	0.0% 100%
	98.0% 2.0%	99.0% 1.0%	100% 0.0%	NaN% NaN%	**99.0%** **1.0%**

(b)

Fig. 6. a) Behavior of the RoI detection at different recalls in each category and b) Accuracy of the Faster R-CNN, where 1, 2 and 3 are Scalpel, Scissor and Screwdriver, respectively, and 4 is that the object does not exist in the workspace or there is no RoI detected.

To know the approximate position of the object in such a way that the robot can move to it, a conversion from pixels to centimeters must be made, since the center of each box that contains the object in the image is known. In this way, with the initial location of the robot, which for simulation purposes is the point (0,0) in centimeters (or (97,175) in pixels, taking into account that the coordinates are in [X, Z]), the position of each element, which is known within the database, and its location within the image in pixels, they are used to find the distance-pixel relationship shown in (1), where R is the conversion factor of pixels to centimeters, d_i is the current distance of the element (known), P_i represents the position of the element in pixels, P_o is the initial position of the robot in pixels and n refers to the number of samples to be taken, that is, the number of elements whose data was used. This operation takes into account the position in both X and Z.

$$R = \frac{\sum_{i=1}^{n} \frac{d_i}{P_i - P_o}}{n} \tag{1}$$

The ratio factor of conversion of pixel to centimeters, for this case, results in a value $R \approx 0.0079$ for both the X and Z axis. With this, it is proceeded to obtain the central position of the box generated by the network for any object randomly positioned, that is, in the cases the value of d is not known. To get *d*, for the X and Z axis, the equations (2) and (3) are shown.

$$d_x = \left(\left(P_x + {}^{W}\!/_{2}\right) - P_{xo}\right) R \tag{2}$$

$$d_z = \left(\left(P_z + {}^{L}\!/_{2}\right) - P_{zo}\right) R \tag{3}$$

Where d is the distance from the starting point of the robot to the object in centimeters, P is the point of the left vertex of the bounding box in pixels, W and L are the width (in X) and long (in Z) of the box, P_o represents the starting point of the robot in pixels and R the ratio factor.

