Scientific Journal of Impact Factor (SJIF): 3.134

e-ISSN(O): 2348-4470
p-ISSN(P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 2,Issue 5, May -2015

Static Hand Gesture Recognition using Freeman Chain Code and Neural Network

Nihar J. Randive[1],Dr. Darshak G. Thakore[2], Dr. Narendra.M.Patel[3]

[1]*Student (M.E.), Department of Computer Engineering, Birla Vishvakarma Mahavidyalaya, Vallabh Vidyanagar, India,*
[2]*Head Of Department, Department of Computer Engineering, Birla Vishvakarma Mahavidyalaya, Vallabh Vidyanagar, India, dgthakore@bvmengineering.ac.in*
[3]*Associate Professor, Department of Computer Engineering, Birla Vishvakarma Mahavidyalaya,Vallabh Vidyanagar, India, nmpatel@bvmengineering.ac.in*

Abstract- *In recent years, much effort is done to provide better way to communicate with other systems. One of them is to provide interactive way to give commands to other systems. The best way to do so is using our hands to give commands. In order to do so, computer system needs to understand the hand signals. The field of image processing gives us the capability to achieve this goal. In past years many research was done to create such system. In this paper, we discuss various aspects of hand gesture recognition along with the limitations and remedies to overcome them. The purpose of this paper is to develop a way that overcomes some limitations and increase the efficiency. However many research is done to achieve this goal yet have some limitations. We analyze these research in literature survey and address to them. We have detected hand based on skin pixels. Then hand contour is created from which hand signature (chain code) is generated. Based on this signature, hand gesture is recognized using Neural Network.*

Keywords-*YCbCr; chain coding; chain code histogram; neural network*

I. INTRODUCTION

As in current era, computers have become necessary in society, for human– computer interaction (HCI), it will have a positive impact on its use. So, the interest is growing for development of new techniques to bridge human–computer barrier. Gestures are considered better as an interaction technique that can deliver easier, natural methods for communicating with our computers [2].

The human gesture recognition has gained much importance in computer-vision field such as visual surveillance, video retrieval, sports video analysis, human- computer interactions, Sign Language Recognition (Gesture recognition) [3]. The large variations in people appearance, motion performance, and clothing makes human action recognition a challenging task to perform. In general, the human action recognition can be defined as the process of recognizing the human movements or actions from images or videos.

II. THEORETICAL BACKGROUND

The standard HGR process has 6 main elements. It is shown in figure 1.

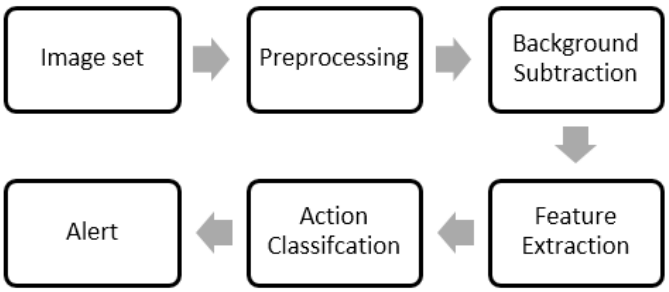

Figure1.Standard HGR Process

Image set: contains the set of images or image on which HGR process is performed.

- Pre-processing: consist of preliminary functions such as reading of image, noise removal, and image conversion.
- Background Subtraction: In this part, the un-necessary part (background) of an image is detected and removed from further processing. It contains methodologies for hand detection.
- Feature Extraction: The feature of hands i.e. hand contour, finger-tip, hand ridges are extracted.
- Action Classification: The appropriate action or signal of hand is classified based on features.

- Recognized Hand Gesture: In the end, the hand gesture is recognized and appropriate alert or function is performed. This is shown in figure 1. [2]

When developing a system for hand gesture recognition, many steps are to be considered. Those steps are discussed below. The first step is about capturing the image or data.

2.1 Data Acquisition or (Image set)

For effective hand gesture recognition, data acquisition should be as much perfect as possible. There are several input devices for data acquisition. Some of them are given below.

- Data gloves: They are the devices for perfect data input with high accuracy and high speed. It provides accurate data of joint angle, rotation, location etc. for application in different virtual reality environments.
- Colored markers: Colored markers attached to the human skin are also used as input. Color properties are used for hand localization. Bare hand is used for hand capture and gesture recognition.

The latest addition to this list is Microsoft Kinect 3D depth sensor Kinect is a 3D motion sensing input device widely used for gaming. It consists of a laser projector and a CMOS sensor for operation in any lighting [19].

