Application of Convolutional Neural Networks for Static Hand Gestures Recognition Under Different Invariant Features

Jose L. Flores C.,E. Gladys Cutipa A.,Lauro Enciso R.,
Ing. Informática y de Sistemas
Universidad Nacional San Antonio Abad del
Cusco - UNSAAC, Perú
E-mails:{121451,efraina.cutipa,lauro.enciso}@unsaac.edu.pe;

***Abstract*—The present work proposes to recognize the static hand gestures taken under invariations features as scale, rotation, translation, illumination, noise and background. We use the alphabet of sign language of Peru (LSP). For this purpose, digital image processing techniques are used to eliminate or reduce noise, to improve the contrast under a variant illumination, to separate the hand from the background of the image and finally detect and cut the region containing the hand gesture. We use of convolutional neural networks (CNN) to classify the 24 hand gestures. Two CNN architectures were developed with different amounts of layers and parameters per layer. The tests showed that the first CNN has an accuracy of 95.37% and the second CNN has an accuracy of 96.20% in terms of recognition of the 24 static hand gestures using the database developed. We compared the two architectures developed in accuracy level for each type of invariance presented in this paper. We compared the two architectures developed and usual techniques of machine learning in results of accuracy.**

I. Introduction

Recognition of hand gestures is a natural medium used for human computer interaction (HCI), is a very active area of research in computer vision and in Machine Learning, each year receives more attention due to its multiple applications in various areas as robotics, Video games, sign language and virtual reality. However, the recognition of these gestures have some problems, due to the fact that they are based on the LSP as with all sign languages show that they are not easy to recognize under invariant features of scale, rotation, and translation; and not flexible to noise and lighting changes. For the difficulties come with the segmentation of the hand with the background of the image, is necessary to properly extract the features of the shape of the palm and fingers. To solve this problem we make use of the techniques of digital image processing. Other research uses colored gloves, markers or electronic gloves to improve the detection of the hand gesture [1] [2] [3] . Nowadays, the investigation is based on the detection of naked hand gestures, without the use of colored gloves or markers [4].

In the literature of hand gesture recognition, there are research works that use different extractors of characteristics that are robust to invariations like scaling, rotation, translation, as well as changes in lighting, complex backgrounds or noise. Principal component analysis [5], zernike moments [6], gabor filters [7], histogram oriented gradients [8] , local binary patterns [9], scale invariant feature transform (SIFT) and speeded up robust features (SURF) [4] [10], Fourier descriptors [11] and others.

The CNNs were created by Yann Lecun [12], According to the state of the art, the last few years deep learning and CNNs has led to very good performance on a variety of problems, such as visual recognition, speech recognition and natural language processing [13]. Especially in the face recognition [14]. CNN can achieve extreme accuracy in image classification, because this model makes use of convolution and subsampling layers to extract the most relevant features of the image, these steps prevent images with invariant features affect the performance of the classifier. Training CNN can take a lot of time , from days to weeks, the use of GPU acceleration can significantly streamline this process, shorting training time for days by hour image classification problem [13].

The paper is organized as follows. Hand gesture detection is presented in Section II. In Section III, the CNNs are trained with the database developed. In Section IV we show the results of each CNN trained by each invariant feature. Finally, concluding remarks are provided in Section V

II. Hand gesture Detection

The present work proposes a method whose sequence of steps is shown in Figure 1

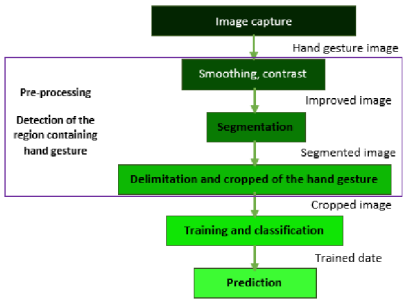

Fig. 1: Proposed method

978-1-5090-6363-5/17/$31.00

A. Capturing Images

The database presents photos of hand gestures based on the alphabet of LSP of a total of 25 people. Generally, for the hand identification process it should not be complex if the image was taken with a simple background and regular illumination, as seen in figure 2. However, images taken with complex backgrounds and varied illumination were used with invariant features. The database consists of 16200 images for the training set and 3240 images for testing, each image was taken with a size of 64x64, are .jpg format. The developed database was called "Hand_Gesture_Dataset_LSP", the Fig 2 show the dataset developed.

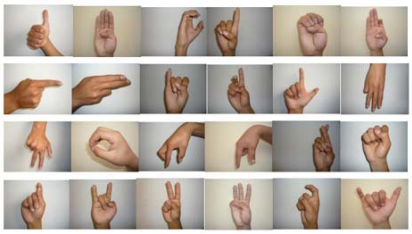

Fig. 2: Images of hand gestures developed based on the alphabet of the LSP

Table I show the distribution of amount for differents invariations for train and test data.

