

# Hand Gesture Recognition using Neural Networks

**Abstract**—Visual Interpretation of gestures can be useful in accomplishing natural Human Computer Interactions (HCI). In this paper we proposed a method for recognizing hand gestures. We have designed a system which can identify specific hand gestures and use them to convey information. At any time, a user can exhibit his/her hand doing a specific gesture in front of a web camera linked to a computer. Firstly, we captured the hand gesture of a user and stored it on disk. Then we read those videos captured one by one, converted them to binary images and created 3D Euclidian Space of binary values. We have used supervised feed-forward neural net based training and back propagation algorithm for classifying hand gestures into ten categories: hand pointing up, pointing down, pointing left, pointing right and pointing front and number of fingers user was showing. We could achieve up to 89% correct results on a typical test set.

**Keywords:** *Human computer Interaction, Hand Gesture Recognition, Computer Vision.*

## I. INTRODUCTION

With the development of information technology in our society, one can expect that computer systems to a larger extent will be embedded into our daily life. These environments will impose needs for new types of human-computer-interaction, with interfaces that are natural and easy to use. Gesture recognition has the potential to be a natural and powerful tool supporting efficient and intuitive interaction between the human and the computer.

Every day new applications and devices are becoming part and parcel of our life, but the means of communication with these are at moment limited to input devices such as mouse, keyboards, joysticks and track balls etc. These input devices have grown to be familiar, but they inherently limit the speed and naturalness of interaction with the computers.

The use of hand gestures provides an attractive alternative to cumbersome interface devices for human-computer interaction (HCI). User's generally use hand gestures for expression of their feelings and notifications of their thoughts. In particular, visual interpretation of hand gestures can help in achieving the ease and naturalness desired for HCI. Recent researches [1-4] in computer vision have established the importance of gesture recognition systems for the purpose of human computer interaction. An extensive survey on Gesture Recognition is presented in [5].

The primary goal of hand gesture recognition research is to create a system which can identify specific hand gestures and

use them to convey information or for device control. The main approaches for analyzing and classifying hand gestures for HCI include Glove based techniques and Vision based techniques. The data gloves method is quite expensive and forces the user to carry a load of cables which are connected to the computer and hinders the ease and naturalness of the user interaction.

Computer vision based techniques are non invasive. These are based on the way human beings perceive information about their surroundings. Although it is difficult to design a vision based interface for generic usage, yet it is feasible to design such an interface for a controlled environment [6].

## II. RELATED WORKS

The first gestures that were applied to computer interactions date back to the PhD work of Ivan Sutherland [7], who demonstrated Sketchpad, an early form of stroke-based gestures using a light pen to manipulate graphical objects on a tablet display. This form of gesturing has since received widespread acceptance in the human-computer interaction (HCI) community.

A vision-based system able to recognize 14 gestures in real time to manipulate windows and objects within a graphical interface was developed by C.W. Ng et al. in [8]. Abe et al. [9] proposed a system which recognizes hand gestures through the detection of the bending of the hand's five fingers, based on image-property analysis.

Hasanuzzaman et al. [10] presented a real-time hand gesture recognition system using skin color segmentation and multiple-feature based template-matching techniques. In their method, the three largest skin-like regions are segmented from the input images by skin color segmentation technique from YIQ color space and they are compared for feature based template matching using a combination of two features: correlation coefficient and minimum (Manhattan distance) distance qualifier. In their experiment, they have recognized ten gestures out of which two are dynamic facial gestures. These Gesture commands are being sent to their pet robots AIBO through Software Platform for Agent and Knowledge Management (SPAK) [11] and their actions are being accomplished according to users pre defined action for that gesture.

In the work of Franklin et al. [12], a robot waiter is controlled by hand gestures using the Perseus architecture for gesture recognition. In the work of Cipolla et al. [13], a gesture-based interface for robot guidance is based on un-calibrated stereo vision and active contours. The research of Guo et al. [14] discusses vehicles controlled by hand gestures,

based on color segmentation. Waldherr et al. [15] proposed a vision-based interface that instructs a mobile robot using pose and motion gestures in an adaptive dual-color tracking algorithm.