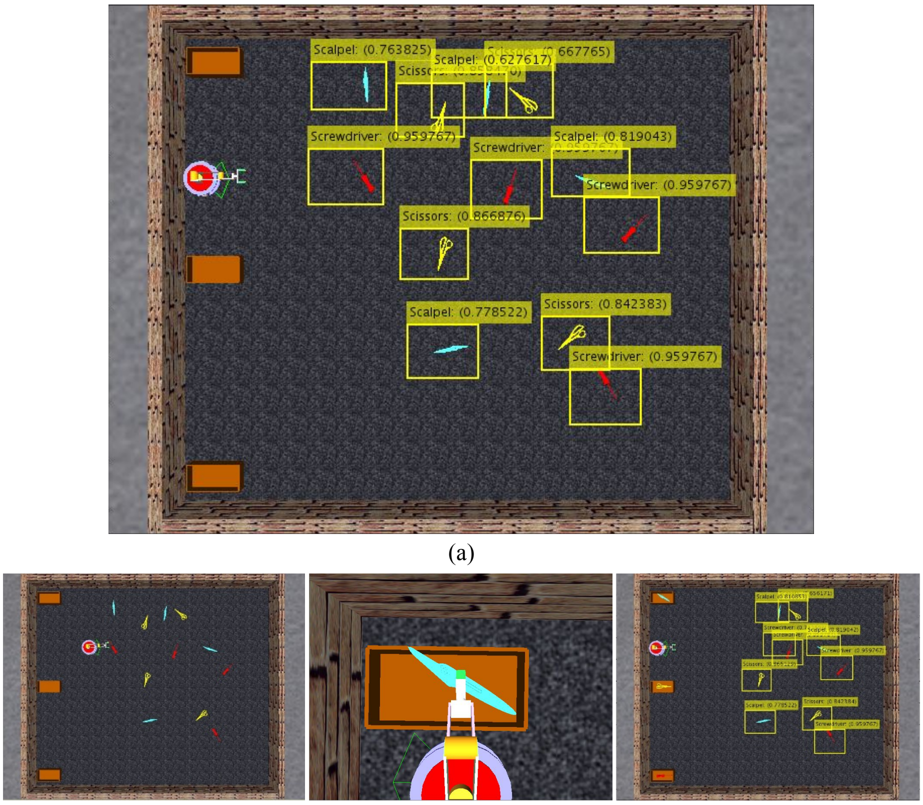
3 Results and Discussions

Working tests are performed to observe the efficiency of the developed algorithm, making repetitions of the collection task in such a way that the Faster R-CNN is able to recognize the objects to be sorted during the whole task and they are taken the operation times of the simulation.

Within each test, as shown in Figure 7a, the first recognition of the environment is made, which presents a precision of 100%, i.e. managing to identify each of the elements distributed on the floor, regardless of whether they were very close to each other. Once this is done, in Figure 7b the robot approaches the nearest object, in this case the screwdriver, to proceed to make its grip. The second element collected is the upper scalpel, where it can be seen in Figure 7c how the robot leaves the object in the box whose position is known. In Figure 7d, it is shown the capture of the workspace with 3 objects already collected and, since within the interface the post-processing of elimination of overlapped boxes is not performed, some objects are recognized twice, however, the robot will make sure to go to the nearest box. In Figures 7e, 7f and 7g, the collection of the last item is shown, where the robot is moving to its initial position, the work area is captured, and it is moved to the element.

On the other hand, it is important to take into account the execution times of the algorithm in terms of detection when all the objects are present, the robot's time to perform a successful collection and the total time of the task. The virtual environment was run on a laptop with the characteristics shown in Table 1. For purposes of computational cost reduction, the view of the upper camera is used to acquire the image that is processed by the Faster R-CNN and, followed by this, it changes to the top view of the robot, to avoid redrawing the objects that are not in view in each frame and thus reduce the number of processes within the virtual world.

Table 2 shows the average times obtained when the Faster R-CNN is executed to recognize the objects, the collection and location of an object made by the robot, and the time it took to organize all the elements in the boxes. To acquire the times, 5 repetitions of the task were done, taking a total of 60 samples for the times of the neural network and for each object, and 5 samples of the whole task. As can be seen, the Faster R-CNN only spends approximately 0.3 seconds, a more efficient time compared to an R-CNN using detection algorithms, where its time reaches more than 1.2 seconds, as reported in [15], being the algorithm used in this work 4 times faster. As for the task performed by the robot, it takes a time of 130 seconds on average to go from the starting point, collect the object, place it in the box and finally return to its initial position, clarifying that the farther the robot is from the element, the faster it will advance. However, when the global task is being finalized, the simulation tends to slow down, since when the mobile robot arrives at the boxes, it has to process each of the objects already deposited, taking approximately 10 seconds more. On the other hand, organizing all the objects took approximately 26 minutes, which in a simulation is not relevant, but if it is done in conjunction with a real robot, it can slow down the speed at which the real mobile can go.

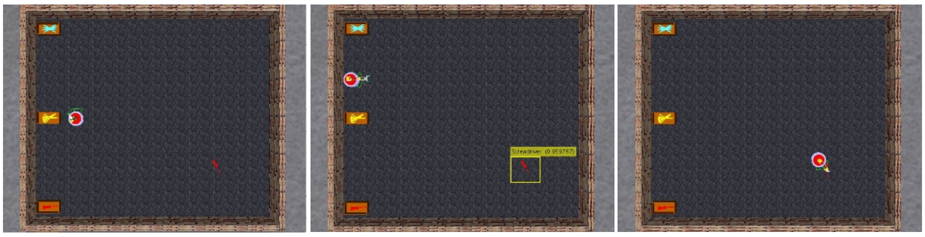


(a)