TABLE I: Amount dates for invariation.

Type	Data Number	
	Train	Test
Scaling	3000	600
Rotation	3000	600
Traslation	3000	600
Illumination	3000	600
Noise	3000	600
Background complex	1200	240
Total	16200	3240

B. Pre Processing

1) Smoothing and contrast enhancement: To improve the image, we proceed to reduce impulsive noise (salt and pepper), making a smoothing based on the median filter, this type of nonlinear filter is widely used in digital image processing because it works very well with impulsive noises on the other hand reduces noise levels without blurring the edges. To improve the contrast of the image, the image is normalized. Once normalized a color image with low illumination as seen in the Fig 3 dates for invariation, the image is segmented.

Fig. 3: Original and standard image taken with low illumination

2) Segmentation: In order to segment the image, segmentation based on a threshold value was used; this type of segmentation consists of setting thresholds (limits) so that the pixels between each pair of limits are points of the object, part of the background. The image works in an RGB space (red, green and blue); this space has a problem when recognizing a color, in an environment of varied luminosity because color can have several tones. Therefore, to recognize the same color with different intensities gives the problem of creating a range filter when it comes of locating the color of the skin, for this case uses a well-known and used color space YCbCr [15] that offers better performance under different lighting conditions. After separating the hand gesture from the background of the image, the separation is represented as a binary image, and it is considered as white pixel, if the pixel of the image in the space Ycbcr is a pixel of the skin, otherwise is considered as black pixel. As shown in the following equation:

* If ($80 < P_{Yi} < 255$ and $80 < P_{Cbi} <$
* 135 and $135 < P_{Cri} < 180$)
* return p_{white}
* Otherwise
* return p_{black}

After segmenting as shown in Fig 4, the noise caused by the segmentation was reduced, the median filter was used, the noise reduced, it is observed that there are pixels in White that do not belong to the hand, are called false positives, the black pixels inside the hand, are called false negatives, to minimize these false positives and negative morphological operations of closure and dilation.

Fig. 4: Normalized and segmented image taken with low illumination

3) Delimitation and extraction of the object of interest: A search was made with a rectangular region of all possible objects that could be the object of interest (hand gesture), the largest rectangular region is selected, this step helps to retain a single object of the image, and this selected object always turns out to be the gesture of the hand. Thereafter, the image is drawn into the selected rectangular region and clipped as shown in Fig 5. Finally, the cropped image is saved and a new database called "Hand_Gesture_Dataset_LSP_Processing".

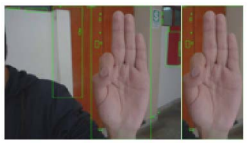

Fig. 5: Detection and cropping of the gesture of the hand or object of interest

III. Training of Networks

The networks are trained with the database "Hand_Gesture_Dataset_LSP_Processing". The database has 16200 processed images of 64 x 64 pixels each image. Two deep convolutional neural network architectures are developed, each convolutional neural network receives as input 4096 neurons. All networks were trained and simulated with a Quadro K620 GPU, using the Python programming language and using libraries such as keras, scikit-learn and others. The first CNN was based on the LeNET-5 architecture and the second CNN was proposed.
In this work two convolutional networks (CNNs) with different depths and parameters are presented.

A. CNNs architectures

1) CNN1: First architecture developed is based on the LeNet-5 shown on Fig 6.

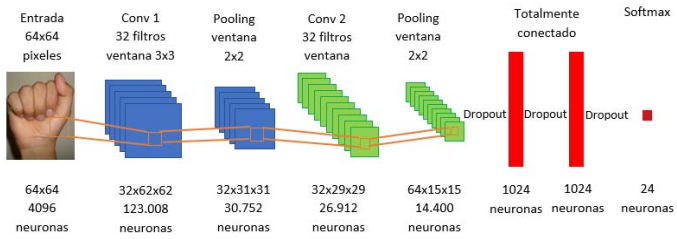

Fig. 6: First Developed Architecture of CNN

Following the input layer, there are 2 convolutional layers, Conv1 and Conv2 with activation functions ReLU, and each accompanied by a layer of Pooling. Table II presents the configuration of CNN1.