Nielsen et al. [16] proposed a real time vision system which uses a fast segmentation process to obtain the moving hand from the whole image and a recognition process that identifies the hand posture from the temporal sequence of segmented hands. The system's visual memory stores all the recognizable postures, their distance transform, their edge map and morphologic information. They have used Hausdorff distance approach for robust shape comparison. Their system recognitions 26 hand postures and achieved a 90% recognition average rate.

Yang et al. [17] employed two-dimensional (2-D) motion trajectories, and a time-delay neural network to recognize 40 dynamic American Sign Language (ASL) gestures. Yin and Xie [18] created a fast and robust system that segments using color and recognizes hand gestures for human-robot interaction using a neural network. A system intended to be particularly robust, in real world environments, is presented by Triesch and Malsburg [19]. The strength of the recognition is based on color features extracted from a cadre of training images, and an elastic-graph-match approach.

Alon et al. [20] applied DSTW to the simultaneous localization and recognition of dynamic hand gestures. They implemented a hand-signed digit recognition system and their algorithm can recognize gestures using a fairly simple hand detection module that yields multiple candidates. The system does not break down in the presence of a cluttered background, multiple moving objects, multiple skin-colored image regions, and users wearing short sleeves shirts.

Ramamoorthy et al. [21] used HMM based real time dynamic gesture recognition system which uses both the temporal and shape characteristics of the gesture for recognition. Use of both hand shape and motion pattern is a novel feature of this work.

Chen et al. [22] proposed a hand gesture recognition system to recognize continuous gestures before stationary background. The system consists of four modules: real time hand tracking and extraction, feature extraction, Hidden Markov model (HMM) training, and gesture recognition.

Yin Xiaoming et al. [23] used an RCE neural network based color segmentation algorithm for hand segmentation, extract edge points of fingers as points of interest and match them based on the topological features of the hand, such as the center of the palm.

Xiong et al. [24] investigated the utility of motion symmetries of a speaker's hands when they are both engaged in communication. For computing hand motion symmetries they have used an approach based on the correlation computations by deploying a two-stage algorithm of window-based correlation and 'hole-filling'. Local symmetries are detected using a windowing operation and demonstrated that the selection of a smaller window size results in better sensitivity to local symmetries at the expense of noise in the form of spurious symmetries and 'symmetry drop offs'.

New. J.R. et al. [25] presented a real-time gesture recognition system which can track hand movement, define orientation, and determine the number of fingers being held up in order to allow control of an underlying application.

### III. OVERVIEW OF PROPOSED WORK

The aim of this paper is to present a model, based on pattern recognition techniques using supervised feed-forward neural net training and back propagation algorithm for classifying hand gestures into ten categories: hand pointing up, pointing down, pointing left, pointing right and pointing front and also to count the number of fingers user was showing. We have applied a simple pattern recognition technique to the problem of hand gesture recognition. This method has a training phase and a testing phase. In the training phase, the user shows hand gestures which were captured using Image Acquisition Toolbox of MATLAB 7.01 and USB based Frontech e-cam camera.

The block diagram of our proposed system is shown in figure 1. Here we have shown the recognition of number of fingers the user is showing; in the same manner we have also trained our system for recognizing the user hand direction.

All videos are captured at 30 FPS in AVI format and saved to disk. We have implemented the method, using an ordinary workstation with no special hardware beyond a web camera. Our proposed system involves following steps.

Step 1: First we captured the background comprising of one frame only as a static image with out the user hands. We have taken two set of videos of 50 frames each from different users, first set consisting of five direction gestures (top, down, front, left, right) and in the second set we have captured five finger count gestures (one-finger, two-finger, three-finger, four finger and , five-finger). In finger counting we are not distinguishing between the finger and the thumb. For finger count gestures, user has to pose the hand in horizontal or vertical manner only.

Step 2: Using "frame2im" function of MATLAB we have extracted frames from videos. As we have kept uniform background, it is possible to build a new image that corresponds to the difference between the current image of hand and the background. For this we have calculated their absolute difference using "imabsdiff" function of MATLAB and the resultant image contains only the hand of the user.

Step 3: Each image extracted from a video is an uncompressed RGB image. We then converted the RGB image into grayscale by using "rgb2gray" function of MATLAB which eliminates the hue and saturation information while retaining the luminance factor and then to binary image using an experimental threshold value. The noise from the binary image is then removed using MATLAB function "bwmorph ( )" that erodes and dilates the noisy picture.