2.2 Gesture Modelling (Background subtraction)

In gesture modelling, different data captured via the input devices are modelled based on application type. Gesture modelling has four different steps, [16][18] hand segmentation, filter/ noise removal, edge/ contour detection and lastly normalization.

Hand segmentation refers to process of locating the hand in the image. There are varioustechniquesfor segmentation. It divides image in 2 parts; foreground containing the hand and background containing the rest.

- Thresholding:In this method the image is divided in two parts, foreground and background.It is based upon a particular value (known as thresholding value). Such techniques are; range/ depth thresholding [19], some of them have used color (RGB, HSV) thresholding [20][22] and some uses speed thresholding.
- Skin-based: Human skin properties are used for detection of hand [24][20][22]. It separates the background and foreground based on defined color range. HSV, YCbCr are example of such techniques.
- Subtraction: In this approach the background of a test image is separated with reference to a previous fixed image. It finds any changes in the test image. This is mostly used in finding moving objects in videos from static camera.
- Statistical model: Here, the image is converted into a statistical model. Then we assign a probabilistic value for each pixel to be a foreground and background loyalty. Some common approaches used are; Bayesian rule based [26], Gaussian mixture model [13], Expectation Maximization (EM).

Noise Removal:Noise is a factor which affects the quality of an image as well as any image processing activity done on it. There are many methods for noise removal. Example of such techniques are; Salt and pepper, Morphology erosion [7][10], Multidimensional mean [13].

Edge detection: Edge or contour detection is a technique which finds boundary of the object from image. Mainly there are two types of edge detection techniques; Gradient based Edge detection and Laplacian based Edge detection. The gradient method detects the edges by finding the maximum and minimum in the first derivative of the image, whereas the Laplacian method determines the zero crossings in the second derivative of the image to detect edges.

Normalization: The last step of gesture modelling is normalization or feature space reduction [13]. As the region of interest for any image is concentrated in a small area, therefore it is desirable to crop only the relevant area and then process it further. This will speed up the processing to locate the geometric features.

2.3 Feature Extraction

Features are the essential element for HGR process. There are large number of features, like, motion, distance, shape, orientation, textures, contour, and centre of gravity. Other types of features are geometric features; handcontour, fingertips, number of finger. But these features suffer from occlusions and illuminations [16]. Some non-geometric features are; color, silhouette, and texture.

2.4 Recognition (Classification)

Once the necessary features are extracted from the images, these features are sent to recognizer or a special-purpose classifiers. Several methods for gesture recognition are: neural network,template matching, dictionary look-up, statistical matching, linguistic matching and ad hoc method.

Hidden Markov Model (HMM) is a doubly stochastic model.It is appropriate for dealing with the stochastic properties in HGR. Some researchers selects HMM as first choice due to higher recognition rates. [7][8]

Takagi Sugeno Kang (TSK) type fuzzy inference system has been used in [11]. Some researchers have widely used different forms of neural network e.g. K-mean based radial basis function neural network [10].

Another popular method used is Support Vector Machine (SVM). SVM is a supervised learning and takes a set of input data and predicts the output, for each given input based on its training on standard available data set.

III. PROPOSED METHOD

Hand gesture classification is the increasing interest for HCI. Hand gesture recognition is generally addressed by using either YCbCr (luminance and chrominance components) or HSV (hue, saturation, value) mappings which assume that a hand can be distinguished from the background from some colorfulness and luminance properties. In addition, current recognition methodsare dependent on descriptors or geometric shapes which can be reliable; however comes at increased computational complexity [1]. The proposed HGR process can be depicted as from figure 2.

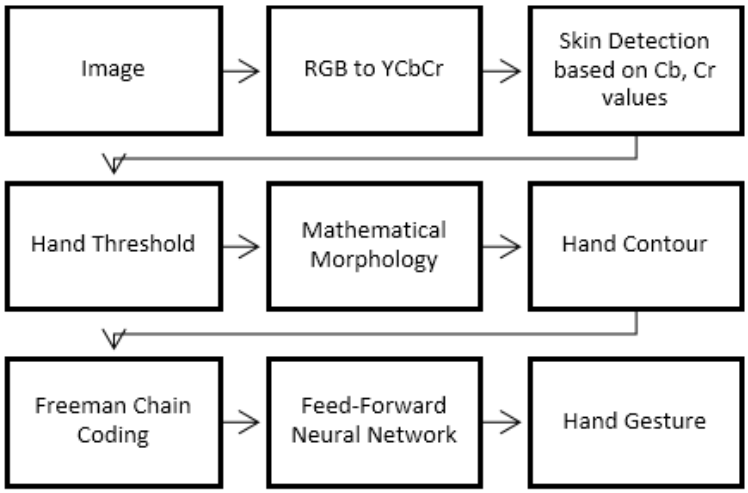

Figure2.Proposed HGR Process

3.1. Preprocessing

In the first step, we read an image from file. The read image will be in RGB. Then we convert the image from RGB to YCbCr.

