Journal of Engineering and Applied Sciences 12 (Special Issue 11): 9278-9285, 2017
ISSN: 1816-949X

Deep Convolutional Neural Network for Hand Gesture Recognition Used for Human-Robot Interaction

Javier O. Pinzon Arenas, Ruben D. Hernandez Beleno and Robinson Jimenez Moreno
Faculty of Engineering, Nueva Granada Military University, Bogota, Colombia

Abstract: This study presents the training and validation of a deep convolutional neural network architecture used for a human-robot interaction. Two different datasets of images were employed with the aim of recognizing 2 kinds of hand gestures which are "closed" and "open" and control a robotic arm with these gestures. To choose the best training in the network, different behavioral parameters such as training accuracy and loss were evaluated to obtain the best training epoch and validation parameters such as validation accuracy and internal behavior of the network through the activations of the convolution layers. Once the trained network is chosen, camera tests and interaction with a robotic arm are performed, evaluating the interaction between the user and the actions of the robot through the network.

Key words: Deep convolutional neural network, hand gesture recognition, layer activations, human-robot interaction, accuracy, parameters

INTRODUCTION

Neural networks have been a major field of research for over half a decade within the field of deep learning. Over the years, different techniques or methods have been developed for its enhancement from its beginnings with the perceptron model which focused on simple linear problems (Schmidhuber, 2015; Poonia *et al*., 2016) to more complex current models such as recurrent neural networks which, for example by Guo *et al*. (2017), allow predicting the useful life of bearing.

Due to the large number of applications that exist, neural networks have been developed for specific applications such as Convolutional Neural Networks (CNN) which are mainly designed for the recognition of patterns in images. Being introduced in the early 90's by LeCun *et al*. (1989), CNN have been developed in different fields for the recognition of patterns, thanks to its performance, mainly in the recognition of handwritten characters or even in the analysis of documents as shown by Simard *et al*. (2003). Additionally, in recent years, researchers have been working on very deep CNN (up to 19 convolution layers) for recognizing large-scale images (Simonyan and Zisserman, 2014).

Hand gestures are part of the interaction between humans (Singer and Goldin-Meadow, 2005) and therefore, have been an important part of the research in the implementation of convolutional neural networks focused on their recognition. In an early start using CNN for hand recognition by Nowlan and Platt a simple convolution layer architecture is developed for hand tracking and recognition. However, nowadays much more robust and deep architectures have been developed in order to recognize more of the characteristics of the hands. An example of this is developed by Barros *et al*. (2014) where it uses a multichannel CNN for hand gesture recognition where its input are 3 channels which are different to the normally used RGB channels.

On the other hand, applications of human-robot interaction using CNN have not been a mayor field of research where a nearby application is developed by Barros *et al*. (2015) in which a cross-channel CNN was develop in order to recognize emotional expression for the interaction with a robot. Another example is described by Wang *et al*. (2016) where a CNN was developed to grasp objects for a robotic gripper, however, there is no interaction with people. The novelty of the development of this research is this interaction which allows a user to control the action of a robotic arm depending on what gesture performs through a deep convolutional neural network where the success of such interaction is based on the process of choosing the trained network to be used.

MATERIALS AND METHODS

In previous trainings to iteratively determine the architecture of a convolutional neural network which was oriented to the recognition of hand gestures of opening and closing and is shown in Fig. 1a an accuracy of 73%

Corresponding Author: Javier O. Pinzon Arenas, Faculty of Engineering, Nueva Granada Military University, Bogota, Colombia