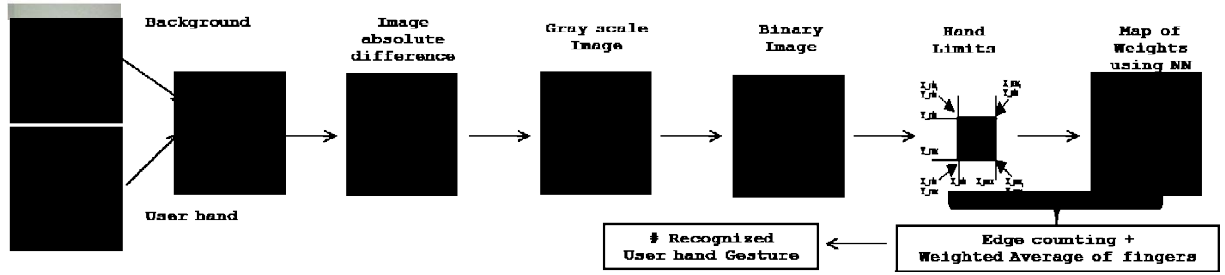


Figure 1: Block diagram of our proposed approach.

Step 4: As our focus is on the palm area of hand and fingers, we have made a MATLAB function which takes binary image from step 3 as input and returns the binary image with the hand portion containing fingers only. This function also computes four extreme coordinates of hand (bounding box) by calculating the sum of elements in each row and columns. To find the extreme coordinates of the fingers in three sides is easy but to find the starting of hand portion from arm side is quite difficult, because the user can place his hand any where with in the visibility of camera. To overcome this we have placed a restriction on the user that he/she has to display the thumb compulsory in the gesture and to find the thumb of the user in the image we have created a line vector which receives the number of edge pixels from the arm side as shown in the figure 2 below.

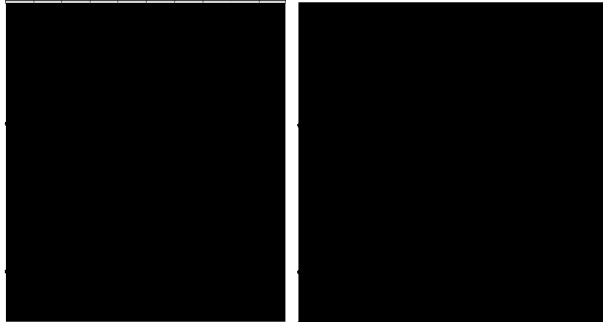


Figure 2: Calculating line of edge pixels of hand images.

When we start counting the edge pixels of the hand from left, when there is the arm, the total number of edge pixels is two. When we overpass the thumb, the number of edge pixels will become four. Once the thumb is found, we came to know that from this position the hand portion of the user was starting. As the user is not supposed to place his/her hand at the same distance every time we will have images of hands in very different sizes i.e. if the user is close to camera the hand will occupy a large part of the image and the hand will appear small

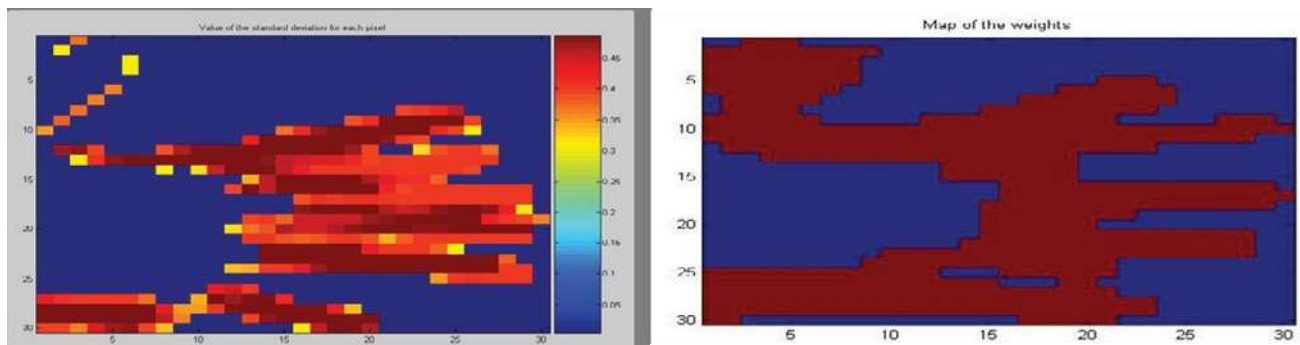
if he is far away from the camera. To overcome this problem, after some tests and measurements we have decided that an image of 30 by 30 will be large enough to contain any given hand image without any loss of information. We have resized all the images to a standard size of 30 by 30 using “imresize” function of MATLAB which uses nearest-neighbor interpolation.

