**Theoretical Background**

**Introduction**

Imagine how nice it would be, the next time you make a presentation, if you did not need to stand close to your laptop, or use a remote control with its

limited functionality. What if you could present your work as naturally as having

a conversation with your audience. You swipe your hand left and right to change

slides. When you point to the slide with your hand, the display shows a cursor

following wherever you point. When you are showing a video, you use a palm

forward hand pose (\stop" gesture) to pause the movie, then move left and right

to fast forward or rewind the video. You can also say \faster" or \slower" to

change the video speed. When you need to jump to a particular slide, you make

a circle gesture to show all the slides, and say \show this slide" while pointing at

that slide. You can also make a dismiss gesture to pause the slide show (making

the screen black) to take the distracting slides o  
 the screen and get the full

**Human Computer Interaction**

HCI (Human-Computer Interaction) is the study of how people interact with computers and to what extent computers are or are not developed for successful interaction with human beings. A significant number of major corporations and academic institutions now study HCI. With the massive influx and advancement of technologies, a computer system has become a very powerful machine which has been designed to make the human beings’ tasks easier. Due to which the HCI (human – computer interaction) has become an important part of our lives. Now-a-days, the progress and development in interaction with computing devices has increased so fast that as a human being even we could not remained left with the effect of this and it has become our primary thing. Human-Computer-Interaction (HCI) is a key topic on the Accessibility area, namely in what concerns a Smart-City environment, where humans have to interact with artifacts spread all around one particular city. In the last decade HCI has experienced a significant evolution towards a desired fully natural user interface, for example in what concerns the ability to recognize a wide range of hand gestures in real-time. Some efforts have lately been made in the hardware sector in order to deploy sensors that may gather information about the human body movements, as detailed as possible. The technologies has so much surrounded us and has made a place in our lives that we use it to communicate, shop, work and even entertain ourselves[1]. There are many applications like media player, MS-office, Windows picture manager etc. which require natural and intuitive interface. Now-a-days most of the users use keyboard, mouse, pen, Joysticks etc. to interact with computers, which are not enough for them. In the near future, these existing technologies which are available for the computing, communication and display will become a bottleneck and the advancement in these technologies will be required to make the system as natural as possible.

Nevertheless the invention of mouse and keyboards by the researchers and engineers has been a great progress; there are still some situations where interaction with computer with the help of keyboard and mouse will not be enough. This is the case with the advancement in hand held devices like mobiles or i-pods or Tabs which are relatively very small in size. It’s very difficult to interact with them due to their determined input spaces and small touch screen or keyboard [2]. This is also the case of interacting 3D objects where these devices are incompatible for HCI. However from the recent years there has been an increased extent in trying to introduce other human-to-human communication modalities in HCI. So hand gestures can provide a natural and visceral alternative to some incompatible devices. We can use hand as a device to interact and communicate with computers as we do in our daily lives to interact with each other. We use our hand to point a person or an object, express or carry information about something, and to move, modify and transform an object.

The area of Accessibility depends, to some extent, on the interaction persons have with machines. An effort has thus been put on finding the means to improve this interaction through a dedicated Human Computer Interaction (HCI) research community. The HCI field is a multidisciplinary one that connects computer and social sciences. The goal of HCI is to create new interaction paradigms between persons and machines, developing descriptive models for these interfaces and the methods to evaluate them. Since HCI studies both the behavior of humans and machines [3], it gathers knowledge from both sides:

* On the machine side, skills like computer graphics, operating systems and programing languages are covered or addressed;
* On the human side the relevant skills are communication theory, linguistics, cognitive psychology and social sciences, among others.

**Definition of Gesture**

Gestures are an important aspect of human interaction, both interpersonally and in the context of man-machine interfaces. A gesture is a form of non-verbal communication in which visible bodily actions communicate particular messages, either in place of speech or together and in parallel with words. Gesture provides a way for computers to understand human body language. Gestures include movement of the hands, face, or other parts of the body. It can originate from any bodily motion or state but commonly originate from the face or hand. Many approaches have been made using cameras and computer vision algorithms to interpret sign language. It enables humans to interface with the machine (HMI) and interact naturally without any mechanical devices.