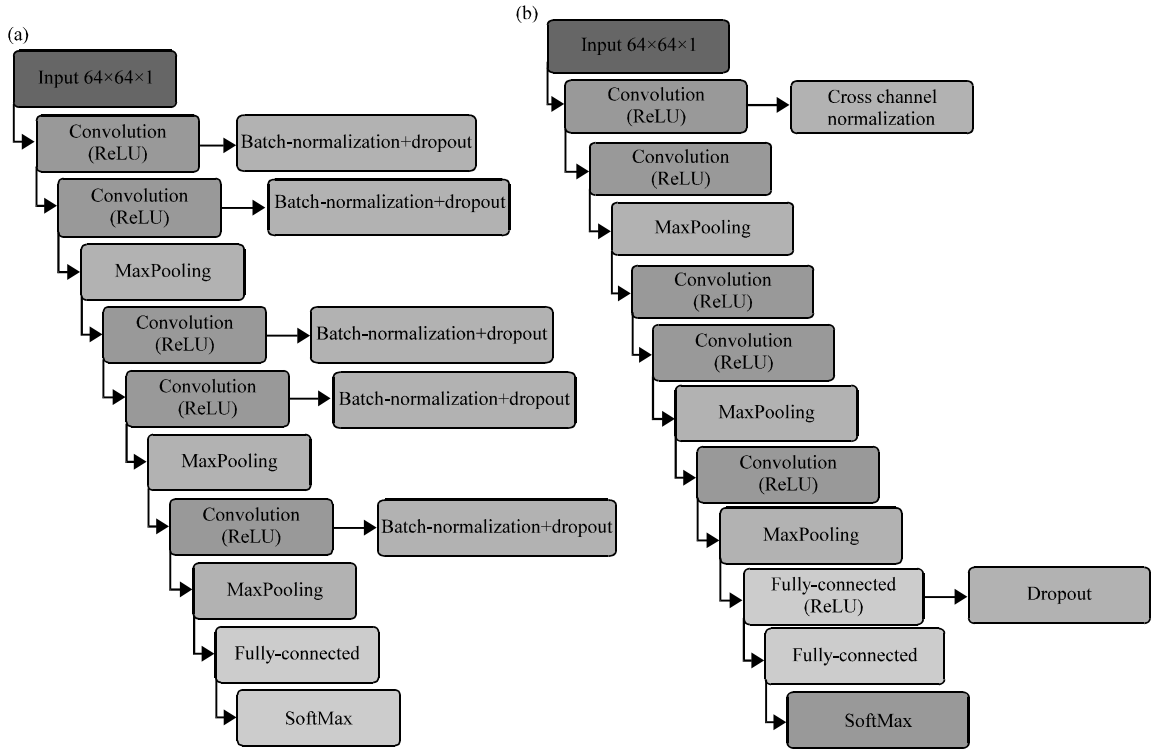

Fig. 1: Arquitectures developed in: a) Previous research and b) Current research

Table 1: Implemented architecture

Type	Kernel		Filters
Convolution	4×4	S = 1/P = 2	20
Convolution	4×4	S = 1	20
MaxPooling	2×2	S = 2/P = 1	-
Convolution	5×5	S = 1	50
Convolution	5×5	S = 1	50
MaxPooling	2×2	S = 2	-
Convolution	4×4	S = 1	200
MaxPooling	3×3	S = 2	-
Fully-Connected	1	200	
Fully-Connected	1	-	
Softmax	2	-	

was achieved using only 1 channel input (grayscale images). Based on this, modifications were made that allowed to obtain an architecture with better precision in the recognition of the gestures of the hand "open" and "closed", this time using the 3 color channels (RGB) as input as can be seen in Fig. 1b.

Network architecture: The architecture used has a flatten layer (or intermediate fully connected layer) which allows a greater learning of the combination of characteristics and a faster learning speed (Jin *et al*., 2014). Table 1 shows the architecture implemented in detail.