Now we have a 3D matrix with size 30 x 30 x 5N where N is the number of examples of each sign. As the Neural Network usually takes a vector as input we converted the 3D matrix into 2D input vector by using the MATLAB function “reshape” which reshapes the array. To decrease the amount of massive calculus, we have removed the pixels whose standard deviation is equal to 0; these pixels are irrelevant for the classification. This was achieved by using “std” function which calculates the standard deviation of the elements of each line of the input vector matrix.

Step 5: Now we have input images ready for training. We have used Neural Networks as they are most suitable solution for image recognition or sign classification. We have chosen a Neural network with 750 neurons in the in input layer, 7 neurons in the single hidden layer, and 5 neurons in the output layer for training.

Once training was over, we have build the map of the weights and found the weighted average of the relevant area in which the fingers exists i.e. in between the 15-25 columns as per figure 3. Using this weighted average realized from NN training, edge pixels counted in the step 4 and with equation (1) given below we have recognized the number of fingers user is displaying. In case of directional gestures we have followed the same procedure up to step 4 and here, whatever image that is being presented to us will be rotated to align with image of hand pointing left and then on the basis the coordinates of bounding boxes (X\_min, X\_max, Y\_min and Y\_max) are used for finding the direction of user hand.

$$WA = \sum_{column=15}^{column=25} \left[ number\_edges(column) * \left( \sum_{line=1}^{line=30} pixel(line, column) \right) \right] \quad (1)$$



## IV. EXPERIMENTS AND DISCUSSIONS

We have captured and saved all videos in AVI format using USB based Frontech e cam camera (JIL 2220) and Image Acquisition Toolbox of MATLAB 7.01 with 320×240 resolution size of images. All videos were taken against a uniform background under normal illumination. The idea is to make the classification of gestures as key problem, not the segmentation. The hand was the only object in the scene and no restrictions were imposed on the user about distance of his hand from the camera, except that user hand should be with in the view of camera and user should display his/her thumb in the gesture.

The proposed method was tested on five different users showing ten gestures such as one, two, three, four, five fingers and pointing gestures such as hand pointing up, pointing down, pointing left, pointing right and pointing front. The system can recognize finger counting gestures regardless of direction of the gestures. In the welcome screen of our system the user was asked to select an option, whether the user want to find the number of fingers or direction of the user hand. In case of fingers counting the user has to select horizontal or vertical hand position. Fig. 4 shows the ten gestures we have used in our system. Fig. 5 shows sample results for hand images displaying “five fingers in horizontal direction”, “three fingers in vertical direction”, “pointing right” & “pointing down” from our gesture recognition system.

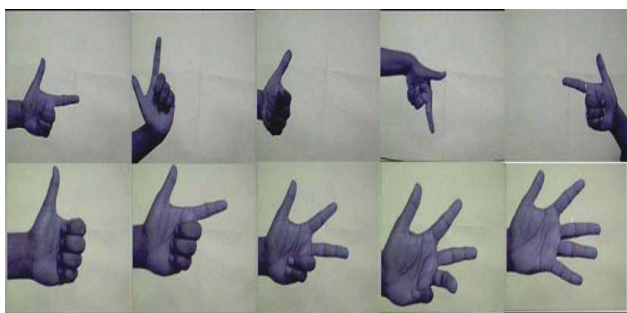


Figure 5: Out put screens of our proposed Hand gesture recognition system.

Table 1 shows the recognition results from ten input images of five different hand gestures. The numbers in the table represent each hand gesture as shown in fig. 4. We have achieved 89 % recognition rate with our captured data. Due to occlusion of fingers the recognition rate of gesture 3 (hand pointing front) and gesture 9 (User showing four fingers) is low. False positives occur due to similar hand gestures performed differently by the different users.



