Asian Journal of Electrical Sciences
ISSN: 2249-6297 (P) Vol.4 No.1, 2015, pp.34-44

DOI: https://doi.org/10.51983/ajes-2015.4.1.1931

An Efficient Vision-Based Hand Beckon Perception for Physically Debilitated People using MCMC and HMM

J.P.Justina[1] and Sangeetha Senthilkumar[2]
[1] II [nd] M.E (CSE), Department of Computer Science, [2]Assistant Professor, Department of Computer Science,
Oxford Engineering College, Trichy, Tamil Nadu, India
E-mail: [1]justinarajasekar@yahoo.com, [2]sangeethasenthilkumar2002@gmail.com
(Received 6 March 2015; Revised 31 March 2015; Accepted 30 April 2015; Available online 10 May 2015)

***Abstract* - Recognition of hand gestures has a significant impact on human society. It is a natural and intuitive way to provide the interaction between human and the computer. It provides touchless interaction and easy way to interact without any external devices. With the ever increasing role of computerized machines in society, Human Computer Interaction (HCI) system has become an increasingly important part of our daily lives. HCI determines the effective utilization of the available information flow of the computing, communication, and display technologies. Gesture recognition pertains to recognizing meaningful expressions of motion by a human, involving the hands, arms, face, head, and/or body. It is of utmost importance in designing an intelligent and efficient human–computer interface. Hidden Markov models (HMMs) and related models have become standard in statistics with applications in diverse areas. Markov chain Monte Carlo (MCMC) is great stuff. MCMC revitalized Bayesian inference and frequents inference about complex dependence. A high performance Artificial Neural Network (ANN) classifier is employed to improve the classification and accuracy.**
***Keywords:* Gesture recognition, Hidden Markov Model (HMM), Markov Chain Monte Carlo (MCMC), Human Computer Interaction (HCI), Artificial Neural Network (ANN).**

I. INTRODUCTION

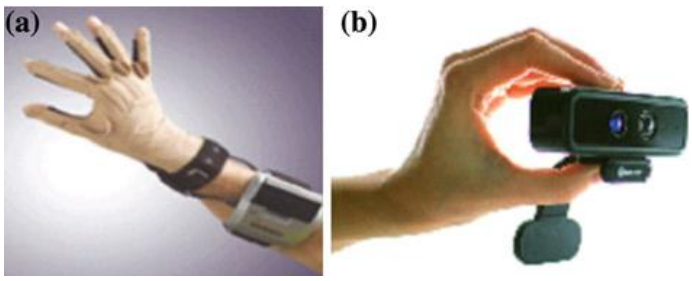
Gesture is basically a movement of the body part/parts which contain information or feelings. Gesture recognition is a mechanism through which a system can understand the meaning of any gesture. ISL recognition system has to recognize both single handed and double handed signs under complex background using a novel set of features. The main gesture taxonomies can be recognized [1].

HCI has a great influence towards the human society. With latest technology advance, we see an increasing availability and demand for intuitive HCI. Devices can also be controlled using gestures in public areas and in our homes instead of controlling by mouse and keyboard. We can now interact with the machines freely around.

Hand gesture recognition system can be used as interface to communicate with speech impaired and will bridge the communication gap between hearing impaired and normal people. Researchers are working in various sign languages for developing such systems.

Static gestures: The static gestures are called “hand postures”. Posture is a specific combination of hand position, orientation and flexion observed at some instance of time. Posture or static gestures are not time varying signals [2], so they can be analyzed using only one or a set of images of hand.

Dynamic gestures: Dynamic gesture is a sequence of postures connected by motions over a short time span. A gesture can be thought of as a sequence of postures. In the video signals, the individual frames define postures and the video sequences define the gestures.



The approaches can be classified into (a) Data-glove based [3] and (b) vision-based [4]. Tracking bare hand and recognizing the hand gestures using low level features such as color, shape or depth information [5] generally require uniform background, invariable illumination, a single person in the camera view. Many recognition systems are based on data glove, an expensive wired electronic device. Various sensors are placed on the glove to detect the global position and relative configurations of the hand. One limitation of this method is the price of the glove and other problem it needs is a physical The vision-based gestural interface system includes the following parts: Image acquisition, Foreground segmentation, Face and Hand detection, link between the users and the computer.

So more researchers show interest in vision-based systems which are wireless and the only thing needed is one or more multiple cameras. In the current script, the tracking algorithm was significantly upgraded and compared with other algorithms to obtain a better tracking performance. Hand tracking under frequent self-occlusion was depicted as a multiobject tracking (MOT) problem. Since hands are non-rigid objects and its form changes among different humans, while performing a certain candidate gesture. Also, since both the hands are looking similar for the same individual, trackers can focus on one hand or exchange positions when the hands are too close to each other. In this paper, an integrated approach was proposed to handle the challenging problem of tracking under self-occlusion.

II. SYSTEM OVERVIEW-OUTLINE OF THE APPROACH

The architecture of the proposed system is illustrated in the following figure.