The output volumes of each layer were calculated by Eq. 1-3 which allows to determine the input sizes of the fully-connected layers and verify that each applied filter fits correctly in its input volume and there is no wrong sized output, i.e., the size of the output volume must be a positive integer number:

$$W_{n+1} = \frac{W_n - F_n + 2P_n}{S_n} + 1 \quad (1)$$

$$H_{n+1} = \frac{H_n - F_n + 2P_n}{S_n} + 1 \quad (2)$$

$$D_{n+1} = K_n \quad (3)$$

Where:

- n = Represents the input volume
- n+1 = Refers to the output volume
- W and H = The width and height of the input image of layer n, respectively
- P = The zero padding or addition of zeros around the input volume
- F = The size of the filter or kernel
- S = The step of the filters
- D = Specifies how deep the layer
- K = Indicates the amount of filters used in layer n

Training and validation of the network is done in MATLAB® Software.

Table 2: Datasets used

Dataset 1		Dataset 2		Validation	
Closed	Open	Closed	Open	Closed	Open
300	300	700	700	250	250
White background		Complex background		Mixed background	

Dataset: For the purpose of training the established network architecture, two databases were built, the first consists of 600 color images of hand gestures on a uniform background (white) with 300 images per category to be recognized (open or closed hand). The second database consists of 1400 color images, having 700 images of hand gestures for each category which are in different types of backgrounds which allows to have greater diversity in the data and that the network not only learn the gestures in a flat or uniform background but also allows a greater range of application regardless of where it is wanted to recognize the gesture required without the training being biased to a controlled environment. Since, the images have different sizes for the input to the network, these are scaled to a size of 64×64 pixels, this because the filters of the architecture are calculated for images of that size, searching reduce the training time. To validate each training of the network, a dataset was elaborated, that consists of 500 images in total (250 by category) where there are images in white background as in more complex backgrounds. Table 2 shows a summary of the databases.

For the training of the network, in order to strengthen the databases, a process of data augmentation is carried out which consists of randomly flipping the images with respect to the vertical and cuts of the same images, this for the dataset 1 and only cuts for the dataset 2 were used which are performed during training, i.e., a certain amount of the input images are changed randomly in each training epoch.

Network training: In order to achieve an effective interaction with the robot manipulator, it is important to evaluate the trained network with each dataset considering how the network is behaving with an input image, its training times and accuracy reached with the validation dataset.

Analyzing the behavior of the network during the training, it can be evidenced that the network trained with the dataset1 presents a better behavior than the network trained with the dataset 2 in terms of the accuracy achieved using the same amount of epochs as shown in Fig. 2. In addition in Fig. 3, it is shown the training loss, i.e., the cost of inaccuracy of the training which should be reduced as much as possible in other words, even if there is an accuracy of 100%, there is still a high training loss (>0.05 for this case), it means that for validation tests, data loss or imprecision in gesture recognition is going to occur, decreasing the validation accuracy which will be