Bobick and Wilson [4] have defined gestures as “The motion of the body that is intended to communicate with other agents. For a successful communication, a sender and a receiver must have the same set of information for a particular gesture.”

Webster's Dictionary defines gestures as “A movement usually of the body or limbs that expresses or emphasizes an idea, sentiment, or attitude." This definition is particularly related to the communicative aspect of the human hand and body movements. However, in HCI, the notion of gestures is somewhat different. In their review of the visual interpretation of hand gestures for HCI, Pavlovic et al. [5] state that in a computer controlled environment one wants to use the human hand to perform tasks that mimics the natural use of the hand both as a manipulator and as used in human-machine communication. They describe this in part by having both manipulative and communicative gestures in their gesture taxonomy.

We adopt this distinction.

**Gesture**

**(The mental concept)**

**Hand/Arm**

**Movement**

**Visual Image**

Gesture

Produces

Observer

Perceives

Figure 1-3: Production and perception of gestures. Hand gestures originate as a mental concept, are expressed through arm and hand motion, and are perceived as visual images [5].

Pavlovic et al. [5] also gives a model (Figure. 1-3) for the production and perception of gestures, based on the model used in the field of spoken language recognition. According to their model, gestures originate as a gesturer's mental concept, possibly in conjunction with speech. They are expressed through the motion of arms and hands, while observers perceive gestures as streams of visual images that they interpret using their knowledge about those gestures. In HCI, the observer is the computer and the knowledge it possesses is the training data.

**Gesture Recognition**

Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Gesture recognition is an important skill for robots that work closely with humans. It is especially valuable in applications involving interaction human/robot for several reasons. It Deals with the goal of interpreting human gestures via mathematical algorithms. It is required to provide a way to explore the use of gestures in HCI so that it can be interpreted by computers. To facilitate and accomplish the advanced interaction between humans and computers, the designing of some special input devices has been found to be of great care in this area. The aggregation of traditional devices (i.e. keyboard, mouse etc.) with the new designed interaction devices such as face and gesture recognition, haptic sensors and tracking devices provides flexibility in Virtual Reality (VR), cars system control, Tele-operating, robot control, text editing etc. Gesture recognition has gained a lot of importance since few years. Various applications can be controlled using gestures. Face gestures like lip movements is used to recognize the language spoken, hand gestures are used in various applications like gaming, mouse control etc. In an application like robot control using hand gestures, the robot responds to hand gestures given by the human. This hand sign of humans is visually observed by robot through a camera. The algorithm that enables the robot to identify the hand gesture from the image is of interest. Each gesture corresponds to a particular command. The command that is identified will be used to control the robot to perform certain action or to execute a certain task. Different gestures will have different meaning associated with them.

**Gesture types**

Gestures usually made with the hand, but in some notable cases with other parts of the body entirely form a symbolic, non-vocal language; a shorthand way of sending a message without the need for words. There are three main types of gestures: adaptors, emblems, and illustrators [g1].