TABLE II: CNN1 configuration

Convolution 32 filters, 3x3 kernel and ReLU
Max pooling 2x2 kernel
Convolution 32 filters, 3x3 kernel and ReLU
Max pooling 2x2 kernel
Dropout 50%
Fully connected with 1024 neurons
Dropout 50%
Fully connected with 1024 neurons
Dropout 50%
Softmax 24 classes

2) CNN2: Second architecture developed proposed we show on the following scheme of Fig 7.

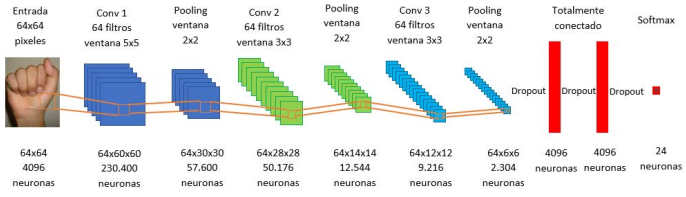

Fig. 7: Second Developed Architecture of CNN

Following the input layer, there are 3 convolutional layers: Conv1, Conv2 and Conv3 with activation functions ReLU, and each accompanied by a layer of Pooling. Table III presents the configuration of CNN2.

TABLE III: CNN2 configuration

Convolution 64 filters, 5x5 kernel and ReLU
Max pooling 2x2 kernel
Convolution 64 filters, 3x3 kernel and ReLU
Max pooling 2x2 kernel
Convolution 64 filters, 3x3 kernel and ReLU
Max pooling 2x2 kernel
Dropout 50%
Fully connected with 4096 neurons
Dropout 50%
Fully connected with 4096 neurons
Dropout 50%
Softmax 24 classes

B. Training CNN1 and CNN2

The table IV shows CNN1 and CNN2 training parameters, CNN training process of the system is based on the combination of the backpropagation algorithm with the stochastic gradient descent method. The cost function chosen is the cross entropy loss, one standard in the design of CNNs.

TABLE IV: CNN training parameters

Network	CNN1	CNN2
Number of training samples	16200	16200
Activation function	ReLU-Softmax	ReLU-Softmax
Learning rate (n)	0.01	0.01
Iterations	400	400
Cost function	cross entropy	cross entropy
Optimization	SGD	SGD

IV. Results

1) CNN1: Fig 8 shows the precision obtained in both subsets approximately 100% for the training and 99.34% for the validation. For the set of test images, the precision obtained with the configuration defined in the previous section is 95,37%. With respect to the error function, its evolution is shown based on 400 iterations.

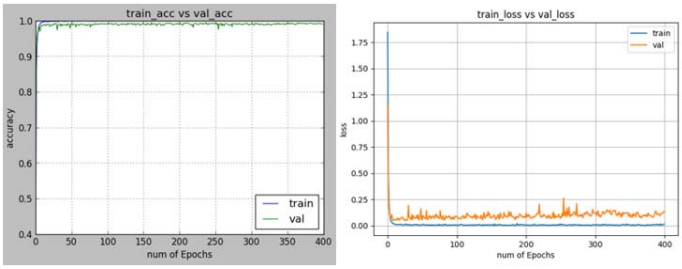

Fig. 8: Accuracy and loss training process of CNN1 between training and validation data

2) CNN2: Fig 9 shows the accuracy obtained in both subsets approximately 100% for the training and 99 .73% for the validation. For the set of test images, the precision

obtained with the configuration defined in the previous section is 96,20%. With respect to the error function, its evolution is shown based on 400 iterations.

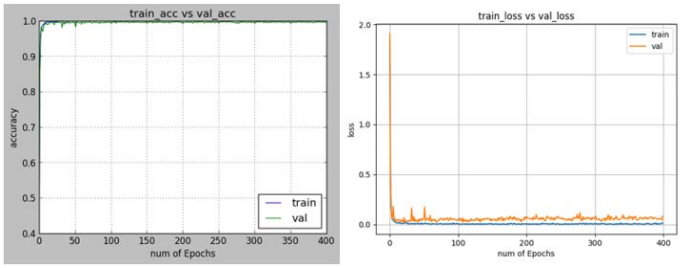

Fig. 9: Accuracy and loss training process of CNN2 between training and validation data

Table V shows the comparison between test accuracy for invariants features present for each CNN developed.

TABLE V: Comparison of test accuracy for features invariants.