Table 1: % Recognition rate of hand gestures using the proposed method for 10 Gestures and 5 users

<b>Gesture User</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>	<b>%</b>
<b># 1</b>	<b>10/10 (100%)</b>	<b>9/10 (90%)</b>	<b>7/10 (70%)</b>	<b>9/10 (90%)</b>	<b>8/10 (80%)</b>	<b>8/10 (80%)</b>	<b>9/10 (90%)</b>	<b>8/10 (80%)</b>	<b>7/10 (70%)</b>	<b>10/10 (100%)</b>	<b>85%</b>
<b># 2</b>	<b>10/10 (100%)</b>	<b>10/10 (100%)</b>	<b>8/10 (80%)</b>	<b>10/10 (100%)</b>	<b>10/10 (100%)</b>	<b>9/10 (90%)</b>	<b>9/10 (90%)</b>	<b>10/10 (100%)</b>	<b>8/10 (80%)</b>	<b>10/10 (100%)</b>	<b>94%</b>
<b># 3</b>	<b>10/10 (100%)</b>	<b>8/10 (80%)</b>	<b>8/10 (80%)</b>	<b>10/10 (100%)</b>	<b>8/10 (80%)</b>	<b>9/10 (90%)</b>	<b>10/10 (100%)</b>	<b>9/10 (90%)</b>	<b>6/10 (60%)</b>	<b>10/10 (100%)</b>	<b>88%</b>
<b># 4</b>	<b>10/10 (100%)</b>	<b>10/10 (100%)</b>	<b>8/10 (80%)</b>	<b>9/10 (90%)</b>	<b>8/10 (80%)</b>	<b>10/10 (100%)</b>	<b>10/10 (100%)</b>	<b>9/10 (90%)</b>	<b>7/10 (70%)</b>	<b>10/10 (100%)</b>	<b>91%</b>
<b># 5</b>	<b>10/10 (100%)</b>	<b>8/10 (80%)</b>	<b>7/10 (70%)</b>	<b>8/10 (80%)</b>	<b>9/10 (90%)</b>	<b>10/10 (100%)</b>	<b>10/10 (100%)</b>	<b>10/10 (100%)</b>	<b>6/10 (60%)</b>	<b>10/10 (100%)</b>	<b>89%</b>
<b>Total</b>	<b>100%</b>	<b>90%</b>	<b>76%</b>	<b>92%</b>	<b>86%</b>	<b>92%</b>	<b>96%</b>	<b>92%</b>	<b>68%</b>	<b>100%</b>	

## V. CONCLUSION

We have proposed a vision based gesture recognition using a simple system connected with a web camera. To detect hand, count fingers and find the direction in which user is pointing we have trained the neural network. We have used neural network functions such as trainidx, learnidx, trains and logsig in our training. The hand of the user was separated from the background, the fingers region was focused and by using the weighted average of fingers region learned through NN training and edge counting method as discussed in step four, our system recognized the number of fingers user was displaying. At the same time the direction in which the user's hand was pointing is also recognized. We have experimented with gesture images captured by a camera and achieved satisfactory results. In the future we are trying to detect moving hand with more features and to recognize the complex gestures.

