Dynamic Gesture Recognition Using Hidden Markov Model in Static Background

Malvika Bansal[1], Shivin Saxena[2], Devendra Desale[3] and Dnyaneshwar Jadhav[4]

**[1] Department of Computer Engineering, MES College Of Engineering
Pune University, Maharashtra, India**

**[2] Department of Computer Engineering, MES College Of Engineering
Pune University, Maharashtra, India**

**[3] Department of Computer Engineering, MES College Of Engineering
Pune University, Maharashtra, India**

**[4] Department of Computer Engineering, MES College Of Engineering
Pune University, Maharashtra, India**

Abstract

Human Computer Interaction is a challenging endeavour. Being able to communicate with your computer (or robot) just as we humans interact with one another has been the prime objective of HCI research since the last two decades. A number of devices have been invented, each bringing with it a new aspect of interaction. Much work has gone into Speech and Gesture Recognition to develop an approach that would allow users to interact with their system by simple 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 and reliable. In our project, we intend to develop one such natural interaction interface, one that can recognize hand gesture movements in real time using HMM but by using Computer Vision instead of sensory gloves.

Keywords: *HCI, Dynamic, Gesture Recognition, Skin colour detection, Computer Vision, HMM*

1. Introduction

Technology has come a long way from computers running on vacuum tubes to semi-conductor chips and finally, the super computers of today's era. In the earlier stages of computing inventions, 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 current boundaries of interaction. Speech Recognition and Synthesis has been a prominent domain of research over the last decade. Even more prominent has been interaction by the use of simple intuitive Hand Gestures. Sign language is a common form of communication between auditory handicapped people. It can further 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. This is the power of Human Computer Interaction.

Both static and dynamic hand gesture recognition is a challenging aspect of HCI. Gesture Recognition can be implemented by one of the two methods, one, by wearing sensory gloves on one's hand or second, by with the aid of Computer Vision (CV). For glove based techniques, it mainly utilizes sensory gloves to measure the angles and spatial positions of a hand and fingers. Referring to one of the papers based on this field, we came across an approach in gesture recognition that 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. A sensory glove would not be a convenient option for daily usage.

For Computer Vision based techniques, one or a set of cameras are used to capture images for hand gesture recognition. Using backward reference from a paper based on the use of Haar Wavelet for recognizing gestures, we came across another proposed algorithm that would first separate the hand region from the complex background images by measuring entropy from adjacent frames. Hand gestures would then be recognized by the approach of improved centroidal profile. However, mis-recognitions can be caused by hand gestures with similar spatial features and therefore the number of gestures that could be recognized was limited.

Referring to [1], the paper based on the use of Haar Wavelet Representation for gesture recognition, we came across another approach for an effective computer vision system. The proposed approach was based on skin-colour detection. Hands were extracted by detecting the skin colour. The problem of hand orientation in the image is also 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 metric with a penalty score during recognition was proposed. Experimental results showed a good hit rate for recognition of gestures.

Referring to [2], we then came across an entirely different approach to gesture recognition that used Hidden Markov Models. In this paper, a Graphic Editor that could recognize five static and twelve dynamic gestures, was being developed. Recognition was facilitated by the 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.

Further, referring to [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 colour detection by first converting the RGB based pixels to YCbCr colour model and then defining a suitable range to recognize skin colour.

In our final year project, we intend to propose an approach for dynamic hand gesture recognition using HMM model against a static background. 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 and even printing the image. Following are the gestures that we will be working with:

