Gestuelle: A System to Recognize Dynamic Hand Gestures using Hidden Markov Model to Control Windows Applications

J.R Pansare
University of Pune
Pune, India

Malvika Bansal
University of Pune
Pune, India

Shivin Saxena
University of Pune
Thane, India

Devendra Desale
University of Pune
Pune, India

ABSTRACT

Human Computer Interaction has always been a challenging adventure for researchers. Communication between computers and humans, just as humans interact with one another has been the prime objective of HCI research. Many efforts have gone into Speech and Gesture Recognition to develop an approach that would allow users to interact with their system by using their voice or simple intuitive gestures as against sitting in front of the computer and using a mouse or keyboard. Natural interaction must be fast, convenient, effective and reliable. This paper introduces an application, "Gestuelle" that makes use of simple gestures to operate on common windows applications such as Windows Media Player, Live Photo Gallery, Power Point, Notepad etc. The idea is to develop a system that can recognize dynamic hand gestures by means of a simple web camera to control the computer even from a distance, without having to use a keyboard and mouse all the time. Gestuelle provides a cheap and easily portable solution to the everyday user as against using an expensive Microsoft Kinect or high resolution cameras or sensors to accomplish the same task. The system makes use of the Hidden Markov Model (HMM), works in real time and is designed to work in static backgrounds. The system makes use of LRB topology of HMM in conjunction with the Baum Welch Algorithm for training and the Forward and Viterbi Algorithms for testing and evaluating the input observation sequences and generating the best possible state sequence for pattern recognition.

General Terms: Computer Vision, Pattern Recognition, Image Processing

Keywords: Computer Vision, Dynamic Gesture Recognition, HCI, HMM, skin detection

1. INTRODUCTION

In the earlier stages of technological advances in the field of computers, a keyboard was the only means of communication to interact with the computer (after magnetic tapes and punch cards became obsolete). The mouse brought with it a revolution and an entirely new dimension to Human Computer Interaction, or popularly abbreviated as HCI. Many inventions followed such as the light pen, tablets, digitizers and more recently the Space Mouse each bringing with it an innovative style of interacting with the computer and opening new possibilities and dimensions of interaction.

Thus, Human Computer Interaction is not just a necessity but a challenging endeavour to push the current boundaries of interaction. Speech Recognition and Synthesis has been a prominent domain of research during the last decade of the twentieth century and of late a number of successful implementations have been encountered in mobile phones etc. However, an even more prominent research field has been that of interaction by the use of simple intuitive hand gestures.

Sign language is a common form of communication between auditory handicapped people, can be employed to communicate with a robot or any computer. Imagine sitting on the couch and operation your computer from a distance with your voice or just simple day-to-day hand movements. It would not only eliminate the need to actually physically touch your mouse and keyboard unless absolutely necessary, but will also be so much more convenient and quick.

Gesture recognition can be performed in a number of ways: some use sensory gloves or high resolution cameras such as the Microsoft Kinect with the XBOX 360. The former technique mainly utilizes specially designed sensory gloves the angles and spatial position of the hand and fingers relative to one another. For example an approach in gesture recognition used a sensing glove with 6 embedded accelerometers. It could recognize 28 static hand gestures and the computation time was about 1 characters/second. However, the proposed algorithm was not efficient enough to be applied in real time. Although the former is a powerful technique it's not really a natural way to interact with the computer because one has to continually wear the gloves. Natural HCI should be glove free, fast, reliable and convenient.

For Computer Vision based techniques, one or a set of cameras are used to capture images for hand gesture recognition. In this case, gesture recognition can further be classified as: Static Gestures and Dynamic Gestures. In case of static gestures, the hand remains in a fixed position and usually snapshots of a certain gesture being performed is compared against an extensive database to recognize the pattern and thus, the gesture. Dynamic Gesture Recognition on the other hand is an entirely different scenario. One has to consider factors such as the direction in which the hand is progressing, velocity, orientation etc. which makes the task of recognizing the gesture even more challenging.