3.2. Segmentation

In second step, for detecting hand portion from image YCbCr model segmentation technique is used. YCbCr model segmentation is color based model for skin detection. In YCbCr, Cb refers to the Blue-Chrominance component of image and Cr refers to the Red-Chrominance component of image while Y is the luminance component which controls lightness of image.
After deep research and practice, some researchers have found a range which detects skin pixel from image with minimal interference. [29]
A mask is prepared based on this value which will contain the hand skin pixels.

3.3. Hand Threshold

In hand thresholding, the RGB image is converted into binary image where the skin mask is kept and non-skin mask area is removed as is considered as background.

3.4. Mathematical Morphology

In this step, we find all connected components from all components in an image. Then we keep the maximum connected component and remove all other. This step removes any small background skin pixels and holes in an image. This generates the clear hand portion.

3.5. Hand Contour

In this step we generate hand contour. This can be done using various edge detetion techniques as well as Morphological operations. Edge detection techniques include Canny, Roberts, Prewitt, Sobel and morphological operations like boundary extraction operation. After implementing all of these techniques, we conclude cany edge detection gives best result. The best result is in terms of smoothness of contour and correctness of edge position. The canny edge detection is a 5 step process [32].

1. Apply Gaussian filter to smooth the image in order to remove the noise
2. Find the intensity gradients of the image
3. Apply non-maximum suppression to get rid of spurious response to edge detection
4. Apply double threshold to determine potential edges
5. Track edge by hysteresis: Finalize the detection of edges by suppressing all the other edges that are weak and not connected to strong edges.

The formula for Gaussian filter is;[32]

$$H_{ij} = \frac{1}{2\pi\sigma^2} * e^{-\frac{(i-k-1)^2+(j-k-1)^2}{2\sigma^2}}$$

3.6. Freeman Chain Coding

Chain code [27] is used to represent a boundary by a connected sequence of straight line segments of specified length and direction. This representation can be based on 4 (or) 8 connectivity of the segments (figure 3). The direction of each segment is coded by using a numbering scheme. A boundary code in form of sequence of such directional numbers is called as a Freeman Chain Code. The result of chain code is depending on starting point. This method allow us to analyze the shape in different manner. Chain Code traverses the shape in clockwise manner and keeps track of the directions as we go from one edge pixel to the next.

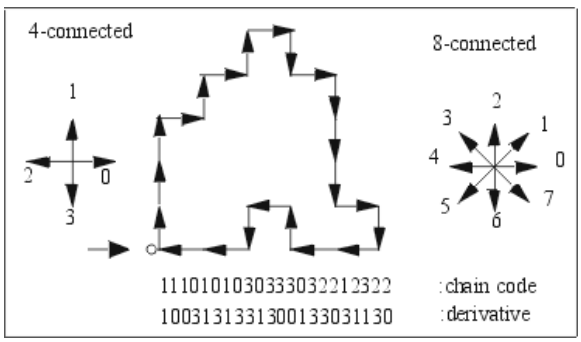

Figure3.Directional Chain Code (4 and 8 Connected)

To find the chain code, first object need to be scanned from left to right. After finding the starting point of an object boundary is traverse till end pixel. Direction code is identified and store into array or list. As shown in figure 3, scanning boundary is started from any point and then traversed clockwise or anti-clockwise. At every new pixel, its chain code is derived from table given below. The value of chain code can be derived as shown in table 1. $\Delta\mathbf{x}$ shows the pixel value changed in x-direction and $\Delta\mathbf{y}$ shows the pixel value changed in y-direction. The corresponding 8-code and 4-code is shown in table 1.

Table1.Chain Coding

Δx	Δy	8 code	4 code
+1	0	0	0
+1	+1	1	
0	+1	2	1
-1	+1	3	
-1	0	4	2
-1	-1	5	
0	-1	6	3
+1	-1	7	

3.7 Chain Code Histogram

To normalize the chain code, Chain Code Histogram is used. The chain code histogram (CCH) is meant to group together objects that look similar to a human observer. CCH has made by counting the number of each kind of steps in Freeman chain code representation of the contour. This is shown in figure 4.