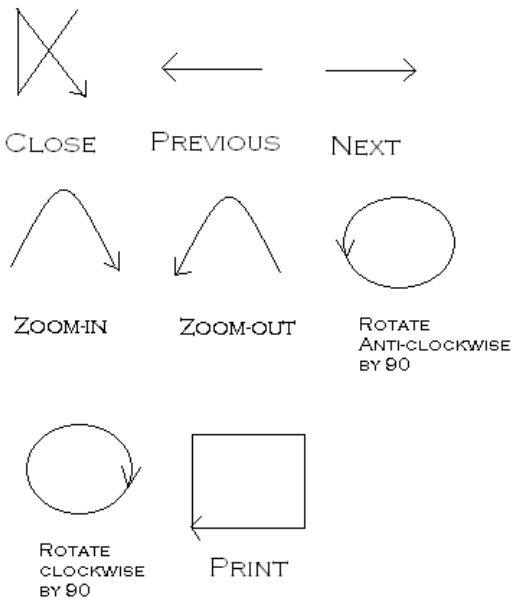

Fig. 1 Various Gestures used in the System

In addition to the HMM model, which will be used to represent the gestures in an adjacency matrix, we will also be making use of the idea of principal axis through the centroid of the hand to standardize the gestures, thereby reducing the size of the image dataset. We will be working with the HSV colour model.

The structure of this paper is as follows: Section II gives the System Overview, Section III describes Segmentation based on Skin Colour Detection, Section IV demonstrated use of HMM for hand gestures, Section V is the implementation and finally Section VI will be the Conclusion.

2. System Overview

The following block diagram summarizes the approach that we will be using to implement our system:-