In this research, a different approach for dynamic hand gesture recognition using HMM model against a static background has been proposed. The concept is to develop an application that would recognize some defined gestures and perform associated tasks, such as opening and closing some application, zooming in and out of an image, rotating it, print

etc on some common windows applications such as Windows Media Player, Windows Live Gallery, Microsoft Power Point and Notepad. The functionality can later be extended to other applications as well. The system is named "Gestuelle", a French word for gesticulate which means to indicate feelings by motions, that is to move the hands in an animated manner. Gestuelle is a simple application that is easily portable to all windows platforms (supporting a minimum of .Net framework 2.0) and having an in-built or externally mounted web camera of even the lowest resolutions. Gestuelle continually captures image frames while the gesture is being performed and processes them in the background so that by the time the gesture is completed, the results are available almost immediately. In order to detect the hand region, skin detection is used and the pattern recognition will be carried out by using the Hidden Markov Model. Following gestures were used:

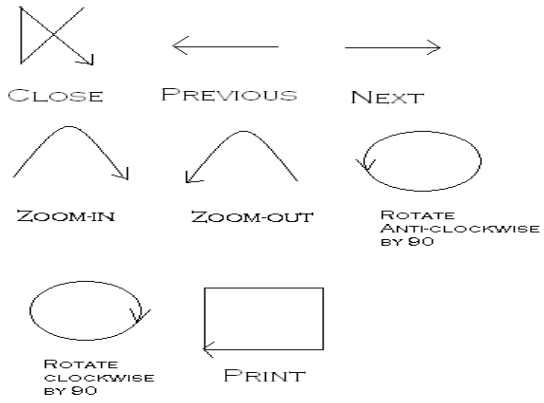

Fig. 1. Various Gestures used in the System

While referring to a number of papers, we came across a number of novel and innovative approaches to recognize dynamic gestures using HMM. In [1], the paper based on the use of Haar Wavelet Representation for gesture recognition discussed an approach for an effective computer vision system. Hands were extracted by detecting the skin colour. The problem of hand orientation in the image was solved by utilizing the idea of axis of elongation. It helped immensely in keeping the database small by standardizing the hand gestures in the database using fixed orientations. To facilitate the searching process, a codeword scheme mechanism was utilized. To further improve the success rate the use of a measurement scheme employing a penalty score during recognition was proposed. Experimental results illustrated in the paper showed a good hit rate for recognition of gestures.

TABLE I. Recognition results of 15 hand gestures

Sign	Correct(%)	Sign	Correct(%)	Sign	Correct(%)
A	86.67	F	96.67	K	96.67
B	90.00	G	93.33	L	96.67
C	90.00	H	90.00	M	96.67
D	96.67	I	100	N	96.67
E	100	J	100	O	93.33

Paper [2] introduces an entirely different approach to gesture recognition that used Hidden Markov Models. Here, a graphic editor that could recognize five static and twelve dynamic gestures, was being developed. Gesture recognition was carried out by using a structural analysis for static gestures and HMM for dynamic ones. According to [2], Hidden Markov models have intrinsic properties which make them an attractive option for gesture recognition, and also explicit segmentation is not necessary for either training or recognition.

In [3], Jinli and Tianding propose a thesis for hand trajectory recognition based on HMM, that can model spatio-temporal information in a natural way. In order to be able to differentiate undefined gestures, a modified threshold model was proposed. The hand was separated from its background by the use of skin color detection by first converting the RGB based pixels to YCbCr color model and then defining a suitable range to recognize skin colour.

In [4], the authors proposed an automatic system that recognizes isolated gesture; in addition meaningful gesture from continuous hand motion for Arabic numbers from 0 to 9 in real-time based on Hidden Markov Models. In order to handle isolated gesture, HMM using Ergodic, Left-Right (LR) and Left-Right Banded (LRB) topologies was applied over the discrete vector feature that was extracted from stereo color image sequences. These topologies were then corresponded to different number of states ranging from 3 to 10.

In this approach to dynamic hand gesture recognition, the hand is detected from the image frames, captured by the webcam, using skin detection by converting from the RGB color model to HSV (Hue, Saturation, and Value). Next, Laplacian of Gaussian filter is used to perform the edge detection and then locate the centroid of the hand region by drawing a blob around the palm. LRB topology of HMM is being used as it not only reduces unnecessary backward transitions (as seen in Ergodic and LR topologies) but also, simplifies the training data. 8 observations symbols and 8 states are being utilized which in this approach is more than sufficient for efficient gesture recognition for a variety of gestures.The structure of this paper is as follows: Section II briefly introduces our system Gestuelle, Section III gives the System Overview, Section IV Hidden Markov Models, Section V shows the results of our implementation, Section VI will be the Conclusion followed by the References.