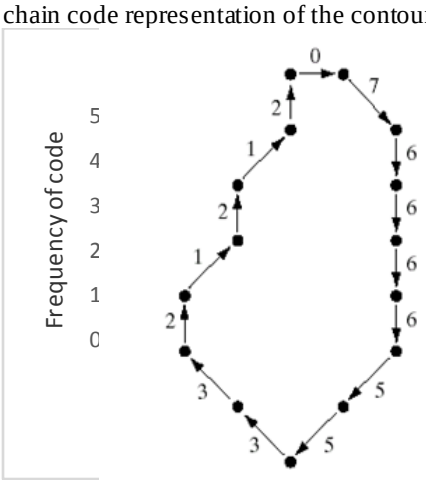

Figure4.Chain Code Histogram

3.8 Recognition

The network structure we use is multi-layer feed-forward network with back-propagation. The output from one layer feeds forward into the next layer of neurons, and nodes between two layers are fully connected with various weights. Error is calculated by the difference of outputs from output layer and target outputs. Then errors are feed backward to adjust weights,which are then applied to next iteration to generate new network outputs.

IV. IMPLEMENTATION & RESULTS

The implementation is done in MATLAB. The image set is prepared manually. It is shown in figure 5. All images are having resolution of 576 x 768. We have used image set of 128 images, some of them are shown in figure 9. The Image set contains 16 classes of gesture.

Figure5.Image set

4.1 Preprocessing

At first, we will read an image from computer.This images will contain the hand gestures.Reading of an image in Matlab is done using `imread()`. Such two images are shown in figure 6.

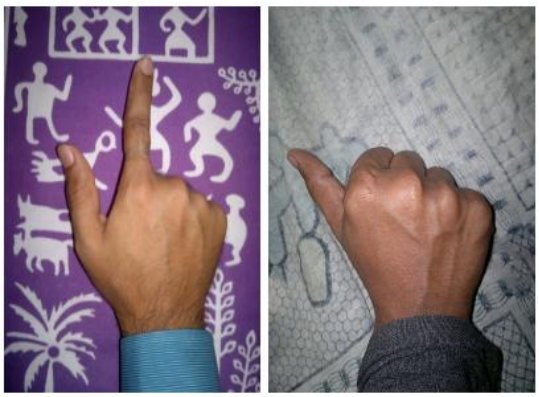

Figure6.Input Image

4.2 YCbCr Model Segmentation

The image used so far is RGB. To detect skin pixels from image, it is converted to YCbCr. This is done using matlab function `rgb2ycbcr()`. The value of Cb and Cr is set in range of [29]

`77 <= Cb <= 127` and `133 <= Cr <=173`

Then for varying range of r, skin is detected. If skin pixel is found, it is set to blue pixel (RGB = [0 0 255]). This skin

Figure7.Hand Segmentation

mask is shown in figure 7.

4.3 Hand Threshold and Mathematical Morphology

Now this image is converted to black andwhite (`im2bw()`) using thresholdingfunction `graythresh()`. Then Maximumcomponent is kept from all components toremove un-necessary components fromimage. Result after that is shown in figure 8.

Figure8.Hand Threshold

4.4 Hand Contour

Then hand contour (boundary) of hand is created using Canny Edge Detection (`contour = edge(img,'canny')`). This is shown in figure 9.
The hand contour is then used to find the boundary of hand. The boundary points of hand (contour points) will be used as an input to chain coding.

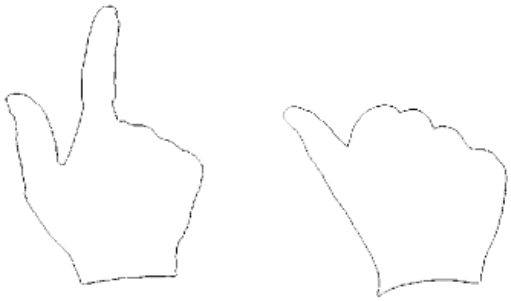

Figure9.Hand Contour

4.5 Freeman Chain Coding

The contour points are given to freeman chain coding algorithm.