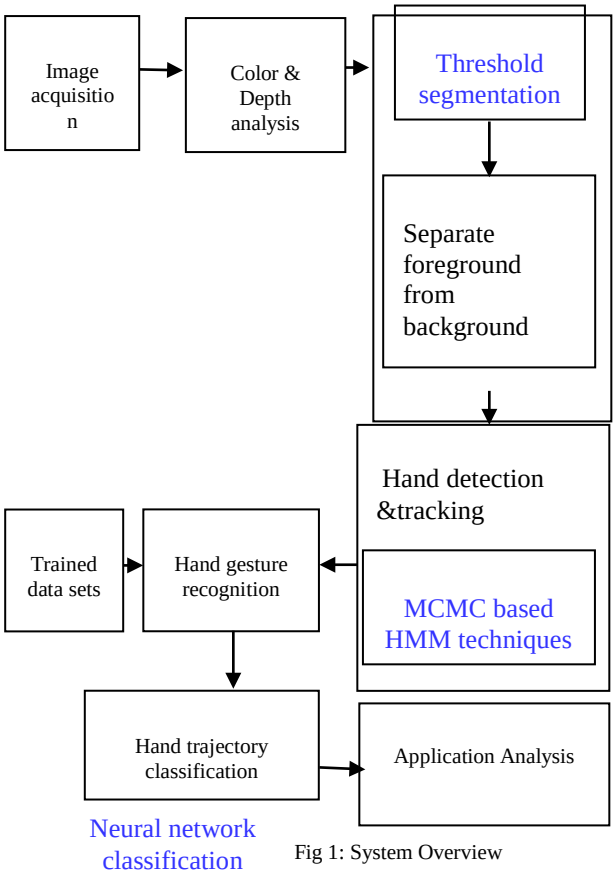


Fig 1: System Overview

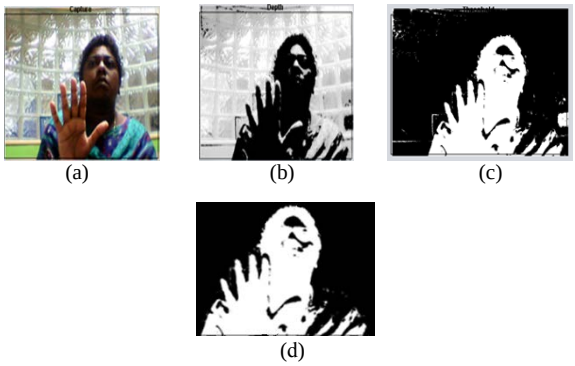
The vision-based gestural interface system includes the following parts: Image acquisition, Foreground segmentation, Face and Hand detection, Hand tracking, Hand trajectory classification, Application analysis.

A. Image Acquisition

The system acquires images captured from a webcam. The output of webcam is basically a video sent to the system. The system will acquire videos in the form of sequence of frames.

B. Foreground Segmentation

Segmentation is one of the crucial steps of gesture recognition. In foreground segmentation section, the background is removed from the captured frames and the whole human body was kept as the foreground.



(a) (b) (c)

(d)

Fig. 2. Foreground Segmentation
(a)Captured image from webcam (b) Depth image. (c) Depth threshold mask. (d) Foreground segmentation mask

C. Face and Hand Detection and Tracking

Face detection is used to remove the face from the skin-color histogram that is constructed. Hand detection is used to initialize the position of hands for the tracking phase. After initialization, hands were tracked through video sequences by MCMC with HMM method.

D. Hand Trajectory classification

Hand tracking results were segmented as trajectories compared with motion models and decoded as commands for the application analysis.

E. Application Analysis

The commands decoded by gesture recognition results were used by various applications and analyzed.

III. GESTURE RECOGNITION

Often, hand gesture recognition involves segmentation of hands, tracking them through occlusion, classification of hand's dynamic trajectories and static pose.

A. Image Acquisition

The system acquires images captured from a webcam. The output of webcam is basically a video sent to the system. The system will acquire videos in the form of sequence of frames. The captured RGB image is converted into depth
map image for the next step of segmentation. Depth Map is created for each frame or a series of homogeneous frames to indicate the depths of objects present in the scene.

A Depth Map Image is a separate gray scale image having the same dimensions as the original image with various shades of gray to indicate the depth of every part of frame.

B. Foreground Segmentation

Image segmentation is the process of dividing the image into continuous regions or set of pixels. Threshold segmentation is employed which is based on the intensity of pixels. Initially, the user's body was treated as a foreground object in order to detect the user's movements. Separating the foreground from background involves two steps. In the first step, the acquired image was thresholded using the depth information. The depth value of each pixel was defined as D (i, j) with i and j indicating the horizontal and vertical coordinates of the pixel in each frame of the video sequence.