2. Gestuelle

In this section, the human-computer interaction interface, Gestuelle has been described. One can directly open the supported applications from the Applications menu, view all camera devices (both internal and external) connected to the computer and also, select one amongst them and then start the preview which appears in the screen beneath the menu-bar. Clicking the Start buttons begins the capturing and processing of image frames, while the gesture is being performed. This mode of operation of Gestuelle is called as the "User Mode". In addition to this, a "Developer Mode" has been provided which has been designed explicitly for advanced users who understand the Hidden Markov Model parameters. Figure 4, gives a glimpse of this mode. Once selected, the user can view the current values of the model parameters that are the Transition matrix, Emission Matrix and the Initial Probabilities. The user can provide their own training samples to the application via this interface and by clicking "Learn" can activate the Baum-Welch algorithm which shall train the model according to the set of training data provided. The efficiency of the current model can be tested by selecting Evaluate from the Developer Mode start screen, which runs the Viterbi algorithm over all the training samples.

The interface window of Gestuelle always remains on top of other application windows so that the preview is always

available for the user to see and is not hidden by any other activity in the background. When minimized, Gestuelle runs in the system tray so that it is easy to restore it back whenever needed and more importantly so that it consumes minimal system resources. Gestuelle can continually record and analyse gestures, that is, once one gesture is performed the next can immediately be performed thereafter without having to press the Start button. One can switch to the Developer Mode from the default User Mode from the Menu tab.

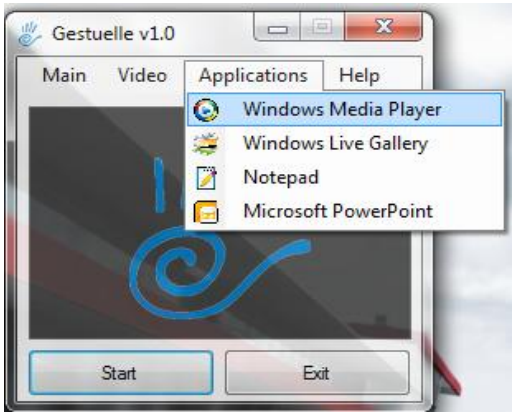

Fig. 2. A look at Gestuelle's user interface. The Applications tab allows one to select anyone of the supported applications.

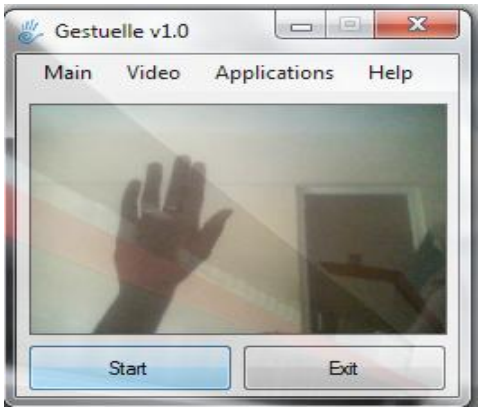

Fig. 3. The Gestuelle interface with the preview enabled and the image frames captured in a background process.

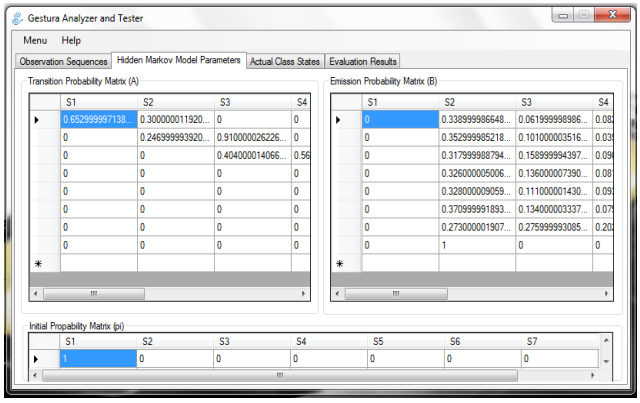

Fig. 4. The "Hidden Markov Model Parameters" Tab(Developer Mode) showing the current values of the Transition, Emission and Initial Probabilities Matrices. This mode also has provision for learning samples and evaluating the results by selecting the other tabs.