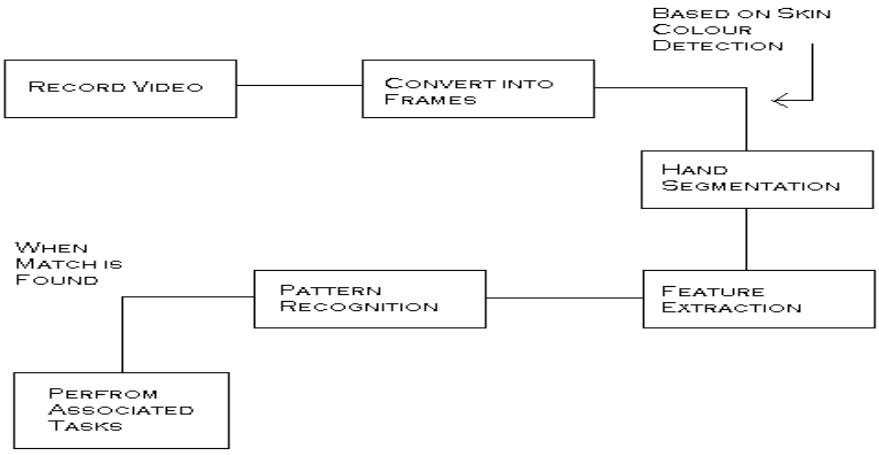

Fig. 2 Block Diagram for the System

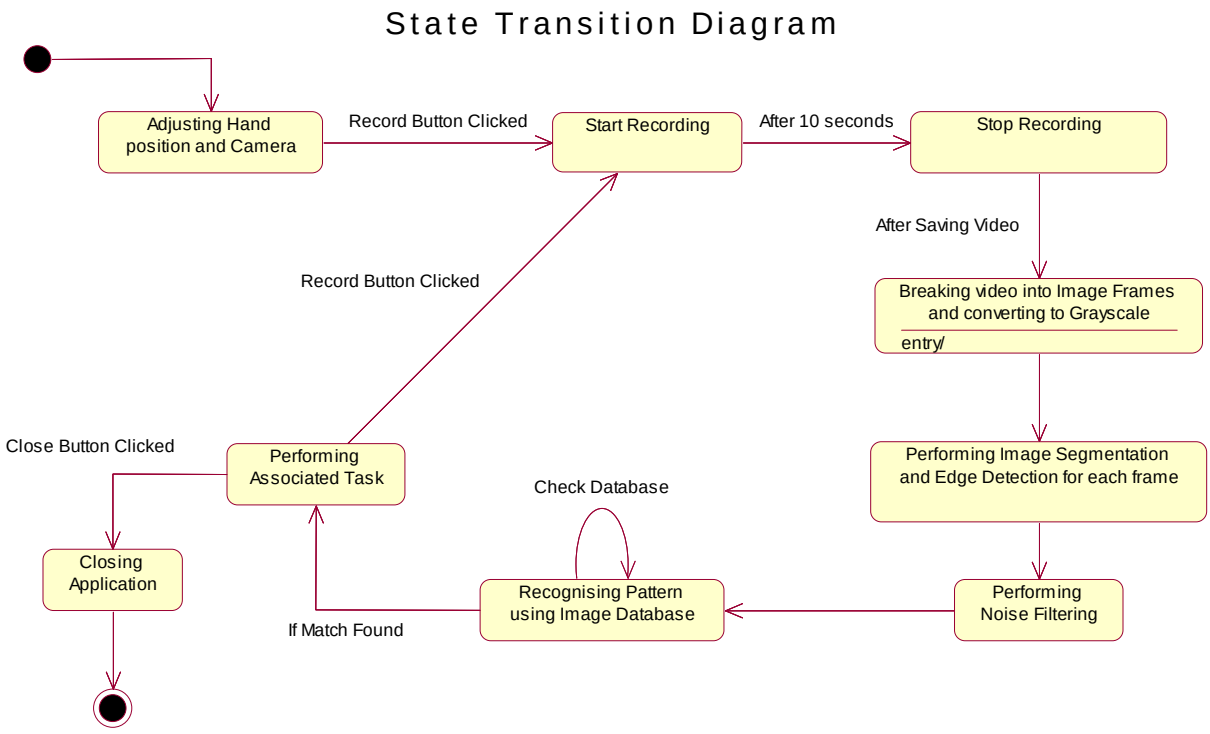

Fig. 3 State Transition Diagram for the System

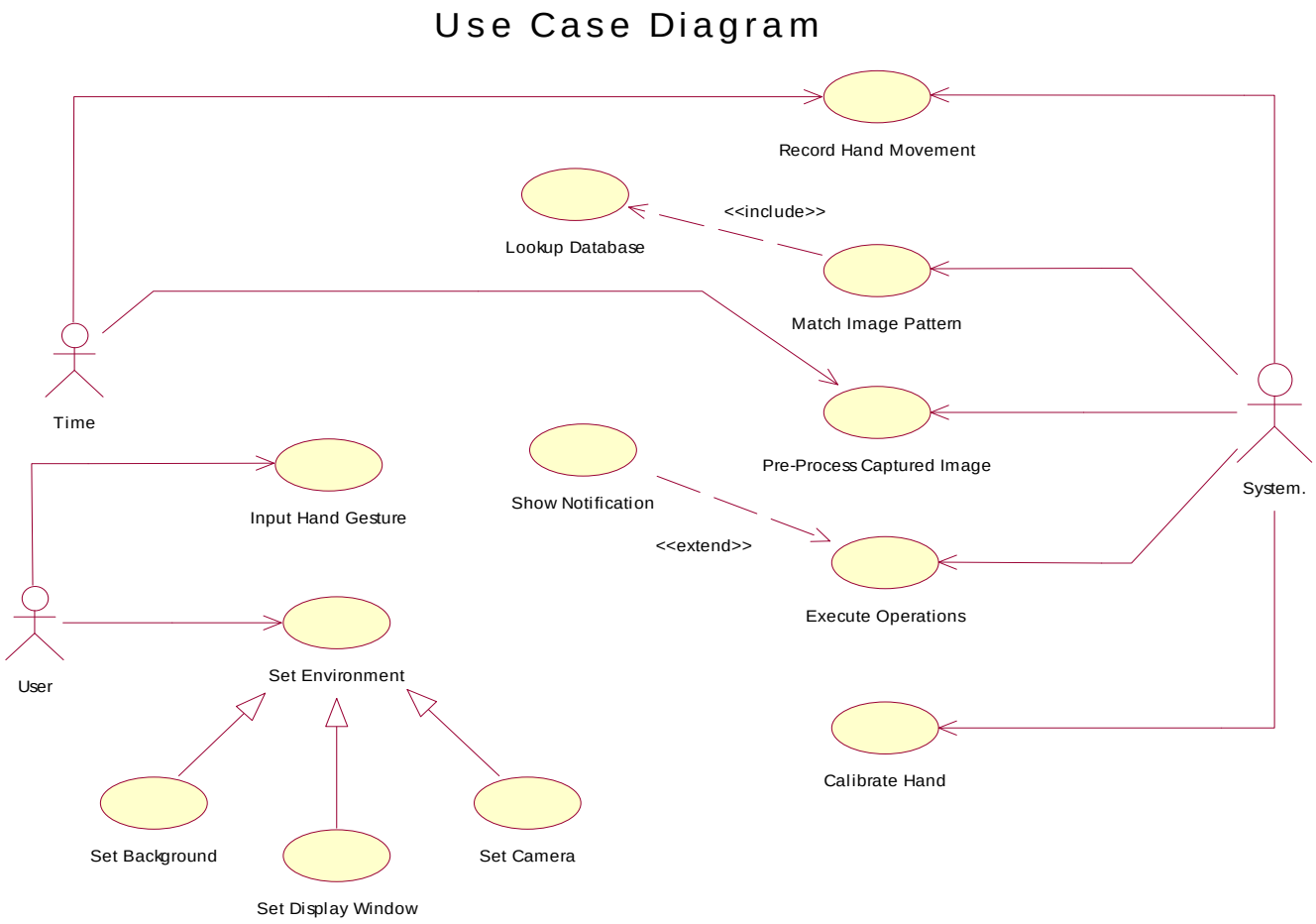

Fig. 4 Use Case Diagram for the System

3. Hand Segmentation using Skin Detection

The RGB colour model includes the information of both colour and brightness, which is vulnerable to the changes in background illumination and environment. To ensure these doesn't affect the skin colour, we will be converting the RGB colour model to HSV colour model, because the latter is more related to human colour perception. The skin in channel H is characterised by values between 0 to 50, in the channel S between 0.23 to 0.68 for Asian and Caucasian skin. The figure below shows the original image in RBG colour format.

Fig. 5 Original Image

After changing the pixel colour values, using RGB to HSV conversion, we get the following image:

Fig. 6 Image after conversion from RGB to HSV

The next image is an intermediate image which is obtained immediately after setting all pixels that fall in our skin colour range to 255(white) and the rest non-skin pixels to 0 (black). However, this image has some noises 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.

The final image obtained after noise removal is shown as under:

Fig. 7 Final Image after Noise Removal

Finally only the skin regions are represented as white pixels. To convert from RGB to HSV (assuming normalized RGB values) first find the maximum and minimum values from the RGB triplet.
Saturation, S, is then given by:

$$S = \frac{(max - min)}{Max} \quad \text{Eq. (1)}$$

and Value, V, is given by:
V = max
The Hue, H, is then calculated as follows. First calculate R'G'B':
R' = (max-R) / (max-min)
G' = (max-G) / (max-min)
B' = max-B / (max-min) Eq. (2)