Table 2.Chain Code Mapping

dx	dy	Code	Mapping
0	+1	2	8
0	-1	6	2
-1	+1	3	7
-1	-1	5	1
+1	+1	1	9
+1	-1	7	3
-1	0	4	4
+1	0	0	6

Here, boundary points are given in pair of (x,y) which is converted to scalar for 8-connected code. Taking base 3, the pairing is done using following formula.

```
idx = 3*(dy+1)+(dx+1) = 3dy+dx+4
   Mapping Array = (base-3) +1
```

4.6 Chain Code Histogram

Now histogram of chain code is prepared i.e. we count the frequency of chain code and store it in an array. For given image, the histogram looks like following figure 10.

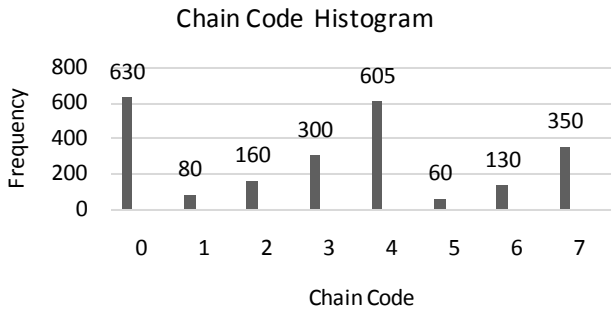

Figure10.Chain Code Histogram

4.7 Recognition

Then FF-Back-propagation NN is used for recognition. The specification are given below in table 3. We have used two hidden layer and varied the no of hidden neurons. After using different number of neurons, maximum recognition is achieved at 10 no of neurons. Therefore we use no hidden neurons as 10.

Table3. NN Input Specification

No of Training Images	No of Testing Images	No of Gestures
96	32	16
Total Images		128

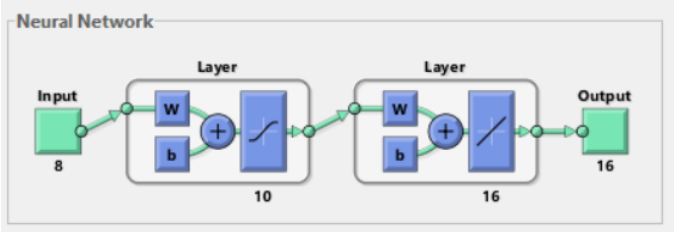

Figure11.Neural Network Parameters

The figure 11 shows the Neural Network parameters. The feature vector size is 8. This is given as input in neural network. The output in one of the 16 gesture class.

The recognition result is given in table 4.We have achieved recognition rate of 90.63% when testing on 32 test images.

Table.4. Recognition Result

No of Test Images	Correctly Recognized	Recognition Rate
32	29	90.63%

V. CONCLUSION

Here we have analyzed various techniques for hand gesture recognition and implemented an approach that increases performance with same or less computational load. Using freeman chain coding has allowed us to classify the many gesture classes with good accuracy. Proposed method works with cluttered images and it is invariant to translation. Our proposed approach is successfully implemented with average recognition rate of 90.63%.In future, we can improve the recognition rate by using multiple features compare it with other techniques.

REFERENCES

[1] Boughnim et al.: Hand posture recognition using jointly optical flow and dimensionality reduction, OpenSpringer EURASIP Journal on Advances in SignalProcessing2013:167

[2] Siddharth S. Rautaray, Anupam Agrawal, "Vision based hand gesture recognition for human computer interaction". Springer Science, Business Media Dordrecht 2012

[3] Niranjan Deokule, Geetanjali Kale. "Human Action Recognition Using Kinect". International Journal Of Engineering And Computer Science ISSN: 2319-7242 Volume 3 Issue 7 July, 2014 Page No. 7199-7202

[4] OttenheimerHJ (2005) The anthropology of language: an introduction to linguistic anthropology. Wadsworth Publishing. ISBN-13: 978-0534594367

[5] Hall ET (1973) The silent language. Anchor Books. ISBN-13: 978-0385055499

[6] Kanniche MB (2009) Gesture recognition from video sequences. PhD Thesis, University of Nice

[7] Yang, Z., Li, Y., Chen, W. and Zheng, Y. (2012) Dynamic Hand Gesture Recognition Using Hidden Markov Models. In Proc. of 7th Int. Conf. on Comp. Sc. and Education

[8] Oka, K., Sato, Y. and Koike, H. (2002) Real-time Tracking of Multiple Fingertips and Gesture Recognition for Augmented Desk Interface Systems. 5th IEEE Int. Conf. on Automatic Face and Gesture Recognition

[9] Wu, Y. and Huang, T. S. (1999) Vision-Based Gesture Recognition: A Review. Int. Gesture Workshop on Gesture-Based Communication in Human-Computer Interaction. Springer Lecture Notes in Computer Science