Thresholding is used to extract an object from its background by assigning an intensity value T(threshold) such that each pixel is either classified as an object point or a background point. Global thresholding is chosen where a threshold T seperates the foreground object from background. Select an initial estimate of T (Initial threshold is the average of the gray values). Compute the means of the two regions determined by T. Set the new T as the average of the two means. Repeat the above steps until the difference in T in successive iterations is smaller than a predefined parameter. A mask image was generated by keeping the pixels with the depth value greater than threshold while discarding the others. In the second step, the region (blob) with the largest area was extracted from the mask image. All the remaining blobs with an area smaller than T_{SH} were discarded. If the extracted region contained an object that was not part of user's body, it would be discarded in a later stage because tracking was performed based on both color and spatial information.

Foreground Segmentation Algorithm

Algorithm1: Foreground Segmentation

Input: Threshold T, pixel value of Depth image
D (i, j)
Output: Pixel value of mask image $D_1(i, j)$; pixel value of foreground mask image $D_2(i, j)$

$$D_1(i, j) = \begin{cases} 1, & D(i,j) > T \\ 0, & otherwise \end{cases}$$

T_{SH} = max (Area (B_i)) //B_i is the i^{th} blob in the mask image D;

$$D_2(i, j) = \begin{cases} 1, & D1(i,j) \in Bi \text{ and } Area(Bi) == TSH \\ 0, & otherwise \end{cases}$$

C. Face and Hand Detection and Tracking

The centroids of the face and hand regions were extracted to initialize the tracking stage. A skin color histogram was created using the HSV color space to achieve higher robustness for skin color detection.

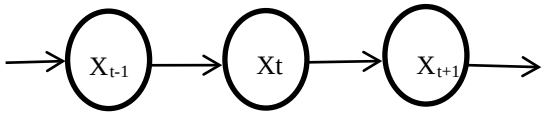
(a) (b) (c)

Fig. 3.Face and hand detection.

(a)Skin Color Detection (b) Hand Extraction (c) Face and Hand Localization

The mask image obtained from histogram back-projection is shown as in Fig 3(a). To obtain the hand regions without the face, a face detector was adopted as in Fig 3(c) to remove the face region from the target image. The largest blob in the target image was then selected as hand region [Fig. 3(b)]. The centroids of the hands were obtained by computing the first moment of the blob. This hand detection procedure was only used to provide automatic initialization to the tracking procedure. Afterwards the hands positions were continuously tracked by MCMC with HMM technique.

A Markov chain is a mathematical model for stochastic systems whose states, discrete or continuous, are governed by a transition probability. The current state in a Markov chain only depends on the most recent previous states.



MCMC works by generating a sequence of samples where each X_i depends on X_{i-1} in only a handful of components by sampling X' from an MCMC proposal distribution $q(X'/X_{i-1})$ and 'accepting' or 'rejecting' this proposal based on how likely it is under $\pi(X)$ compared to X_{i-1}.

In a *hidden* Markov model, the state is not directly visible, but output, dependent on the state, is visible. Each state has a probability distribution over the possible output tokens. Therefore the sequence of tokens generated by an HMM gives some information about the sequence of states. 'Hidden' refers to the state sequence through which the model passes, not to the parameters of the model. The observation is turned to be a probabilistic function (discrete or continuous) of a state instead of an one-to-one correspondence of a state. Each state randomly generates one of M observations.

Algorithm2: MCMC with HMM

Input: input state X_i
Output: output state X_{i+1} such that
$E[f(x)] \approx \frac{!}{N} \sum_i f(X_i)$ as $N \rightarrow \infty$

1. Sample $X' \sim q(X'/X_i)$
2. Calculate acceptance ratio

$$\alpha(X_i, X') - \min \frac{\bar{\pi}(X') q(Xi|X')}{\bar{\pi}(X') q(Xi|X')}, 1$$

3. Sample $u \sim$ Unif (0, 1). If $u < \alpha(X_i, X')$ let $X_{i+1} - X'$, else $X_{i+1} - X_i$.

D. Hand Trajectory Classification

For each frame in the video sequence, the hands were tracked during the tracking stage. The motion model for each gesture trajectory was created based on the data collected from the gestures performed. The neural network algorithm is employed to classify the static and dynamic hand gesture trajectories in the lexicon. An artificial neural network is an interconnected group of nodes, akin to the vast network of neurons in a brain.

Advantages of ANN

1. Provides high tolerance to noisy data.
2. Ability to classify patterns on which they have not been trained.
3. Faster learning rate of the classifier and the time required for classification is less.
4. Neural networks are nonlinear models, which makes them flexible in modeling real world complex relationships.
5. Improved classification accuracy.

Training an Artificial Neural Network

In the training phase, the correct class for each record is known (this is termed supervised training), and the output nodes can therefore be assigned "correct" values -- "1" for the node corresponding to the correct class, and "0" for the others. (In practice it has been found better to use values of 0.9 and 0.1, respectively.) It is thus possible to compare the network's calculated values for the output nodes to these "correct" values, and calculate an error term for each node (the "Delta" rule). These error terms are then used to adjust the weights in the hidden layers so that, hopefully, the next time around the output values will be closer to the "correct" values.