(b)

(c)

(d)



(e) (f) (g)

Fig. 7. Different moments of the test in mode “Auto”

Table 1. Computer Characteristics

Processor	RAM	GPU	GPU Memory	Clock Rate
i7-4510U 4th Generation Intel® Core™	16 GB	GeForce GT 750M	2048 MB GDDR5	941 MHz

Table 2. Average times

Faster R-CNN	Single Object	Task
0.3005 s	130.8750 s	26.1750 min

4 Conclusions

In this work, a virtual environment was created within which a mobile robotic arm is simulated to perform a specific task: collection and organization of objects. Within the development, a Faster R-CNN was used for identification and location of the objects, achieving a 99% accuracy in the recognition of each one, being a very high accuracy for this type of applications.

Although the Faster R-CNN was trained to recognize a single element per category, during the tests it was shown that it works optimally even by adding more elements, allowing the robot to collect each of the disordered objects. Likewise, it showed the speed with which a Faster R-CNN works in comparison with a basic R-CNN, achieving average times of 0.3 seconds to recognize 12 objects in an image of 700x525 px dimensions.

This work can be the basis to develop tasks of greater complexity, which can have trajectory planning with evasion of obstacles, collection of dangerous elements or in high risk areas by means of robots, among other applications, making use of neural networks such as the Faster R-CNN as a technique of identification and location of the objects to be manipulated. However, it is necessary to optimize the execution times of the simulation, using a dedicated computer or with better capacity than the one used in this work, so that it can operate at the same time as a real robot.

5 Acknowledgment

The authors are grateful to the Nueva Granada Military University, which, through its Vice chancellor for research, finances the present project with code IMP-ING-2290 (2017-2018) and titled "Prototype of robot assistance for surgery", from which the present work is derived.

6 References

[1] Jongwon Lee, Inwook Hwang, Keehoon Kim, Seungmoon Choi, Wan Kyun Chung, Young Soo Kim. “Cooperative robotic assistant with drill-by-wire end-effector for spinal fusion surgery”, Industrial Robot: An International Journal, 2009, vol. 36, no 1, pp. 60-72.

[2] S. W. Kuo, P. M. Huang, M. W. Lin, K. C. Chen and J. M. Lee, “Robot-assisted thoracic surgery for complex procedures”, Journal of thoracic disease, 2017, vol. 9, no 9, pp. 3105. https://doi.org/10.21037/jtd.2017.08.11

[3] J. Baeten, K. Donné, S. Boedrij, W. Beckers and E. Claesen, “Autonomous fruit picking machine: A robotic apple harvester”, In Field and service robotics, Springer Berlin Heidelberg, 2008, pp. 531-539. https://doi.org/10.1007/978-3-540-75404-6_51

[4] M. Hassan, D. Liu, G. Paul and S. Huang, “An approach to base placement for effective collaboration of multiple autonomous industrial robots”, In Robotics and Automation (ICRA), 2015 IEEE International Conference on. IEEE, 2015, pp. 3286-3291. https://doi.org/10.1109/ICRA.2015.7139652

[5] J. Broekens, M. Heerink and H. Rosendal, “Assistive social robots in elderly care: a review”, Gerontechnology, 2009, vol. 8, no 2, pp. 94-103. https://doi.org/10.4017/gt.2009.08.02.002.00

[6] R. M. Ferrús and M. D. Somonte, “Design in robotics based in the voice of the customer of household robots”, Robotics and Autonomous Systems, 2016, vol. 79, pp. 99-107. https://doi.org/10.1016/j.robot.2016.01.010

[7] S. Dubey, M. C. Chinnaaiah, C. S. Kiran, B. S. Priyanka and P. P. Rao, “An FPGA based service Robot for floor cleaning with autonomous navigation”, In Research Advances in Integrated Navigation Systems (RAINS), International Conference on. IEEE, 2016, pp. 1-6. https://doi.org/10.1109/RAINS.2016.7764425

[8] N. Aamer and S. Ramachandran, “Neural Networks Based Adaptive Approach for Path Planning and Obstacle Avoidance for Autonomous Mobile Robot (AMR)”, International Journal of Research in Computer Applications and Robotics (IJRCAR), 2015, vol. 3, no 12, pp. 66-79.