**Adaptors:**

Adaptors are touching behaviors and movements that indicate internal states typically related to arousal or anxiety. Adaptors can be targeted toward the self, objects, or others. In regular social situations, adaptors result from uneasiness, anxiety, or a general sense that we are not in control of our surroundings. Many of us subconsciously click pens, shake our legs, or engage in other adaptors during classes, meetings, or while waiting as a way to do something with our excess energy. Public speaking students who watch video recordings of their speeches notice nonverbal adaptors that they didn’t know they used. In public speaking situations, people most commonly use self- or object-focused adaptors. Common self-touching behaviors like scratching, twirling hair, or fidgeting with fingers or hands are considered self-adaptors. Some self-adaptors manifest internally, as coughs or throat-clearing sounds. My personal weakness is object adaptors. Specifically, I subconsciously gravitate toward metallic objects like paper clips or staples holding my notes together and catch myself bending them or fidgeting with them while I’m speaking. Other people play with dry-erase markers, their note cards, the change in their pockets, or the lectern while speaking. Use of object adaptors can also signal boredom as people play with the straw in their drink or peel the label off a bottle of beer. Smartphones have become common object adaptors, as people can fiddle with their phones to help ease anxiety. Finally, as noted, other adaptors are more common in social situations than in public speaking situations given the speaker’s distance from audience members. Other adaptors involve adjusting or grooming others, similar to how primates like chimpanzees pick things off each other. It would definitely be strange for a speaker to approach an audience member and pick lint off his or her sweater, fix a crooked tie, tuck a tag in, or pat down a flyaway hair in the middle of a speech.

**Emblems:**

Emblems are gestures that have a specific agreed-on meaning. These are still different from the signs used by hearing-impaired people or others who communicate using American Sign Language (ASL). Even though they have a generally agreed-on meaning, they are not part of a formal sign system like ASL that is explicitly taught to a group of people. A hitchhiker’s raised thumb, the “OK” sign with thumb and index finger connected in a circle with the other three fingers sticking up, and the raised middle finger are all examples of emblems that have an agreed-on meaning or meanings with a culture. Emblems can be still or in motion; for example, circling the index finger around at the side of your head says “He or she is crazy,” or rolling your hands over and over in front of you says “Move on.”

Figure 2: Emblems are gestures

Just as we can trace the history of a word, or its etymology, we can also trace some nonverbal signals, especially emblems, to their origins. Holding up the index and middle fingers in a “V” shape with the palm facing in is an insult gesture in Britain that basically means “up yours.” This gesture dates back centuries to the period in which the primary weapon of war was the bow and arrow. When archers were captured, their enemies would often cut off these two fingers, which was seen as the ultimate insult and worse than being executed since the archer could no longer shoot his bow and arrow. So holding up the two fingers was a provoking gesture used by archers to show their enemies that they still had their shooting fingers [g2].

**Illustrators:**

Illustrators are the most common type of gesture and are used to illustrate the verbal message they accompany. For example, you might use hand gestures to indicate the size or shape of an object. Unlike emblems, illustrators do not typically have meaning on their own and are used more subconsciously than emblems. These largely involuntary and seemingly natural gestures flow from us as we speak but vary in terms of intensity and frequency based on context. Although we are never explicitly taught how to use illustrative gestures, we do it automatically. Think about how you still gesture when having an animated conversation on the phone even though the other person can’t see you.

**Hand Gesture Types:**

Gesture recognition is the process of recognizing and interpreting a stream continuous sequential gesture from the given set of input data. So basically hand gestures are of two types.

* Contact based Hand Gesture
* Non-Contact based Hand Gesture

**Contact based Hand Gesture:**

Touch screens are growing rapidly in popularity as an input method for smart phones and other mobile devices [1]. These devices have few, if any, hard buttons; instead soft buttons are the dominant command invocation paradigm on commercial touch-based phones, and are effective when the user is seated and focusing directly on the phone, with no distractions (an ideal environment). However, users rely on mobile devices while sitting, walking, driving, and in diverse environments with various distraction levels [2]. When the user is in a non-ideal environment, such as walking through an airport looking for a currency exchange, he/she must navigate, maintain awareness of his/her location and avoid obstacles. These situational impairments [3] are environmental factors that inhibit the ability to perform tasks on a device.

In this regard, direct touch gestures offer some possible advantages. Some direct-touch gestures have the potential to be articulated eyes-free or with reduced visual monitoring, making them more resilient to distraction. Gestures can be committed to muscle memory, which helps users focus on their task [4], and it is possible to articulate some gestures with one hand. Gestures also require no dedicated screen space, which is a limited resource on mobile phones.