The Iterative Learning Process

A key feature of neural networks is an iterative learning process in which data cases (rows) are presented to the network one at a time, and the weights associated with the input values are adjusted each time. After all cases are presented, the process often starts over again. During this learning phase, the network learns by adjusting the weights so as to be able to predict the correct class label of input samples. Neural network learning is also referred to as "connectionist learning," due to connections between the units. The most popular neural network algorithm is back-propagation algorithm proposed in the 1980's.

Once a network has been structured for a particular application, that network is ready to be trained. To start this process, the initial weights are chosen randomly. Then the training, or learning, begins.

The network processes the records in the training data one at a time, using the weights and functions in the hidden layers, and then compares the resulting outputs against the desired outputs. Errors are then propagated back through the system, causing the system to adjust the weights for application to the next record to be processed. This process occurs over and over as the weights are continually tweaked. During the training of a network the same set of data is processed many times as the connection weights are continually refined.

Feedforward, Back-Propagation

The feedforward, back-propagation architecture was developed by several independent sources. This independent co-development was the result of a proliferation of articles and talks at various conferences which stimulated the entire industry. Currently, this synergistically developed back-propagation architecture is the most popular, effective, and easy-to-learn model for complex, multi-layered networks. Its greatest strength is in non-linear solutions to ill-defined problems. The typical back-propagation network has an input layer, an output layer, and at least one hidden layer. There is no theoretical limit on the number of hidden layers but typically there are just one or two. Each layer is fully connected to the succeeding layer.

As noted above, the training process normally uses some variant of the Delta Rule, which starts with the calculated difference between the actual outputs and the desired outputs. Using this error, connection weights are increased in proportion to the error times a scaling factor for global accuracy. Doing this for an individual node means that the inputs, the output, and the desired output all have to be present at the same processing element. The complex part of this learning mechanism is for the system to determine which input contributed the most to an incorrect output and how does that element get changed to correct the error. An inactive node would not contribute to the error and would have no need to change its weights. To solve this problem, training inputs are applied to the input layer of the network, and desired outputs are compared at the output layer. During the learning process, a forward sweep is made through the network, and the output of each element is computed layer by layer. The difference between the output of the final layer and the desired output is back-propagated to the previous layer(s), usually modified by the derivative of the transfer function, and the connection weights are normally adjusted using the Delta Rule. This process proceeds for the previous layer(s) until the input layer is reached.

Algorithm: Backpropagation.

Input:
D, a data set consisting of the training tuples and their associated target values;
l, the learning rate;
network, a multilayer feed-forward network.

Output: A trained neural network.

Method:

Initialize all weights and biases in network;
While terminating condition is not satisfied {
for each training tuple X in D{
// propagate the inputs forward;
for each input layer unit j{
$O_i = I_i$; // output of an input unit is its actual input value
for each hidden or output layer unit j{
$I_i = \sum_i W_{ij} O_i + \Theta_j$; //compute the net input of unit j with respect to the previous layer ,i
$O_j = \frac{1}{1+e^{-I_j}}$;}
//Backpropagate the errors;
for each unit j in the output layer
$\mathbf{Err_j = O_j (1-O_j) (T_j-O_j);}$ //Compute the error
for each unit j in the hidden layers, from the last to the first hidden layer

E. OPTIMAL ESTIMATION

Feature tracking has been extensively studied in computer vision. In this context, the optimal estimation framework provided by the Kalman filter has been widely employed in turning observations (feature detection) into estimations (extracted trajectory).The reasons for its popularity are real-time performance, treatment of uncertainty, and the provision of predictions for the successive frames. Vision based hand gesture recognition for human computer interaction the image, based on a hypothesis formulation and validation/rejection scheme. The problem of multiple blob tracking [11] was investigated in where blob tracking is performed in both images of a stereo pair and blobs are

corresponded, not only across frames, but also across cameras. The obtained stereo information not only provides the3D locations of the hands, but also facilitates the potential motion of the observing stereo pair which could be thus mounted on a robot that follows the user. Snakes integrated with the Kalman filtering framework have been used for tracking hands. Robustness against background clutter is achieved where the conventional image gradient is combined with optical flow to separate the foreground from the background.

F. TEMPLATE BASED

This class of methods exhibits great similarity to methods for hand detection. Members of this class invoke the hand detector at the spatial vicinity that the hand was detected in the previous frame, so as to drastically restrict the image search space. The implicit assumption for this method to succeed is that images are acquired frequently enough.

1. Correlation based feature tacking

Correlation-based template matching is utilized to track hand features across frames. Once the hand(s) have been detected in a frame, the image regions in which they appear is utilized as the prototype to detect the hand in the next frame. Again, the assumption is that hands will appear in the same spatial neighborhood. This technique is employed for to obtain characteristic patterns or signatures of gestures, as seen from a particular view. A target is viewed under various lighting conditions. Then, a set of basis images that can be used to approximate the appearance of the object viewed under various illumination conditions is constructed. Tracking simultaneously solves for the affine motion of the object and the illumination.