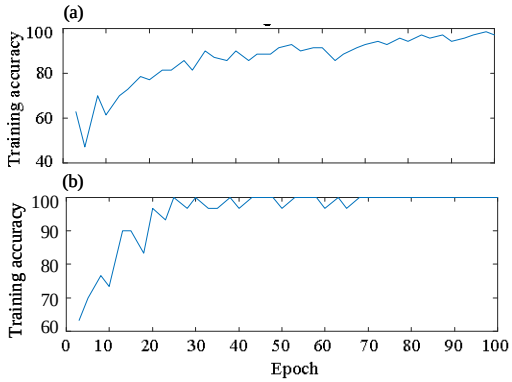

Fig. 2: Accuracy response; a) Training dataset 2 and b) Training dataset

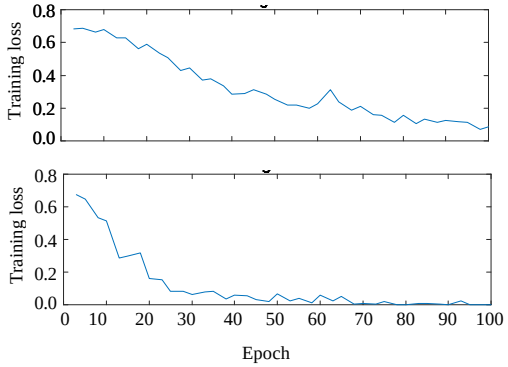

Fig. 3: Training loss; a) Training dataset 2 and b) Training dataset 1

shown later. Considering this, the network trained with the dataset 1 continues to have a better behavior regarding the training loss, however, it should be noted that for dataset 2, having more images and not only having a flat and uniform background, the amount of effort that has to be made to be able to differentiate hands from other objects is greater, therefore, only with 100 epochs is not enough to reach a behavior similar to the one trained with the dataset 1. For this, it was proposed a training with 300 times for the network with the dataset 2 and thus, observe its behavior which is shown in Fig. 4. This is done in order to be able to compare the two networks with similar accuracy and training loss.

Trained CNN validation: To make a comparison of the trained network with each dataset, 2 training epochs are taken from each trained network, an epoch with 100% accuracy and a training loss >0.04 (Epoch 1) and an epoch with an accuracy of 100% and the lower training loss obtained during its training (Epoch 2). Table 3 shows the selected epochs.

For each of the chosen epochs, the respective validation is performed, obtaining its results by means of

Table 3: Selected epochs of the training

	Dataset 1		Dataset 2	
Variables	Epoch 1	Epoch 2	Epoch 1	Epoch 2
Epoch	30	100	118	280
Training accuracy	100%	100%	100%	100%
Training loss	0.0639	0.0020	0.0673	0.0046
Training time (sec)	423.16	1409.54	4672.66	11068.00

Table 4: Network validation

	Dataset 1		Dataset 2	
Variables	Epoch 1	Epoch 2	Epoch 1	Epoch 2
Total images per category	250	250	250	250
True positive closed hand	137	215	234	224
True positive open hand	248	196	209	227
Overall accuracy (%)	77.0	82.2	88.6	90.2

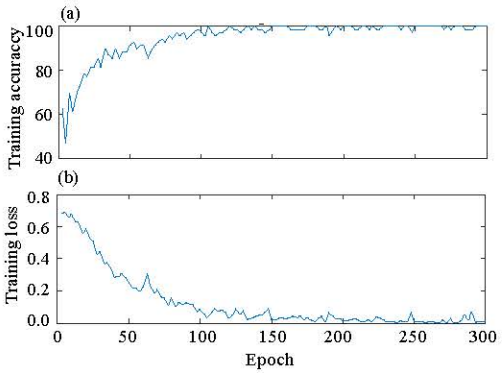

Fig. 4: CNN training with 300 epochs using Dataset 2; a) Training dataset and b) Training dataset

a confusion matrix, taking the true positive data of each category (how many times each category was well recognized from the total images presented) and the overall accuracy, shown in Table 4.

From Table 4, it can be observed that, although in Table 3 in each one of the epochs the training accuracy reached 100% with a higher training loss, they tend to have a lower overall percentage of validation accuracy. An example of this is presented in the network trained with dataset 1 where its cost of inaccuracy is high, reaching a value of 0.0639, causing it to recognize fewer closed hands in non-training images.

On the other hand, comparing the results obtained between the two best epochs trained with each dataset, for the case of the two Epoch 2, there is an 8% difference in accuracy, this is because as the validation dataset contained images with varied backgrounds, the network trained with the dataset 1 is not able to clearly differentiate the hands from the background, causing misclassifications.

The filters obtained from the first layer of the two Epoch 2 are shown in Fig. 5. Here, for example for Fig. 5a,

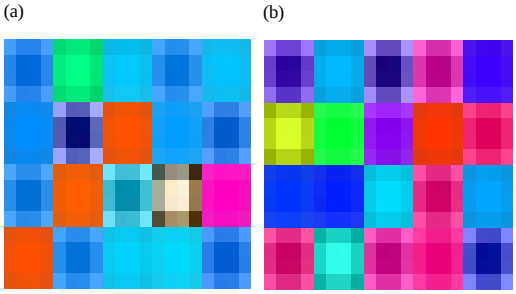

Fig. 5: Features of the first convolutional layer for CNN trained with; a) Dataset 1 and b) Dataset 2

Fig. 6: Strongest activations for closed hand in CNN using dataset 1

it is shown the variety of features learned by said layer. It is also noticed that the features of the network trained with the dataset 1 are much more homogeneous than those of the dataset 2, possibly because this first one learned background characteristics while the second one learned more variety of characteristics belonging to the hands.