3. System Overview

In this section, the following block diagram summarizes the approach that will be used to implement the system:

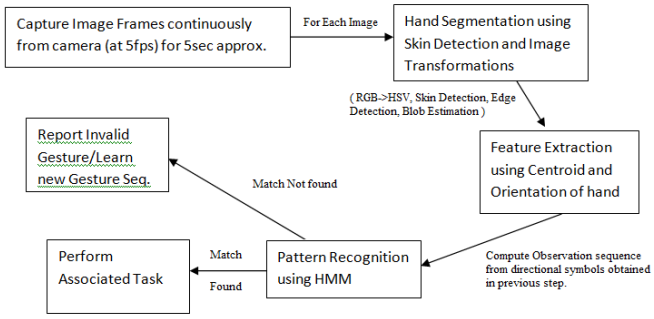

Fig. 5. Block Diagram for the System

A. Frame Capture

The first stage of the process implementation is capturing image frames from the web camera. The frequency of image capture is set at 5 frames per second and every third frame is analysed.

B. Skin Detection

Referring to [5], it is known that the RGB color model includes the information of both color and brightness, which is vulnerable to the changes in background illumination and environment. To ensure this doesn't affect the skin color, convert RGB color model to HSV color model, because the latter is more related to human color perception. The skin in channel H is characterized by values between 0 and 50 and in the channel S from 0.23 to 0.68 for Asian and Caucasian skin.

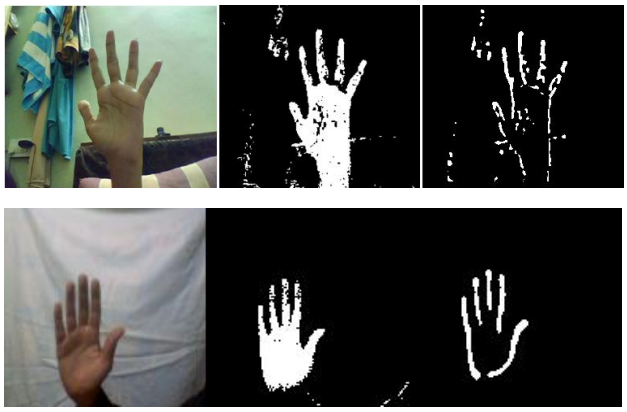

Fig. 6. The first image in the first set is the original image frame captured from the web camera. The center image is a binary image obtained after converting from RGB to HSV and then whitening out the skin pixels while blackening the rest. The last image is obtained after edge detection. The images in the second set were taken against a plain white background.

After changing the pixel color values, using RGB to HSV conversion and then setting all pixels that fall in the range of skin values to white (255) and the non-skin pixels to black (0), a binary image is obtained. However, this image has some noise in it i.e. some pixels that are actually not part of the skin

but still have fallen in the given range and must be eliminated. This is accomplished by the help of morphological filters.

C. *Edge Detection and Feature Extraction*

Once the skin regions have been whitened out (while blackening the rest), perform edge detection using Laplacian of Gaussian filter which makes use of the concept of zero crossing (that is, an edge is sure to be present wherever there is a sudden change in intensity from white to black or vice-versa).

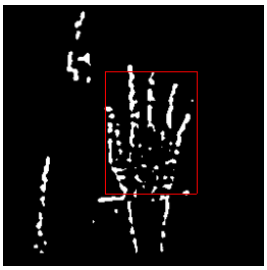

Fig. 7. Image obtained after Blob Detection

Next, a blob is drawn only around the hand region (as shown in the figure above) and find the centroid of this rectangular region. The feature used for forming the states of HMM is the 'orientation' of the hand which is calculated by finding the centroid of all image frames and then finding the angle formed between the centroids of two consecutive image frames.

To find the orientation, code numbers have been assigned to the directions in which the hand seems to be progressing. 8 observation symbols have been used in the Markov model, which will be explained in the following section. The figure (a) represents the observation symbols that are going to be used. Figure (b) was used in [6], [7], [9] and [12]. Through experimentation, we realised that there was no need for so many symbols and that 8 symbols were sufficient for performing most rectangular gestures and also, gestures involving curves.

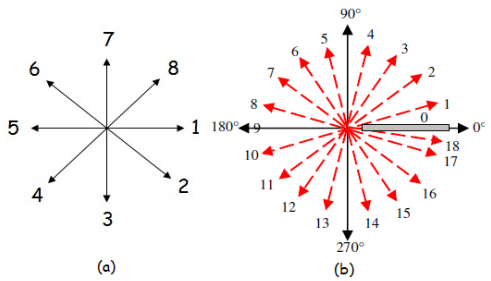

Fig. 8. **The Orientation symbols: (a) 8 code words from 1 to 8 (b) 18 directional code words ranging from 1 to 18 including a zero codeword for continuous gestures**

The trick is to find the angle between consecutive frames, and then round it off to the nearest direction to get the required code words.