2. Contour based tracking

Deformable contours or “snakes” have been utilized to track hand regions in successive image frames. Typically, the boundary of this region is determined by intensity or color gradient. Nevertheless, other types of image features (e.g. texture) can be considered. The technique is initialized by placing a contour near the region of interest. The contour is then iteratively deformed towards nearby edges to better fit the actual hand region. This deformation is performed through the optimization of energy functional that sums up the gradient at the locations of the snake while, at the same time, favoring the smoothness of the contour. When snakes are used for tracking, an active shape model is applied to each frame and the convergence of the snake in that frame is used as a starting point for the next frame. Snakes allow for real-time tracking and can handle multiple targets as well as complex hand postures. They exhibit better performance when there is sufficient contrast between the background and the object. On the contrary, their performance is compromised in cluttered backgrounds. The reason is that the snake algorithm is sensitive to local optima of the energy function, often due to ill foreground/background separation or large object displacements and/or shape deformations between successive images

IV. GESTURE RECOGNITION

A. HIDDEN MARKOV MODEL (HMM)

Since the speech recognition field adopted HMMs with great success many years ago, these techniques are just now entering the computer vision area where time variation is significant. Hidden Markov models have intrinsic properties which make them attractive for gesture recognition, and also explicit segmentation is not necessary for either training or recognition. A hidden Markov model is a collection of finite states connected by transitions [12]. Each state is characterized by two sets of probabilities: a transition probability, and either a discrete output probability distribution or a continuous output probability density function. HMM can be defined by: (1) A set of states {S} with an initial state Si and a final state SF; (2) The transition probability matrix, A = [a$, where u_{ij} is the transition probability of taking the transition from state i to state j; (3) The output probability matrix 3.For a discrete HMM, 3 = @ j(Ok)),where ok represents discrete observation symbol. For a continuous HMM, B = @j(x)), where x represents continuous observations of k-dimensional random vectors. With the initial state distribution r_x = in), the complete parameter set of the HMM can be expressed compactly as h = (A, 3, IT). HMM can be based either on discrete observation probability distributions or continuous mixture probability density functions.

In the discrete HMM, the discrete probability distributions are sufficiently powerful to characterize random events with a reasonable parameters. The principal advantage of continuous HMM is the ability to directly model the parameters of a continuous signal. A semi-continuous HMM provides a framework for unifying the discrete and continuous HMMs.

In our research, we only consider a discrete HMM. There are three basic problems that must be solved for the real application of HMM:1) evaluation, 2) decoding and 3) learning. The solutions to these three problems are the Forward-Backward algorithm, the Viterbi algorithm, and the Baum-Welch algorithm. The evaluation problem is that given an observation sequence and a model, what is the probability that the observed sequence is generated by the model (P(0I hi)).If this can be calculated for all competing models for an observation sequence, then the model with the highest probability can be selected as the 1st objective gesture model. In our research, the observation symbol 0 is represented as the sequence of O_t.

$$O = O_1, O_2, ...O_{t-1}$$

where O_{t-1} is the 8 directional chain code between time t-2 and t. Therefore our problem is to find the gesture model with the highest probability on given O.

B.PRINCIPAL COMPONENT ANALYSIS

Principal component analysis (PCA) is a standard tool in modern data analysis in diverse fields from neuroscience to computer graphics – because it is simple, non-parametric method for extracting relevant information from confusing datasets. With minimal effort PCA provides a roadmap for how to reduce a complex data set to a lower dimension to reveal the sometimes hidden, simplified structures that often underlie it [15]. It is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences. Since patterns in data can be hard to find in data of high dimension, where the luxury of graphical representation is not available, PCA is a powerful tool for analyzing data. The other main advantage of PCA is that once you have found these patterns in the data, and you compress the data by reducing the number of dimensions without much loss of information. This technique is used in image compression.

PCA is rather general statistical technique that can be to reduce the dimensionality of feature space. The idea behind PCA is quite old. Pearson introduced first it in 1901 as linear regression. Hotelling then proposed it for the purpose of revealing the correlation structures behind many random variables. During the 1940's, Karhunen and Loeve independently extended it into a continuous version by using K-L Transform (KLT).

C. SUPPORT VECTOR MACHINES

Support vector machines are used to classify the feature vectors derived from the shape context cost matrix instead of making it subject to bipartite graph matching. We classify the vector of the matching costs ($v_\Delta(\alpha,\beta)$) between the shape context features extracted from two different hand shape contours α and β in order to decide whether they represent the same pose. This increases the scalability compared with the conventional multi-class approach, in which every gesture constitutes a separate class, However, this requires processing training sets that contain a large number of matching cost vectors which must be reduced due to the high $O(n^3)$ time complexity of the SVM training. It is a non-linear classifier.