[9] M. D. Zeiler and R. Fergus, “Visualizing and Understanding Convolutional Networks”, 2013, arXiv preprint arXiv:1311.2901

[10] J. Nagi, et al. “Max-pooling convolutional neural networks for vision-based hand gesture recognition”, In 2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA), IEEE, 2011, pp. 342-347. https://doi.org/10.1109/ICSIPA.2011.6144164

[11] A. Krizhevsky, I. Sutskever and G. E. Hinton, “ImageNet classification with deep convolutional neural networks”, In Advances in neural information processing systems, 2012, pp. 1097-1105

[12] K. Simonyan and A. Zisserman, “Very Deep Convolutional Networks For Large-Scale Image Recognition”, 2015, arXiv preprint arXiv:1409.1556v6

[13] R. Girshick, J. Donahue, T. Darrell and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation”, In Proceedings of the IEEE conference on computer vision and pattern recognition, 2014. pp. 580-587.
[14] S. Ren, K. He, R. Girshick and J. Sun, “Faster R-CNN: towards real-time object detection with region proposal networks” In Proceedings of the 28th International Conference on Neural Information Processing Systems, MIT Press, 2015, pp. 91-99.
[15] J. O. P. Arenas, R. J. Moreno and P. C. U. Murillo, “Hand Gesture Recognition by Means of Region-Based Convolutional Neural Networks”, Contemporary Engineering Sciences, 2017, Vol. 10, no. 27, pp. 1329-1342. https://doi.org/10.12988/ces.2017.710154
[16] J. Long, E. Shelhamer and T. Darrell, “Fully convolutional networks for semantic segmentation”, In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2015, pp. 3431-3440.

7 Authors

Javier Orlando Pinzón Arenas was born in Socorro-Santander, Colombia, in 1990. He received his degree in Mechatronics Engineering (Cum Laude) and Specialization in Engineering Project Management at the Nueva Granada Military University - UMNG - in 2013 and 2016, respectively. He has experience in the areas of automation, electronic control and machine learning. Currently, he is studying for a Master’s degree in Mechatronics Engineering and working as Research Assistant at the UMNG with emphasis on Robotics and Machine Learning.

Robinson Jiménez Moreno was born in Bogotá, Colombia, in 1978. He received the Engineer degree in Electronics at the Francisco José de Caldas District University - UD - in 2002, respectively. M.Sc. in Industrial Automation from the Universidad Nacional de Colombia - 2012 and PhD candidate in Engineering at the Francisco José de Caldas District University - UD. He is currently working as a Professor in the Mechatronics Engineering Program at the Nueva Granada Military University - UMNG. He has experience in the areas of Instrumentation and Electronic Control, acting mainly in: Robotics, control, pattern recognition and image processing.

Paula Catalina Useche Murillo is a Mechatronics Engineer graduated with honors in 2017 from the Nueva Granada Military University in Bogotá, Colombia, where she currently studies a M.Sc. in Mechatronics Engineering and works as a research assistant in the Mechatronics Engineering program. She published two papers in the International Journal of Applied Engineering Research (IJAER) in 2016 about the implementation of Myo Armband in applications for manipulation of a robot arm through myoelectric signals. Her research focuses on the use of convolutional neural networks for object recognition, and image processing for grasp detection and trajectory planning.

Article submitted 20 February 2018. Resubmitted 22 February 2018. Final acceptance 22 April 2018. Final version published as submitted by the authors.