4. The Hidden Markov Model

Mathematically, the Hidden Markov Model is specified by [11]:

The set of states S = {s1, s2, . . . , sN}, (corresponding to the N possible gesture conditions above) and a set of parameters: $\lambda = \{ \Pi, A, B\}$ For observations (symbols) **O** = { O_k }, k = 1, . . . , M .

A. Transition probabilities

- **A** = $\{a_{ij} = P(q_j \text{ at } t+1 \mid q_i \text{ at } t)\}$, where P(a | b) is the conditional probability of a given b, t=1,. . . , T is time, and q_i in **Q**. Informally, **A** is the probability that the next state is $q_j$given that the current state is q_i.

B. Emission probabilities B is

- **B** = { $b_{ik} = b_i(O_k)$ = $P(O_k \mid q_i)$ }, where o_k in **O**. Informally, **B** is the probability that the output is o_k given that the current state is q_i.

C. Initial state probabilities

Π = $\{p_i = P(q_i \text{ at } t = 1)\}$.

Now, in order to find the observation sequence associated with each gesture, represent each change in direction involved while performing the gesture with the appropriate observation symbol.

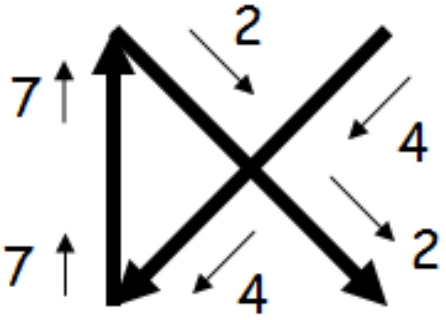

Fig. 9. Each line segment in this gesture is being shown along with the observation symbol (direction) associated with it.

In this approach, 8 distinct directions and therefore, 8 discrete Observation symbols have been used. As a result of this and the choice of our gestures, 8 states are being used.

There are 3 canonical problems associated with HMM that must be solved:

A. Given the model parameters, compute the probability of a particular output sequence. This problem is solved by the Forward and Backward algorithms (described below).

B. Given the model parameters, find the most likely sequence of (hidden) states which could have generated a given output sequence. Solved by the Viterbi algorithm and posterior decoding.

C. Given an output sequence, find the most likely set of state transition and output probabilities. Solved by the Baum-Welch algorithm.

Putting the above in different words, Forward and Viterbi algorithms will be used to find and test the best possible state sequence that can generate the input observation sequence given the current model parameters (π, A, B).

The Baum-Welch algorithm has been used for the learning or training process which involves estimating appropriate values for the model parameters such that the likelihood of producing the given input data samples is maximised, that is, finding parameter values that best fit the training data set. The learning process can be either iteration based or convergence based. In case a limited number of iterations are used, the algorithm will repeat the learning process a fixed number of times. In case the learning is convergence based, the algorithm will stop when the change in the likelihood for two consecutive iterations has not changed by more than X percent of the likelihood (given that the value of the convergence parameter is X). In this paper, the second approach for Baum Welch learning has been used.

5. Result

The recognition rate of different gestures is as per the table. From the results, it is observed that it is relatively easier to capture the linear gestures, rather than inclined or circular ones, but still the algorithms implemented perform effectively and the overall recognition rate achieved by the system is high because of the use of the Markov model and adequate training (960 samples were used for training the model). Here is an example of the training set used (The final test data has values spread across four columns and 960 rows. There are 13 symbols in each sequence.):

TABLE II. A SUBSET OF THE ACTUAL TRAINING SET USED

Observation Sequence	Label	No. of States	Gesture
1-1-0-0-0-0-0-0-0-2-2-2	A	2	Left
0-0-0-0-1-1-0-0-0-0-0-2	A	2	Left
2-2-5-5-4-4-4-4-5-5-4-4-4	B	2	Right
6-6-4-4-4-4-4-4-4-4-2-2-6	B	2	Right
2-2-5-5-6-6-6-6-7-7-6-6-7	C	3	Up
2-2-6-6-6-6-6-6-6-6-6-6-2	C	3	Up
6-6-5-5-4-4-4-4-3-3-3-3-2	E	5	Semi-circle Clockwise
6-6-5-5-4-4-4-4-3-3-3-3-3	E	5	Semi-circle Clockwise