**Touch based Gestures**

**Vision based gesture**

**Hand Gesture type**

**Non-Contact based**

**Contact based**

**Device based Gesture**

**Static & Dynamic**

**Static**

Figure 3: Hand Gesture types

**Non-Contact based gesture:**

Non-Contact based gesture is defined as an expressive movement of body parts which has a particular message, to be communicated precisely between a sender and a receiver. It has two categories- one is Device based Gesture and another is Vision based Gesture.

Scientifically gesture categorized into two distinctive categories: dynamic and static [x1]. Static gestures refer to the orientation and position of the hand in space during an amount of time without any movement. Dynamic gestures refer to the same but with movement. Dynamic gestures include those involving body parts, such as waving the hand , whereas static gestures include single formation without movement, such as jamming the thumb and forefinger to form the OK symbol (i.e., a static pose). According to[x2], 35 % of human communication consists of verbal communication, and 65 % is nonverbal gesture-based communication. Gestures can be classified into five types: emblems, affect displays, regulators, adaptors, and illustrators [x3]. Emblematic, emblem, or quotable gestures are direct translations of short verbal communication, such as waving the hand for goodbye or nodding for assurance. Quotable gestures are culture-specific. Gestures conveying emotion or intention are called affect displays. Affect displays generally depend less on culture. Gestures that control interaction are called regulators. Gestures such as head shaking and quickly moving the leg to release body tension are called adaptors, which are generally habits unintentionally used during communication. Illustrator gestures emphasize key points in speech and thus inherently depend on the thought process and speech of the communicator. Illustrator gesticulations can further be classified into five sub categories: beats, deictic gestures, iconic gestures, metaphoric gestures, and cohesive gestures [x3]. Beats are short, quick, rhythmic, and often repetitive gestures. Pointing to a real location, object, or person or to an abstract location or period of time is called deictic gesture. Hand movements that represent figures or actions, such as moving the hand upward with wiggling fingers to depict tree climbing, are called iconic gestures. Abstractions are depicted by metaphoric gestures. Thematically related but temporally separated gestures are called cohesive gestures. The temporal separation of these thematically related gestures is due to the interruption of the communicator by another communicator.

**Computer Vision Techniques for Hand Gesture Recognition**

Most of the complete hand interactive systems can be considered to be comprised of three layers: detection, tracking and recognition. The detection layer is responsible for defining and extracting visual features that can be attributed to the presence of hands in the field of view of the camera. The tracking layer is responsible for performing temporal data association between successive image frames, so that, at each moment in time, the system may be aware of “what is where”. Moreover, in model-based methods, tracking also provides a way to maintain estimates of model parameters, variables and features that are not directly observable at a certain moment in time.

Last, the recognition layer is responsible for grouping the spatiotemporal data extracted in the previous layers and assigning the resulting groups with labels associated to particular classes of gestures. In this section, research on these three identified sub problems of vision-based gesture recognition is reviewed.

**2.1 Detection**

The primary step in gesture recognition systems is the detection of hands and the segmentation of the corresponding image regions. This segmentation is crucial because it isolates the task-relevant data from the image background, before passing them to the subsequent tracking and recognition stages. A large number of methods have been proposed in the literature that utilize a several types of visual features and, in many cases, their combination. Such features are skin color, shape, motion and anatomical models of hands. In [6], a comparative study on the performance of some hand segmentation techniques can be found.

**2.1.1 Color**

Skin color segmentation has been utilized by several approaches for hand detection. A major decision towards providing a model of skin color is the selection of the color space to be employed. Several color spaces have been proposed including RGB, normalized RGB, HSV, YCrCb, YUV, etc. Color spaces efficiently separating the chromaticity from the luminance components of colors are typically considered preferable. This is due to the fact that by employing chromaticity-dependent components of color only, some degree of robustness to illumination changes can be achieved. They typically eliminate the luminance component, to remove the effect of shadows, illumination changes, as well as modulations of orientation of the skin surface relative to the light source(s). The remaining 2D color vector is nearly constant for skin regions and a 2D histogram of the pixels from a region containing skin shows a strong peak at the skin color. Regions where this probability is above a threshold are detected and described using connected components analysis. In several cases (e.g. [7]), hysteresis thresholding on the derived probabilities is also employed prior to connected components labeling. The rationale of hysteresis thresholding is that pixels with relatively low probability of being skin-colored, should be interpreted as such in case that they are connected to pixels with high such probability.