Based on a labeled training set, SVM's determine a hyperplane that linearly seperates two classes in a higher dimensional kernel space. The hyperplane, later used to classify the data, is defined by a small subset of the vectors from the entire training set, termed support vectors (SV). The problem of using large data sets for the SVM training has been already identified. The simplest method consists in a random selection of the training subset (RS-SVM). In many cases it is possible to find a representative sample in this way, provided that a sufficiently large number of draws are proceeded. Random sampling was further extended to reduced support vector machines , where the SV's are selected from a randomly chosen small subset, but the entire training set is used to determine the separating hyperplane. However although these methods reduce the training complexity, they are still highly dependent on the training set size which reduces their applicability. SVM is a nonlinear classifier [16].

D. CONDENSATION ALGORITHM

Condensation Algorithm Our goal is to take a set of M model trajectories $\{m^{\mu}, \mu\text{-}1\ldots M\}$ and match them against an N-dimensional input trajectory, given by $z_t\text{-}(z_{t,1},\ldots z_{t,N})$ at time t . The models are taken to be discretely sampled curves (though they maybe continuous as well) with a phase parameter $\Phi \in [0, \Phi_{max}]$ representing the current position in the model. The model values at position Φ are a vector of N values $m_\Phi{}^\mu$- $(m_{\Phi,1}{}^\mu \ldots m_{\Phi,N}{}^\mu)$where the stored discrete Curve is linearly interpolated at phase Φ . The parameters we need to estimate to match a model to the data are:

μ: an integer indicating the model type that is being matched,
Φ: position (or phase) with in the model that align sit with the data at time t,
α: an amplitude parameter used to scale the model vertically to match the data, and
∂: a rate parameter that is used to scale the model in the time dimension.

We define a state at time to be a vector of parameters $s_t=(\mu,\Phi,\alpha,\partial)$.We would like to find the $states_t$ that is most likely to have given rise to the observed data $Z_t=(z_t;_{zt-1,\ldots\ldots})$.Let $Z_{t;i}= (z_{t;I}, z_{(t-1);I}, z_{(t-2)};I,\ldots.)$be the vector of observations for the i^{th} coeffcient over time .We define the likelihood of an observation z_t given the $states_t$ as

$$\mathbf{p(z_t|s_t) - \prod^{N}_{i=1} p(Z_{t,i}|s_t)},$$

and where w is the size of a temporal window backwards in time over which we want the curves to match. The σ_i are estimates of the standard deviation for curve i. Also, $\alpha m^{\mu}{}_{(\Phi\text{-}\partial j),i}$ is simply the value of the i^{th} coefficient in the model μ interpolated at time $\Phi\text{-}\partial j$ and scaled by α. Given a definition for $p(z_t|s_t)$, we can construct a discrete representation of the entire probability distribution over the possible states. Initially we sample uniformly for the state parameters

$\mu \in [0, \mu_{max}]$

$$\Phi\text{-}\ \frac{1 - \sqrt{y}}{\sqrt{y}} \quad \text{where } y \in [0,1]$$

$\alpha \in [\alpha_{min}, \alpha_{max}]$
$\partial \in [\partial_{min}, \partial_{max}]$

The Condensation algorithm [17] uses this information (the sample states and their weights) to predict the entire probability distribution at the next time instant. Unlike traditional tracking methods (e.g. Kalman filtering) this approach can deal well with ambiguous data since multiple matches are propagated in time.

E. DYNAMIC TIME WARPING

It has long been used to find the optimal alignment of two signals. The DTW algorithm [18] calculates the distance between each possible pair of points out of two signals in terms of their associated feature values. It uses these distances to calculate a cumulative distance matrix and finds the least expensive path through this matrix. This path represents the ideal warp—the synchronization of the two signals which causes the feature distance between their synchronized points to be minimized. Usually, the signals are normalized and smoothed before the distances between points are calculated. DTW has been used in various fields, such as speech recognition, data mining, and movement recognition [19]. Previous work in the field of DTW mainly focused on speeding up the algorithm, the complexity of which is quadratic in the length of the series. Eamonn and Pazzani [20] proposed a form of DTW called Derivative DTW (DDTW). Here, the distances calculated are not between the feature values of the points, but between their associated first order derivatives. In this way, synchronization is based on shape characteristics (slopes, peaks) rather than simple values. Most work, however, only consideredOne-dimensional series.

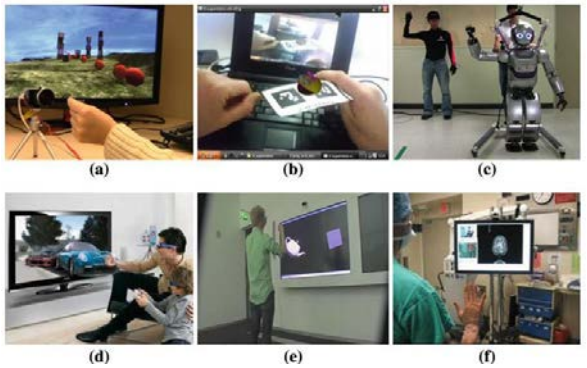
F.FINITE STATE MACHINES

A finite state machine [21] is one that has a limited or finite number of possible states (an infinite state machine can be conceived but is not practical). A finite state machine can be used both as a development tool for approaching and solving problems and as a formal way of describing the solution for later developers and system maintainers. There are a number of ways to show state machines, from simple tables through graphically animated illustrations. Usually, the training of the model is done off-line, using many possible examples of each gesture as training data, and the parameters (criteria or characteristics) of each state in the FSM are derived. When input data (feature vectors such as trajectories) are supplied to the gesture recognizer, the latter decides whether to stay at the current state of the FSM or jump to the next state based on the parameters of the input data. If it reaches a final state, we say that a gesture has been recognized.