Due to the difficulty of guessing what represents each of this features, there is another way to show how the already trained networks behave is through their activations in the hidden layers, i.e., see what the network is learning to recognize each of the categories. In these activations, beginning from the early layers, general characteristics such as shapes, edges and colors are learned and in deeper layers, features such as parts of the fingers or lines of the palm of the hand are learned. In this way, it is possible to understand and visualize how the network is working and its behavior with an input in each layer (Yu *et al*., 2015; Zeiler and Fergus, 2014).

For these tests, only the two best epochs of each trained network (Epoch 2) were taken. To show the activations, it was taken an image of each category and each of its stronger or relevant activations after the ReLU layer for each convolution. Figure 6 and 7 show the

Fig. 7: Strongest activations for open hand in CNN using dataset 1

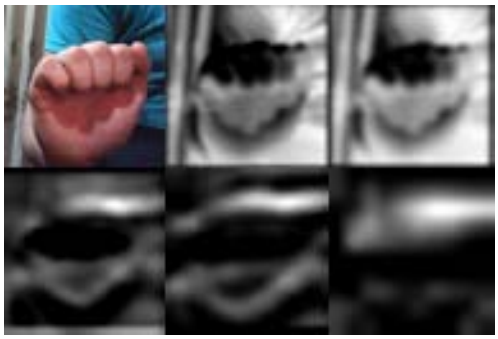

Fig. 8: Strongest activations for closed hand in CNN using dataset 2

original image of the closed and open hand, respectively and the activations of the convolution layers 1 and 2 at the top of the image and the convolution layers 3-5 at the bottom for the network trained with the dataset 1.

As it can be seen in Fig. 6, activations begin by recognizing first the background and then some features of the hand such as the edges of the fingers and part of the lower region of the hand. However, through each convolution layer, the background remains as a relevant part of the image and more specific features of the hand are not observed, this causes a misclassification of the image, recognizing it as "Open". The same happens in Fig. 7 where the first convolutions detail general features such as the contour but as it goes deeper in the network, they are lost and mixed with activations generated by other parts of the image such as the shoulder that is seen in the image, making it recognized as "Closed".

Compared to the previous trained network, more specific activations can be seen in CNN trained with dataset 2. For example, in Fig. 8 the first two convolutions are very similar with respect to Fig. 6 but from the convolution 3 onwards marked differences are shown such as that appreciated in the convolution 3 where the

Fig. 9: Strongest activations for open hand in CNN using dataset 2

strongest activation belongs to the identification of specific characteristics of the hand such as the edges of the fingers and something relevant is that it discriminates the background, making it disappear from the upcoming activations. In the same way, it occurs when comparing Fig. 9 and 7 where in the first case, if it is possible to detail lines of the palm in convolutions 3 onwards, it also activates what are the fingertips in the convolution 4 and the contour of the hand is more demarcated and although, it continues activating the person's shoulder, it has separate the activations with the ones belonging to the hand. For both cases of dataset 2, it correctly recognized each category.

Once the validation tests have been performed, the network that will be used for the interaction with the manipulator can be chosen. For this case, the CNN trained with the dataset 2 in the "Epoch 2" is chosen which although, in its training it needed more epochs to be able to reach a better accuracy and a reduced training loss, its answer regarding validation was superior to that trained with the dataset 1, obtaining a greater general accuracy and more successful activations.

RESULTS AND DISCUSSION

Camera test: With the CNN already chosen, it is proceeded to make tests with a webcam in real-time. This test is performed with 8 subjects different from those belonging to the training and validation dataset. The 10 tests per category are done to each person, placing the hands in different positions and observing if the network recognizes them correctly. Some results of the tests are shown in Fig. 10.