Having selected a suitable color space, the simplest approach for defining what constitutes skin color is to employ bounds on the coordinates of the selected space [8]. These bounds are typically selected empirically, i.e. by examining the distribution of skin colors in a preselected set of images. Another approach is to assume that the probabilities of skin colors follow a distribution that can be learned either online or by employing an online iterative method [9]. The perceived color of human skin varies greatly across human races or even between individuals of the same race. Additional variability may be introduced due to changing illumination conditions and/or camera characteristics. Therefore, color-based approaches to hand detection need to employ some means for compensating for this variability. However, most of these methods are still sensitive to quickly changing or mixed lighting conditions. A simple color comparison scheme is employed in [DKS01], where the dominant color of a homogeneous region is tested as if occurring within a color range that corresponds to skin color variability. Other approaches [10] consider skin color to be uniform across image space and extract the pursued regions through typical region-growing and pixel-grouping techniques.

**2.1.2 Shape**

The characteristic shape of hands has been utilized to detect them in images in multiple ways. Much information can be obtained by just extracting the contours of objects in the image. If correctly detected, the contour represents the shape of the hand and is therefore not directly dependent on viewpoint, skin color and illumination. On the other hand, the expressive power of 2D shape can be hindered by occlusions or degenerate viewpoints. In the general case, contour extraction that is based on edge detection results in a large number of edges that belong to the hands but also to irrelevant background objects. Therefore, sophisticated post-processing approaches are required to increase the reliability of such an approach. In this spirit, edges are often combined with (skin) color and background subtraction/motion cues.

In the 2D/3D drawing systems of [11], the user's hand is directly extracted as a contour by assuming a uniform background and performing real-time edge detection in this image. Examples of the use of contours as features are found in both model and appearance based techniques. Local topological descriptors have been used to match a model with the edges in the image. In [12], the shape context descriptor is proposed, which characterizes a particular point location on the shape. This descriptor is the histogram of the relative polar coordinates of all other points. Detection is based on the assumption that corresponding points on two different shapes will ideally have a similar shape context. The descriptor has been applied to a variety of object recognition problems, with limited background clutter.

The approaches in [13] utilize curvature as a cue to fingertip detection. Another technique that has been employed in fingertip detection is template matching. Templates can be images of fingertips or fingers or generic 3D cylindrical models [14]. Such pattern matching techniques can be enhanced by using additional image features, like contours. The template-matching technique was utilized also in, with images of the top view of fingertips as the prototype. The pixel resulting in the highest correlation is selected as the position of the target object.

**2.1.3 Learning detectors from pixel values**

Significant work has been carried out on finding hands in grey level images based on their appearance and texture. The suitability of a number of classification methods for the purpose of view-independent hand posture recognition was investigated. Several methods [15] attempt to detect hands based on hand appearances, by training classifiers on a set of image samples. The basic assumption is that hand appearance differs more among hand gestures than it differs among different people performing the same gesture. Still, automatic feature selection constitutes a major difficulty. The work in [16], investigates the difference between the most discriminating features (MDFs) and the most expressive features (MEFs) in the classification of motion clips that contain gestures. It is argued that MEFs may not be the best for classification, because the features that describe some major variations in the class are, typically, irrelevant to how the sub-classes are divided. MDFs are selected by multi-class, multivariate discriminate analysis and have a significantly higher capability to catch major differences between classes. Their experiments also showed that MDFs are superior to the MEFs in automatic feature selection for classification.