G. **APPLICATION DOMAINS**

Vision based hand gesture recognition systems has a wide range of real time applications. The advanced applications include other related application domains like tablet PC, games, medicine environment, augmented reality.

Early work on hand gestures concentrates on smart home applications like on/off the light control, electronic equipments control (i.e.) controls the TV (change channel, change volume, pause video, surf), music player control etc. Hand gestures for virtual reality applications include navigation and manipulation tasks. Hand gestures to arrange virtual objects and to navigate around 3D information space such as a graph, using a stereoscopic display, control of Google Earth using mouse simulation local feature classifier. In army and defense military air marshals use hand and body gestures to direct flight operations. Gestures can control a robots hand [22] and arm movements to reach for and manipulate actual objects, as well its movement through the world. Finally, we look at hand gestures for computer games. We track a player's hand or body position [23] to control movement and orientation of interactive game objects such as cars. We can use gestures to control the movement of avatars in a virtual world. Augmented realty applications [24] often user markers, consisting of patterns printed on physical objects, which can more easily be tracked using computer vision, and that are used for displaying virtual objects in augmented reality displays.



Hand gestures are also used in computer supported collaborative work (CSCW) applications to enable multiple users to interact with a shared display, using a variety of computing devices such as desktop or tabletop. Notes and annotations can be shared within groups using strokes either locally or for remote interactions. Annotations can be transmitted using live video streams, to enable remote collaborations between students and instructor.

Hand gestures can enable eyes-free interactions with mobile devices that allow users to focus their visual attention on their primary task [25]. PDA's augmented with touch sensitive screens can also interpret finger gestures or strokes as input, and provide audio output to the user to support eyes-free interactions.

Vision based hand gesture recognition for human computer interaction with mobile devices. Gestures are used with telematics to enable secondary task interactions to reduce the distraction caused to the primary task of driving. A number of hand gesture recognition techniques for human vehicle interface have been proposed time to minimize distraction while driving. The primary motivation of research into the use of hand gestures for in-vehicle secondary controls is broadly based on the premise that taking the eyes off the road to operate conventional secondary controls can be reduced by using hand gestures.

Computer information technology is increasingly penetrating into the medical domain. It is important that such technology be used in a safe manner to avoid serious mistakes leading to possible fatal incidents. Keyboards and mice are today's principle method of human computer interaction. Unfortunately, it has been found that a common method of spreading infection from one person to another involves computer keyboards and mice in intensive care units (ICUs) used by doctors and nurses [26]. In a setting like an operating room (OR), touch screen displays must be sealed to prevent the buildup of contaminants, and require smooth surfaces for easy cleaning with common cleaning solutions. A gesture plays an important role in such situations for interaction with computing devices. A surgeon can control the motion of the laparoscope by simply making the appropriate face gesture, without hand or foot switches or voice input. Surgeons are enabled to perform standard mouse functions like pointer movement and button presses with hand gestures.

V. CONCLUSION

Hand gesture recognition for real time applications is very challenging because of its requirements on the robustness, accuracy and efficiency. This survey has identified more publications in many journals and conferences. Increased activity in this research area has been a driven by both scientific challenge of recognizing hand gestures and the demands of potential applications related to desktop and tablet PC applications, virtual reality etc. This survey is an endeavor to provide the upcoming researchers in the field of human computer interaction a brief overview of the core technologies related to and worked upon in the recent years of research. There are well known limitations of the core technologies that need to be addressed and provide the scope for future research and development.

Analysis of the comprehensive surveys and articles indicates that the techniques implemented for hand gesture recognition are often sensitive to poor resolution, frame rate, drastic illumination conditions, changing weather conditions and occlusions among other prevalent problems in the hand gesture recognition systems. The survey enlists some the common enabling technologies of hand gesture recognition and advantages and disadvantage related to them. The state of art for applications of the hand gesture recognition systems present desktop applications to be the most implemented application for hand gesture recognition systems.

Future research in the field of hand gesture recognition systems provide an opportunity for the researchers to come up with efficient systems overcoming the disadvantages associated with the core technologies in the current state of art for enabling technologies gesture representations and gesture recognition systems as a whole.

VI. FUTURE WORK

Future work for this paper may include 1) develop more efficient and more robust algorithms to solve false labeling and false merging problems of hand tracking through interaction and occlusion. 2) extend the laboratory task to increase the pool of participating users. Ideally users with physical impairments can participate and provide feedback about the usability, learning and adaptability to the interface suggested. Classification rate and accuracy can be improved by employing a better classifier.