## REFERENCES

- [1] Pavlovic, V., Sharma, R. & Huang, T.S.: Visual interpretation of hand gestures for human-computer interaction: A review. IEEE Transaction on Pattern Analysis and Machine Intelligence, 19(7), pp 677–695, 1997.
- [2] Wu, Y. & Huang, T.S.: Vision-based gesture recognition: A review. In Lecture Notes in Computer Science, Gesture Workshop, 1999.
- [3] Konstantinos G. Derpanis.: A Review of Vision-Based Hand Gestures. Internal Report, Department of Computer Science, York University, 2004.
- [4] Watson, Richard.: A Survey of Gesture Recognition Techniques. Technical Report TCD-CS-93-11, Department of Computer Science, Trinity College Dublin, 1993.
- [5] Sushmita Mitra and Tinku Acharya: Gesture Recognition: A Survey. IEEE Transactions on Systems, Man and Cybernetics - Part C: Applications and Reviews, Vol. 37, No. 3, 2007.
- [6] T.S. Hunang and V.I. Pavloic: Hand Gesture Modeling, Analysis, and Synthesis. Proc. of International Workshop on Automatic Face-and gesture recognition, Zurich, pp. 73-79, 1995
- [7] Sutherland, I. E. Sketchpad: A man-machine graphical communication system. In: Proceedings of the AFIPS Spring Joint Computer Conference 23. pp. 329–346, 1963.
- [8] C. W. Ng and S. Ranganath.: Real-time gesture recognition system and application. Image Vis. Comput., vol. 20, no. 13–14, pp. 993–1007, 2002
- [9] K. Abe, H. Saito, and S. Ozawa.: Virtual 3-D interface system via hand motion recognition from two cameras. IEEE Trans. Syst., Man, Cybern. A, vol. 32, no. 4, pp. 536–540, 2002
- [10] Md. Hasanuzzaman, V. Ampornaramveth, Tao Zhang, M.A. Bhuiyan, Y. Shirai and H. Ueno.: Real-time Vision-based Gesture Recognition for Human Robot Interaction. In the Proceedings of the IEEE International Conference on Robotics and Biomimetics, Shenyang China, 2004
- [11] Vuthichai Ampornaramveth and Haruki Ueno.: Software Platform for Symbiotic Operations of Human and Networked Robots. NII Journal, Vol. 3, pp 73-81, 2001
- [12] D. Franklin, R. E. Kahn, M. J. Swain, and R. J. Firby: Happy patrons make better tippers-Creating a robot waiter using Perseus and the animate agent architecture. In Proc. Int. Conf. Automatic Face and Gesture Recognition, Killington, VT, pp. 14–16, 1996
- [13] R. Cipolla and N. J. Hollinghurst.: Human-robot interface by pointing with un-calibrated stereo vision. Image Vis. Computing., vol. 14, no. 3, pp. 171–178, 1996
- [14] D. Guo, Y. H. Yan, and M. Xie.: Vision-based hand gesture recognition for human-vehicle interaction. In 5th Int. Conf. Control, Automation, Robotics and Vision, Singapore, pp. 151–155, 1998
- [15] S. Waldherr, R. Romero, and S. Thrun.: A gesture-based interface for human-robot interaction. Auton. Robots, vol. 9, no. 2, pp. 151–173, 2000
- [16] Elena Sanchez-Nielsen, Luis Anton-Canalis and, Mario Hernandez-Tejera.: Hand Gesture Recognition for Human-Machine Interaction. Journal of WSCG, Vol. 12, No. 1-3, Plzen, Czech Republic, 2003
- [17] M. H. Yang, N. Ahuja, and M. Tabb.: Extraction of 2-D motion trajectories and its application to hand gesture recognition. IEEE Trans. PAMI. vol. 29, no. 8, pp. 1062–1074, 2002
- [18] X. Yin and M. Xie.: Finger identification in hand gesture based human-robot interaction. J. Robot. and Auton. Syst., vol. 34, no. 4, pp. 235–250, 2001
- [19] J. Triesch and C. von der Malsburg: A system for person-independent hand posture recognition against complex backgrounds. IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 12, pp. 1449–1453, 2001
- [20] Jonathan Alon, Vassilis Athitsos, Quan Yuan, and Stan Sclaroff.: Simultaneous Localization and Recognition of Dynamic Hand Gestures. Proc. IEEE Workshop on Motion and Video Computing, 2005
- [21] Aditya Ramamoorthy, Namrata Vaswani, Santanu Chaudhury, Subhashis Banerjee.: “Recognition of dynamic hand gestures”. Pattern Recognition (36), pp 2069–2081, 2003.
- [22] F. S. Chen, C.M. Fu and C.L. Huang.: Hand gesture recognition using a real-time tracking method and hidden Markov models. Image Vision Computer, pp. 745–758, 2003
- [23] Y. Xiaoming and X. Ming.: Estimation of the fundamental matrix from uncalibrated stereo hand images for 3D hand gesture recognition”, Pattern Recognition, vol. 36, pp 567–584, 2003.
- [24] Yingen Xiong, Francis Quek & David McNeill.: Hand Gesture Symmetric Behavior Detection and Analysis in Natural Conversation. In the Proceedings of the Fourth IEEE Int. Conference on Multimodal Interfaces, 2002.
- [25] New, J.R., Hasanbelliu, E. and Aguilar, M.: Facilitating User Interaction with Complex Systems via Hand Gesture Recognition. In Proceedings of the 2003 Southeastern ACM Conference, Savannah, GA., 2003.