[10] Ghosh, D. K. and Ari, S. (2011) A Static Hand Gesture Recognition Algorithm using K-Mean based Radial Basis Function Neural Network. In Proc. of 8th Int. Conf. on Information, Communications & Signal Processing

[11] Symeonidis, K. 2000. Hand Gesture Recognition Using Neural Networks. MS Thesis. University of Surrey, UK

[12] Pavlovic, V. I., Sharma, R. and Huang, T. S. Visual Interpretation of Hand Gestures for Human - Computer Interaction: A Review. IEEE Transactions on Pattern Analysis and Machine Intelligence, 19(7) (1997), 677 – 695

[13] Hasan, M. M. and Mishra, P. K. Hand Gesture Modeling and Recognition using Geometric Features: A Review. Canadian J. on Image Processing and Computer Vision, 3(1)1 (2012)

[14] Lamar, M.V., Bhuiyan, Md. S. and Iwata, A. 1999. Hand Gesture Recognition using Morphological Principal Component Analysis and an Improved CombNET-II. In Proc. IEEE Int. Conf. on Systems, Man, and Cybernetics (Vol. 4)

[15] Liang, R. and Ouhyoung, M. 1998. A Real-time Continuous Gesture Recognition System for Sign Language. In Proc. 3rd IEEE Int. Conf. on Automatic Face and Gesture Recognition

[16] Murthy, G. R. S. and Jadon, R. S. A Review of Vision Based Hand Gestures Recognition. Int. J. of Information Technology and Knowledge Management, 2(2) (2009), 405 – 410

[17] Erol, A., Bebis, G., Nicolescu, M., Boyle, R. D., and Twombly, X. Vision Based Hand Pose Estimation: A Review. Computer Vision and Image Understanding, 108 (2007), 52– 73

[18] Wu, Y. and Huang, T. S. Hand Modeling, analysis and Recognition. IEEE Signal Processing Magazine, 18(3) (2001), 51 – 60

[19] Li, Y. 2012. Hand Gesture Recognition using KINECT. MS thesis. University of Louisville

[20] Rajesh, J. R., Nagarjunan, D., Arunachalam, R. M. and Aarthi, R. Distance Transform based Hand Gestures Recognition for Power point presentation Navigation. An Int. J. of Adv. Computing. 3(3) (May 2012), 41- 48

[21] Gurjal, P. and Kunnur, K. Real time Hand Gesture Recognition using SIFT. Int. J. of Electronics & Electrical Engg. 2(3) (March 2012), 19 – 33

[22] Rautaray, S. S. and Agarwal, A. Real time Hand Gesture Recognition System for Dynamic Applications. Int. J. of UbiComp, 3(1) (January 2012), 21 – 31

[23] Sultana, A. and Rajapuspha, T. Vision based Gesture Recognition for Alphabetical Hand Gestures using the SVM Classifier. Int. J. of Comp. Sc. And Engg. Tech. 3(7) (July 2012), 218 – 223

[24] Malima, A., Ozgur, E. and Cetin, M. 2006. A Fast Algorithm for Vision-Based Hand Gesture Recognition for Robot Control. In Proc. of 14th IEEE Signal Processing and Communications Applications

[25] Freeman, W. T., Weissman, C. D. (1995). Television Control by Hand Gestures. IEEE International Workshop on Automatic Face and Gesture Recognition

[26] Bhuyan, M. K., Neog, D. R. and Kar, M. K. Fingertip Detection for Hand Pose Recognition. Int. J. On Computer Sc. and Engg. , 4(3) (March 2012), 501-511

[27] Kshama Fating, Archana Ghotkar PERFORMANCE ANALYSIS OF CHAIN CODE DESCRIPTOR FOR HAND SHAPE CLASSIFICATION International Journal of Computer Graphics & Animation (IJCGA) Vol. 4, No.2, April 2014

[28] URL:http://www.seattlerobotics.org/encoder/nov98/neural.html

[29] D. Chai, and K.N. Ngan, "Face segmentation using skin-color map in videophone applications". IEEE Trans. on Circuits and Systems for Video Technology, 9(4): 551-564, June 1999

[30] Bourke A, O'Brien J, Lyons G (2007) Evaluation of a threshold-based tri-axial accelerometer fall detection algorithm. Gait & Posture 26(2):194–199.

[31] URL: http://www.mathworks.com/matlabcentral/fileexchange/29518-freeman-chain-code

[32] URL:http://en.wikipedia.org/wiki/Canny_edge_detector