REFERENCES

[1] Vanco, M; Minarik. I; Rozinaj G, "Dynamic gesture recognition for next generation home multimedia,"ELMAR, 201355th International Symposium, Vol., no.,pp 219, 222, 25-27 Sept.2013.

[2] B.Bauer and H.Hienz, "Relevant features for Video based continuous Sign Language Recognition", Proceedings of the Fourth IEEE International Conference on Automatic face and Gesture Recognition, pp.64-75, 2000.

[3] R.Liang, M.Ouhyoung, A Real time Gesture Recognition system for sign language, Proc.ThirdIntConf.Autom.Face Gesture Recognition.(1998)

[4] G.R.S Murthy&R.S.Jadon, A Review of vision-based hand gestures recognition,International Journal of Information Technology and Knowledge Management.

[5] Y.Fang, K.Wang, J.Cheng&H.Lu, A Real time hand gesture recognition method, IEEE 2007.

[6] Hairong Jiang, Bradley SDuerstock, Juan P.Wachs Member, IEEE,"A Machine Vision-Based Gestural interface People With Upper Extremity Physical Impairments,System, Man and Cybernatics.

[7] SiddharthS.Rautaray AnupamAgrawal,"Vision Based hand gesture recognition for human Computer interaction"Springer 2012.

[8] A.Doucet, N.De Freitas, N.Gordan et al., Sequential Monte Carlo Methods inPractice, vol 1, Springer New York 2001.

[9] J.Corander, M.Ekdahl and T.Koski, Parallel interacting mcmc for learningof topologies of graphical models" Data Mining and Knowledge Discovery, vol 17, No 3, pp.431-456, 2008.

[10] G.R.Bradski, "Computer vision face tracking as a

Component of a perceptual user interface" in Proc. Workshop Application Computer Vision, 1998, pp.214-219
[11] Argyros A, Lourakis MIA (2004a) Real-time tracking of multiple skin-colored objects with a possibly moving camera. In: Proceedings of the European onference on computer vision, Prague, pp 368–379
[12] J. Yang, et al., "Human Action Learning via Hidden Markov Model, "IEEE Trans. On Systems, Man, and Cybernetics, Vol. 27, No.1, pp. 34-44, January 1997.
[13] Lungociu Corneliu, "Real time Sign Language Recognition using Artificial Neural Networks" in studia. Univ babes-bolyai, Informatica, Volume Lvi , November 4, 2011.
[14] Peter Wray Vamplew, PhD Thesis, "Recognition of Sign Language using Neural Networks" for flinders University of South Australia, 1990.
[15] Lalitakumari, Swapan Debbarma, Nikhil Debbarma, Suman Deb, Image Pattern Matching using Principal Component Analysis Method International Journal of Advanced Engineering & application, June 2011.
[16] Burges CJC (1998) A tutorial on support vector machines for pattern recognition. Kluwer, Boston 1–43.
[17] Michael J. Black, Allan D. Jepson, "A Probabilistic framework for matching temporal trajectories CONDENSATION based recognition of gestures and expressions" Springer-Verlag Berlin Heidelberg 1998.
[18] Senin P (2008) Dynamic time warping algorithm review, technical report. http:// csdl.ics.hawaii.edu/tech reports/08-04/ 08-04.pdf.
[19] Andrea C (2001) Dynamic time warping for offline recognition of a small gesture vocabulary. In: Proceedings of the IEEE ICCV workshop on recognition, analysis, and tracking of faces and gestures in real-time systems, July–August, p 83.
[20] Eamonn K, Pazzani MJ (2001) Derivative dynamic time warping. In: First international SIAM international conference on data mining, Chicago.
[21] Holzmann GJ (1925) Finite state machine: Ebook.http://www.spinroot.com/spin/Doc/Book91_PDF/F1.pdf.
[22] Goza SM, Ambrose RO, Diftler MA, Spain IM (2004) Telepresence control of the nasa/darpa robonaut on a mobility platform. In: Conference on human factors in computing systems. ACM Press, pp 623–629.
[23] Freeman W, Tanaka K, Ohta J, Kyuma K (1996) Computer vision for computer games. In: Proceedings of the second international conference on automatic face and gesture recognition, pp 100–105.
[24] Buchmann V, Violich S, Billinghurst M, and Cockburn A (2004) Fingartips: gesture based direct manipulation in augmented reality. In: 2nd international conference on computer graphics and interactive techniques, ACM Press, pp 212–221.
[25] Schmandt C, Kim J, Lee K, and Vallejo G, Ackerman M (2002) Mediated voice communication via mobile ip.In: Proceedings of the 15th annual ACM symposium on User interface software and technology. ACM Press, pp 141–150
[26] Schultz M, Gill J, Zubairi S, Huber R, Gordin F (2003) Bacterial contamination of computer keyboards in a teaching hospital. Infect Control Hosp Epidemiol 4(24):302–303
[27] Graetzel C, Fong TW, Grange S, Baur C (2004) A non-contact mouse for surgeon-computer interaction. Technolgy Health Care 12(3):245–257.