In the real-time tests where their results are summarized in Table 5, it was obtained that the network recognized the open hands in 97.5% and closed in 95%, meaning that, compared to the results obtained in Table 4, the chosen network is responding better when

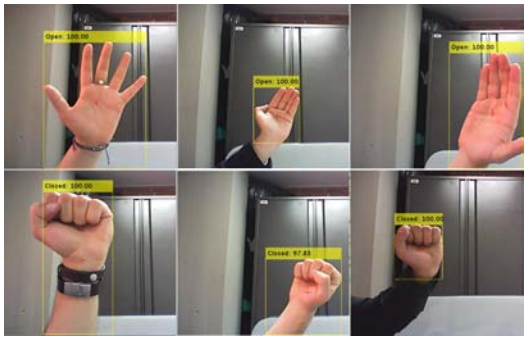

Fig. 10: Camera test samples

Table 5: Results of the network in the camera test

	Category	
Subjects	Closed	Open
1	10	10
2	10	9
3	8	10
4	10	10
5	9	10
6	10	10
7	10	9
8	9	10
Total correct tries	76	78
Total tries	80	80
Accuracy (%)	95	97.5

used in a real-time application and in a real environment. In some cases the network did not recognize the category correctly because the hand was almost horizontal or because much of the arm was recognized inside the tracking box which made the hand very small and when entering the network could not clearly recognize the characteristics.

Human-robot interaction: Once the results of the camera tests with the validation of the network have been compared, the tests with the robotic manipulator are carried out which will be controlled by means of the hand gestures performed.

The human-robot interaction to be performed consists of the collection of an object by a robotic arm when it is recognized that the user has the hand open and delivers it in the user's hand, once delivered, the robot will return to its initial position when it is recognized that the user has closed the hand as a signal that received the object. For the interaction between the user and the robotic arm, tests are done with the same 8 subjects who did the camera test. These users will make the gestures with the hand, so that, the robot performs the action corresponding to the gesture that recognizes. For this, a microcontrolled robotic arm is used which is shown in Fig. 11. For data acquisition, a webcam is used to acquire real-time images of hand gestures in a test environment.

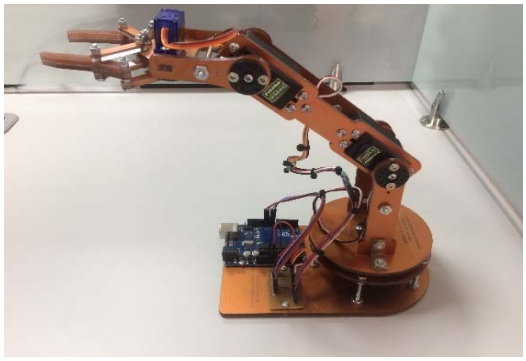

Fig. 11: Robotic arm used

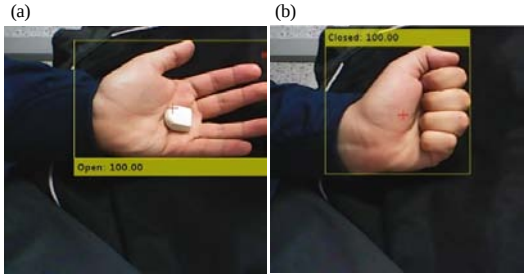

Fig. 12: a, b) Camera view for the interaction with the robotic arm

Table 6: Test results

Variables	Process 1	Process 2
Total tries	40	40
Succeed	40	40
Accuracy (%)	100	100

An example of the test environment seen by the webcam is shown in Fig. 12 where in Fig. 12a the user has received the object that the robotic arm has collected and in Fig. 12b, the user has closed the hand for the robot to continue its action.