If saturation, S, is 0 (zero) then hue is undefined (i.e. the colour has no hue therefore it is monochrome) otherwise: then,
if R = max and G = min
H = 5 + B' Eq. (3)
else if R = max and G 6= min
H = 1 – G' Eq. (4)
else if G = max and B = min
H = R' + 1 Eq. (5)
else if G = max and B 6= min
H = 3 – B' Eq. (6)
else if R = max
H = 3 + G' Eq. (7)
otherwise
H = 5 – R' Eq. (8)

Hue, H, is then converted to degrees by multiplying by 60 giving HSV with S and V between 0 and 1 and H between 0 and 360.

4. The Markov Model

We shall first give a brief introduction to the Hidden Markov Model. Consider a person who is sitting inside a

room and has three coins. He is tossing these coins in sequence known only to him. We are positioned outside this room and are shown by means of a display (also placed outside the room), the outcomes of the man flipping the coins, for example, HTTHTHTTHHTHTHH. This is referred called the Observation Sequence. We do not know the sequence in which the coins are being tossed and neither do we know the bias of the individual coins. To understand the significance of the impact of the bias of the coins on the outcome, imagine that the third coin is highly biased to generate Tails. Now, if all the coins are tossed with equal probability then it would be naturally expected that the output will have more Tails than Heads. Further, consider that the probability of moving from the first or second coin (state) to the third coin (state) is zero. Now if we had started the tossing with the first and second coins then the output sequence will generate more tails because of the transition probability between the coins/states and also, the initial state. These three sets, namely, the set of individual bias of the three coins, the set of transition probabilities from one coin to the next and the set of initial probabilities characterize what is called as the Hidden Markov Model.