TABLE III. PERCENTAGE ACCURACY OF THE RECOGNITION OF DIFFERENT GESTURES

Gesture	Recognition Accuracy (%)
Right	98
Left	98
Up	97
Down	95
Semi-Circle Clockwise	94
Semi-Circle Anti-Clockwise	91
Circle Clockwise	93
Circle Anti-Clockwise	90

6. Conclusion

The Hidden Markov Model serves as an indispensible tool for the recognition of dynamic gestures in real time. Based on observations from other references the accuracy of our proposed approach is expected to be high.

In the future, this approach can be further improved upon by taking into account the effect of speed of movement of hand, more complicated and dynamic backgrounds, better tolerance to background illumination and implementing the system for both hands.

A. Advantages

1) The system can be used conveniently to communicate with the computer at a distance and since we are making use of HMM, the accuracy rate is also increased with increased training of the system.
2) The system works perfectly fine for even the simplest of web cameras and no expensive sensors or high resolution cameras are needed. It is easily portable and will work on most windows platforms (having .Net 2.0 and higher).
3) The application gives results in real time that is, is very fast and in future, the functionality can be extended to more applications.

B. Limitations

1) There might be miss-recognitions in case the background has elements that resemble the human skin.
2) A number of other factors such as velocity of movement, orientation and low background illumination might take a toll on the system's accuracy.

7. References

[1] Wing Kwong Chung, Xinyu Wu, Yangsheng Xu, "A realtime hand gesture recognition based on Haar wavelet representation", Robotics and Biomimetics, 2008. ROBIO 2008. IEEE International Conference, pp. 336 – 341, 22-25 Feb. 2009.

[2] Byung-Woo Min, Ho-Sub Yoon, Jung Soh, Yun-Mo Yangc', Toskiaki Ejima, "Hand Gesture Recognition Using Hidden Markov Model", pp. 305-333.

[3] Jinli Zhao and Tianding Chen, "An Approach to Dynamic Gesture Recognition for Real-time Interaction", ISNN 2009, pp. 369-377.

[4] Shuying Zhao, Wenjun Tan, Chengdong Wu Chunjiang Liu, Shiguang Wen, "A novel interactive method of virtual reality system based on hand gesture recognition", Control and Decision Conference, 2009. CCDC '09. Chinese, pp. 5879 – 5882, 17-19 June 2009.

[5] Rokade, Doye, Kokare, "Digital Image Processing, 2009 International Conference", pp. 288-291, 7-9 March 2009.

[6] Mahmoud Elmezain, Ayoub Al-Hamadi, J¨org Appenrodt, and Bernd Michaelis, "A Hidden Markov Model-Based Isolated And Meaningful Hand Recognition", Institute for Electronics, Signal Processing and Communications (IESK), Otto-von-Guericke-University Magdeburg, D-39106 Magdeburg, Germany.

[7] Mahmoud Elmezain, Ayoub Al-Hamadi, J¨org Appenrodt, and Bernd Michaelis," Hand Gesture Recognition Based on CombinedFeatures Extraction", Otto-von-Guericke-University Magdeburg, D-39106 Magdeburg, Germany.

[8] S. Mitra, and T. Acharya, Gesture Recognition: A Survey, IEEE Transactions on Systems, MAN, and Cybernetics, pp. 311-324, 2007.

[9] M. Elmezain, A. Al-Hamadi, G. Krell, S. El-Etriby, and B. Michaelis, Gesture Recognition for Alphabets from Hand Motion Trajectory Using Hidden Markov Models, The IEEE International Symposium on Signal Processing and Information Technology, pp. 1209-1214, 2007.

[10] Y. Ho-Sub, S. Jung, J. B. Young, and S. Y. Hyun, Hand Gesture Recognition using Combined Features of Location, Angle and Velocity, Journal of Pattern Recognition, Vol. 34(7), pp. 1491-1501, 2001.

[11] Lawrence R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition", Readings in speech recognition Pages 267 – 296, ISBN:1-55860-124-4.

[12] M. Elmezain, A. Al-Hamadi, and B. Michaelis, "A Novel System for Automatic Hand Gesture Spotting and Recognition in Stereo Color Image Sequences", The Journal of WSCG, Vol. 17 No. 1, pp. 89-96, 2009