For the tests, two robot action times are considered where the first one goes from the moment the user indicates the opening of the hand, taking the manipulator's movements from its initial position to the delivery of the object which will be called "Process 1" (Fig. 13a) and the second one goes from hand closing to the return of the manipulator to the initial position which will be called "Process 2" (Fig 13b).

For each subject, 5 repetitions of each action were performed, having a total of 40 trials where as can be seen in Table 6, the robotic arm performed the action corresponding to the gesture that the subject performed, i.e., correctly performed all the attempts for each subject, reaching an accuracy of 100%, surpassing even the percentage reached in the camera tests which the highest was 97.5%.

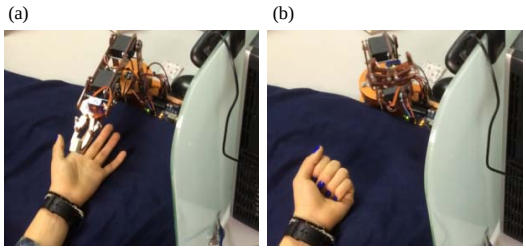

Fig. 13: a) Process 1 when the robotic arm is giving the objet to the user and b) Process 2 when the user closed the hand and the robotic arm is returning to its initial position

Table 7: Times obtained from tests performed

	Process 1		Process 2	
Subjects	CNN clasification (sec)	Robot action (sec)	CNN clasification (sec)	Robot action (sec)
1	0.028	15.043	0.030	4.521
2	0.048	14.881	0.024	4.517
3	0.020	14.706	0.027	4.462
4	0.030	15.099	0.029	4.574
5	0.028	14.763	0.026	4.458
6	0.029	14.761	0.029	4.470
7	0.027	14.836	0.033	4.482
8	0.031	14.775	0.028	4.477

It must be taken into account that in order for the interaction between the user and the robotic manipulator to be correct, the processing times of the algorithm must be considered which should not be long in other words, once the user opens or closes the hand, the robot initiates its action. To evaluate the performance of the interaction and whether the classification of the network affects the execution times of the process, "Process 1" and "Process 2" times are taken. The times were taken from the tests performed on each subject, of which the average of each is shown in Table 7.

From Table 7, it can be seen that the hand gesture classification process is carried out in an approximate average of 29 msec in general, allowing the robot to perform its tasks without delays caused by the algorithm of hand gesture recognition. On the other hand, the robot's action times were similar with differences of maximum 0.4 sec, however, these times were long for relatively short actions. For future research the human-robot interaction can be improved by improving the trajectory times.

CONCLUSION

In this research, it was presented a novel application for convolutional neural networks oriented to the human-machine interaction using an own deep CNN architecture in order to archive a high precision in the interaction for object delivery done by a robotic arm.

It is possible to restructure neural networks already pre-established in order to improve their performance for a given application as in this research. By means of two different databases, the training process of this network was validated, observing different evaluation parameters, both in the training and in the validation, a trained network with a validation accuracy of 90.2% was obtained with correct recognition of high and fair gestures between categories. Taking into account that depending on the type of application in which the neural network is to be used, the error range for the robot-machine interaction in which it is being used in this research is sufficient, since, it does not harm as such the robot's action if in some case it does not correctly recognize the gesture performed. However, it must be careful with the overfitting of the neural network to avoid that the network only recognizes very precise gestures doing that in practice if it incurs high errors and a poor human-robot interaction.

For the interaction with the robotic arm, the reaction was evaluated with respect to the gesture used by the user, observing if when the user opened the hand effectively the robot started process 1 and if the user closed the hand, it performed process 2 for which an accuracy of 100% was obtained. This allowed to observe that within the actual tests of interaction, the neural network responded efficiently, even improving the accuracy of the tests performed with only the webcam. Additionally, it should be taken into account whether the time taken by the algorithm to classify the type of gesture influenced the execution times of the tasks of the robot, obtaining that the times that the network uses in classifying the gesture of the hand did not affect the action of the robotic arm, since in making the gesture, immediately the robot initiated with the action.