The HMM Model is specified by:

- The set of states S = {s1, s2, . . . , sNs}, (corresponding to the N possible gesture conditions above),
And a set of parameters: λ = { Π, A, B}

- Transition probabilities
  $\boldsymbol{A} = \{a_{ij} = P(q_j \text{ at } t+1 \mid q_i \text{ at } t)\}$,
- where $P(a \mid b)$ is the conditional probability of a given b, $t = 1, \ldots, T$ is time, and q_i in $\boldsymbol{Q}$. Informally, $\boldsymbol{A}$ is the probability that the next state is $q_j$given that the current state is q_i.
- Observations (symbols) $\boldsymbol{O} = \{ O_k \}, k = 1, \ldots, M$.
- Emission probabilities B is:
  $\boldsymbol{B} = \{ b_{ik} = b_i(O_k) = P(O_k \mid q_i) \}$, where o_k in $\boldsymbol{O}$. Informally, $\boldsymbol{B}$ is the probability that the output is o_k given that the current state is q_i.
- Initial state probabilities $\boldsymbol{\Pi} = \{p_i = P(q_i \text{ at } t = 1)\}$.

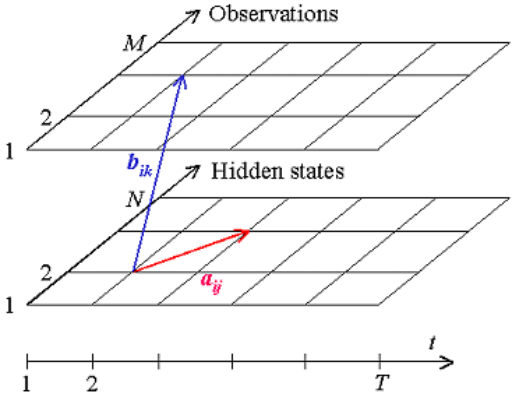

Fig. 8 Observed and Hidden Sequences

A HMM allowing for transitions from any emitting state to any other emitting state is called an Ergodic HMM. The other extreme, a HMM where the transitions only go from one state to itself or to a unique follower is called a left-right HMM.

There are 3 canonical problems to solve with HMMs:

1. Given the model parameters, compute the probability of a particular output sequence. This problem is solved by the Forward and Backward algorithms (described below).
2. 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.
3. Given an output sequence, find the most likely set of state transition and output probabilities. Solved by the Baum-Welch algorithm.

4.1 Forward Algorithm

Let $\alpha_t(i)$ be the probability of the partial observation sequence $O_t = \{o(1), o(2), \ldots, o(t)\}$ to be produced by all possible state sequences that end at the i-th state.

$$\alpha_t(i) = P(o(1), o(2), \ldots, o(t) \mid q(t) = q_i) \quad \text{Eq. (9)}$$

Then the unconditional probability of the partial observation sequence is the sum of $\alpha_t(i)$ over all N states.

The Forward Algorithm is a recursive algorithm for calculating $\alpha_t(i)$ for the observation sequence of increasing length t . First, the probabilities for the single-symbol sequence are calculated as a product of initial i-th state probability and emission probability of the given symbol $o(1)$ in the i-th state. Then the recursive formula is applied. Assume we have calculated $\alpha_t(i)$ for some t. To calculate $\alpha_{t+1}(j)$, we multiply every $\alpha_t(i)$ by the corresponding transition probability from the i-th state to the j-th state, sum the products over all states, and then multiply the result by the emission probability of the symbol $o(t+1)$. Iterating the process, we can eventually calculate $\alpha_T(i)$, and then summing them over all states, we can obtain the required probability.