Type	Test Accuracy	
	CNN1	CNN2
Scaling	87.33	93.67
Rotation	94.00	94.83
Traslation	91.00	91.66
Illumination	92.67	93.00
Noise	89.00	89.33
Background Complex	84.44	85.34
Total	95.37	96.20

Table VI shows the comparison between the CNNs developed with usual techniques of machine learning

TABLE VI: Comparison techniques of machine learning with CNNs

Method	Accuracy
SVM	85.12
Fourier Descriptor + SVM	87.14
Local binary pattern + SVM	89.14
K Nearest Neighbors	89.40
Random forest	90.83
CNN1	95.37
CNN2	96.20

V. Conclusion

There is a difficulty in finding systems that recognize static hand gestures with great precision for different types of lighting, noise and complex backgrounds, as well as invariations of scale, rotation and translation. However, CNNs were used as a solution to improve the rate of recognition accuracy for these types of invariance. It was proposed to develop two convolutional networks, each with a different number of layers, depth and number of parameters per layer, comparing the results of precision and error obtained. Networks with greater depth or number of hidden layers have greater recognition accuracy compared to networks with fewer hidden layers. The CNNs shown better results in comparison to the usual techniques of machine learning. A good use of digital image processing techniques will help to detect a better the region that contains the hand gesture, minimizing the error caused by complex image background and varied illumination. Complex background images obtain low accuracy compared to other invariations, skin color based segmentation does not provide optimum separation compared to detection techniques such as Viola Jones [4]. However this segmentation is most commonly used for its simplicity to separate interest object from complex background.

References

[1] F. Parvini and C. Shahabi, "An algorithmic approach for static and dynamic gesture recognition utilising mechanical and biomechanical characteristics," *Int. J. Bioinformatics Res. Appl.*, vol. 3, no. 1, pp. 4–23, Dec. 2007. [Online]. Available: http://dx.doi.org/10.1504/IJBRA.2007.011832

[2] R. Y. Wang and J. Popović, "Real-time hand-tracking with a color glove," in *ACM SIGGRAPH 2009 Papers*, ser. SIGGRAPH '09. New York, NY, USA: ACM, 2009, pp. 63:1–63:8. [Online]. Available: http://doi.acm.org/10.1145/1576246.1531369

[3] N. Y. Y. Kevin, S. Ranganath, and D. Ghosh, "Trajectory modeling in gesture recognition using cybergloves reg; and magnetic trackers," in *2004 IEEE Region 10 Conference TENCON 2004.*, vol. A, Nov 2004, pp. 571–574 Vol. 1.

[4] L. Yun and Z. Peng, "An automatic hand gesture recognition system based on viola-jones method and svms," in *2009 Second International Workshop on Computer Science and Engineering*, vol. 2, Oct 2009, pp. 72–76.

[5] T. N. T. Huong, T. V. Huu, T. L. Xuan, and S. V. Van, "Static hand gesture recognition for vietnamese sign language (vsl) using principle components analysis," in *2015 International Conference on Communications, Management and Telecommunications (ComManTel)*, Dec 2015, pp. 138–141.

[6] M. A. Aowal, A. S. Zaman, S. M. M. Rahman, and D. Hatzinakos, "Static hand gesture recognition using discriminative 2d zernike moments," in *TENCON 2014 - 2014 IEEE Region 10 Conference*, Oct 2014, pp. 1–5.

[7] M. A. Amin and H. Yan, "Sign language finger alphabet recognition from gabor-pca representation of hand gestures," in *2007 International Conference on Machine Learning and Cybernetics*, vol. 4, Aug 2007, pp. 2218–2223.

[8] K. p. Feng and F. Yuan, "Static hand gesture recognition based on hog characters and support vector machines," in *2013 2nd International Symposium on Instrumentation and Measurement, Sensor Network and Automation (IMSNA)*, Dec 2013, pp. 936–938.

[9] Y. Ding, H. Pang, X. Wu, and J. Lan, "Recognition of hand-gestures using improved local binary pattern," in *2011 International Conference on Multimedia Technology*, July 2011, pp. 3171–3174.

[10] M. Murugeswari and S. Veluchamy, "Hand gesture recognition system for real-time application," in *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, May 2014, pp. 1220–1225.

[11] H. M. Gamal, H. M. Abdul-Kader, and E. A. Sallam, "Hand gesture recognition using fourier descriptors," in *2013 8th International Conference on Computer Engineering Systems (ICCES)*, Nov 2013, pp. 274–279.

[12] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov 1998.

[13] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, and G. Wang, "Recent advances in convolutional neural networks," *CoRR*, vol. abs/1512.07108, 2015. [Online]. Available: http://arxiv.org/abs/1512.07108

[14] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: a convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, Jan 1997.

[15] K. B. Shaik, G. P., V. Kalist, B. S. Sathish, and J. M. M. Jenitha, "Comparative study of skin color detection and segmentation in hsv and ycbcr color space," in *3rd International Conference on Recent Trends in Computing 2015*, Jul 2015, pp. 41–48.