The tests performed for the control of a robotic agent make it possible to verify that the use of convolutional neural networks for control and/or interaction with robots is possible, since, depending on the robustness of the training performed to the network, it is possible to command a robot only using hand gestures with high accuracy. This makes it possible to extend the use of convolutional neural networks to fields where there is human-machine interaction, for example in medical assistance robots where their function is to deliver surgical tools to the surgeon or even in vehicle control such as wheelchairs which allow the user to control it only by using signs with his hands.

ACKNOWLEDGEMENTS

Researchers are grateful to the Nueva Granada Military University which through its Vice Chancellor

for research, finances the present project with code IMP-ING-2290 and titled "Prototype of robot assistance for surgery" from which the present research is derived.

REFERENCES

Barros, P., C. Weber and S. Wermter, 2015. Emotional expression recognition with a cross-channel convolutional neural network for human-robot interaction. Proceedings of the IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), November 3-5, 2015, IEEE, Seoul, South Korea, ISBN:978-1-4799-6885-5, pp: 582-587.

Barros, P., S. Magg, C. Weber and S. Wermter, 2014. A Multichannel Convolutional Neural Network for Hand Posture Recognition. In: Artificial Neural Networks, Wermter, S., C. Weber, W. Duch, T. Honkela and P. Koprinkova-Hristova *et al.* (Eds.). Springer, Cham, Switzerland, ISBN:978-3-319-11178-0, pp: 403-410.

Guo, L., N. Li, F. Jia, Y. Lei and J. Lin, 2017. A recurrent neural network based health indicator for remaining useful life prediction of bearings. Neurocomputing, 240: 98-109.

Jin, J., A. Dundar and E. Culurciello, 2014. Flattened convolutional neural networks for feedforward acceleration. Proceedings of the 3rd International Conference on Learning Representations (ICLR'15), May 7-9, 2015, Hilton San Diego Resort & Spa, San Diego, California, pp: 1-11.

LeCun, Y., B. Boser, J.S. Denker, D. Henderson and R.E. Howard *et al.*, 1989. Backpropagation applied to handwritten zip code recognition. Neural Comput., 1: 541-551.

Poonia, P., V.K. Jain and A. Kumar, 2016. Deep learning: Review. Intl. J. Comput. Math. Sci., 5: 43-47.

Schmidhuber, J., 2015. Deep learning in neural networks: An overview. Neural Networks, 61: 85-117.

Simard, P.Y., D. Steinkraus and J.C. Platt, 2003. Best practices for convolutional neural networks applied to visual document analysis. Proceedings of the of 7th International Conference on Document Analysis and Recognition ICDAR Vol. 3, August 3-6, 2003, IEEE, New York, USA., pp: 958-962.

Simonyan, K. and A. Zisserman, 2014. Very deep convolutional networks for large-scale image recognition. Proceedings of the Conference on Learning Representations (ICLR'15), May 7-9, 2015, Hilton San Diego Resort & Spa, San Diego, California, pp: 1-14.

Singer, M.A. and S. Goldin-Meadow, 2005. Children learn when their teacher's gestures and speech differ. Psychol. Sci., 16: 85-89.

Wang, Z., Z. Li, B. Wang and H. Liu, 2016. Robot grasp detection using multimodal deep convolutional neural networks. Adv. Mech. Eng., Vol. 8,

Yu, W., K. Yang, Y. Bai, H. Yao and Y. Rui, 2015. Visualizing and comparing convolutional neural networks. Proceedings of the 3rd International Conference on Learning Representations (ICLR'15), May 7-9, 2015, Hilton San Diego Resort & Spa, San Diego, California, pp: 1-10.

Zeiler, M.D. and R. Fergus, 2014. Visualizing and Understanding Convolutional Networks. In: Computer Vision, Fleet, D., T. Pajdla, B. Schiele and T. Tuytelaars (Eds.). Springer, Cham, Switzerland, ISBN:978-3-319-10589-5, pp: 818-833.