In a similar manner, we can introduce a symmetrical backward variable $\beta_t(i)$ as the conditional probability of the partial observation sequence from $o(t+1)$ to the end to be produced by all state sequences that start at i-th state (3.13).

$$\beta_t(i) = P(o(t+1), o(t+2), \ldots, o(T) \mid q(t) = q_i) \quad \text{Eq. (10)}$$

The Backward Algorithm calculates recursively backward variables going backward along the observation sequence. The Forward Algorithm is typically used for calculating the probability of an observation sequence to be emitted by an HMM, but, as we shall see later, both procedures are heavily used for finding the optimal state sequence and estimating the HMM parameters.

4.2 Viterbi algorithm

The Viterbi algorithm chooses the best state sequence that maximizes the likelihood of the state sequence for the given observation sequence.

Let $\delta_t(i)$ be the maximal probability of state sequences of the length t that end in state i and produce the t first observations for the given model.

$$\delta_t(i) = \max\{P(q(1), q(2), ..., q(t\text{-}1) ; o(1), o(2), ... , o(t) \mid q(t) = q_i).\} \quad \text{Eq. (11)}$$

The Viterbi algorithm is a dynamic programming algorithm that uses the same schema as the Forward algorithm except for e

1. It uses maximization in place of summation at the recursion and termination steps.
2. It keeps track of the arguments that maximize $\delta_t(i)$ for each t and i, storing them in the N by T matrix $\boldsymbol{\psi}$. This matrix is used to retrieve the optimal state sequence at the backtracking step.

4.3 Baum-Welch Algorithm

An important aspect of the HMM training is that the model should decode the observation sequence in such a way that if an observation sequence having many characteristics similar to the given one be encountered later it should be able to identify it. There are two methods that can be used for such identification:

- The Segmental K-means Algorithm
- The Baum-Welch Re-estimation Formulas

(We will be using and discussing the second method)

Let us define $\xi_t(i,j)$, the joint probability of being in state q_i at time t and state q_j at time $t+1$, given the model and the observed sequence:

$$\xi_t(i,j) = P(q(t) = q_i, q(t+1) = q_j \mid O, \Lambda) \quad \text{Eq. (12)}$$

Therefore we get

$$\xi_t(i,j) = \frac{\alpha_t(i)\, a_{ij}\, b_j(o(t+1))\, \beta_{t+1}(j)}{P(O \mid \Lambda)} \quad \text{Eq. (13)}$$

The probability of output sequence can be expressed as

$$P(O \mid \Lambda) = \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i)\, a_{ij}\, b_j(o(t+1))\, \beta_{t+1}(j) = \sum_{i=1}^{N} \alpha_t(i)\, \beta_t(i) \quad \text{Eq. (14)}$$

The probability of being in state q_i at time t:

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) = \frac{\alpha_t(i)\, \beta_t(i)}{P(O \mid \Lambda)} \quad \text{Eq. (15)}$$

5. Conclusion

The Hidden Markov Model serves as an indispensible tool for the recognition of dynamic gestures in real time. Also, standardising the axis through the centroid will greatly reduce the database size. Based on observations from other references the accuracy of our proposed algorithm 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 and also, implement the system for both hands and to able to recognize and make use of the entire forearm.

5.1 Advantages

- 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.
- The idea of standardizing the orientation of images in the database greatly reduces its size and thus facilitates a faster searching process.

5.2 Disadvantages

- There might be miss-recognitions in case the background has elements that resemble the human skin.

- A number of other factors such as velocity of movement, orientation and low background illumination might take a toll on the system's accuracy.

5.3 Acknowledgment

All the authors would like to extend their sincere gratitude and thanks to their project guides who in spite of their hectic schedule were a constant source of guidance and motivation and helped them explore the depths of image processing and computer vision.

6. 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] Rokade, Doye, Kokare, "Digital Image Processing, 2009 International Conference", pp. 284-287, 7-9 March 2009.