The AdaBoost algorithm [17] provides a learning method for finding suitable collections of weak classifiers. For training, it employs an exponential loss function that models the upper bound of the training error. The method utilizes a training set of images that consists of positive and negative examples (hands and non-hands, in this case), which are associated with corresponding labels. Weak classifiers are added sequentially into an existing set of already selected weak classifiers in order to decrease the upper bound of the training error. It is known that this is possible if weak classifiers are of a particular form. AdaBoost was applied to the area of face and pedestrian detection [18] with impressive results. However, this method may result in an excessive number of weak classifiers. The problem is that AdaBoost does not consider the removal of selected weak classifiers that no longer contribute to the detection process. To create a labeled database of training images for the above tree structure, an automatic method [19] for performing grouping of images of hands at the same posture is proposed, based on an unsupervised clustering technique.

**2.1.4 3D model-based detection**

A category of approaches utilize 3D hand models for the detection of hands in images. One of the advantages of these methods is that they can achieve view independent detection. The employed 3D models should have enough degrees of freedom to adapt to the dimensions of the hand(s) present in an image. Different models require different image features to construct feature-model correspondences. Point and line features are employed in kinematic hand models to recover angles formed at the joints of the hand [20]. Hand postures are then estimated provided that the correspondences between the 3D model and the observed image features are well established. Various 3D hand models have been proposed in the literature. A full hand model is proposed which has 27 degrees of freedom (DOF) (6 DOF for 3D location/orientation and 21 DOF for articulation). In [21] a “Cardboard model” is utilized, where each finger is represented by a set of three connected planar patches. In [22], edge features in the two images of a stereoscopic pair are corresponded to extract the orientation of in-between joints of fingers. These are subsequently utilized for model based tracking of the hands. Artificial neural networks that are trained with body landmarks are utilized for the detection of hands in images. In [23], the process is enhanced with anatomical data of the human hand that are incorporated into the model. Also the hand model to an image of a real hand, characteristic points on the hand are identified in the images, and virtual springs are implied which pull these characteristic points to goal positions on the hand model.

**2.1.5 Motion**

Motion is a cue utilized by a few approaches to hand detection. The reason is that motion-based hand detection demands for a very controlled setup, since it assumes that the only motion in the image is due to hand movement. Indeed, early works (e.g. [24]) assumed that hand motion is the only motion occurring in the imaged environment. In more recent approaches, motion information is combined with additional visual cues. In the case of static cameras, the problem of motion estimation reduces to that of background maintenance and subsequent subtraction. For example in [25] such information is utilized to distinguish hands from other skin-colored objects and cope with lighting conditions imposed by colored lights. The difference in luminance of pixels from two successive images is close to zero for pixels of the background. By choosing and maintaining an appropriate threshold, moving objects are detected within a static scene.

In [26], a novel feature, based on motion residue, is proposed. Hands typically undergo non-rigid motion, because they are articulated objects. Consequently, hand detection capitalizes on the observation that for hands, interface appearance changes are more frequent than for other objects such as clothes, face and background.

**2.2 Tracking**

Tracking or the frame-to-frame correspondence of the segmented hand regions or features is the second step in the process towards understanding the observed hand movements. The importance of robust tracking is twofold. First, it provides the inter-frame linking of hand finger appearances, giving rise to trajectories of features in time. These trajectories convey essential information regarding the gesture and might be used either in a raw form (e.g. in certain control applications like virtual drawing the tracked hand trajectory directly guides the drawing operation) or after further analysis (e.g. recognition of a certain type of hand gesture). Second, in model-based methods, tracking also provides a way to maintain estimates of model parameters variables and features that are not directly observable at a certain moment in time.

**2.2.1 Template based tracking**

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. Correlation-based feature tracking is directly derived from the above approach. In [27] 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 a static camera, to obtain characteristic patterns (or “signatures”) of gestures, as seen from a particular view. Real-time performance is achieved by pre-computing “motion templates” which are the product of the spatial derivatives of the reference image to be tracked and a set of motion fields. Some approaches detect hands as image blobs in each frame and temporally corresponding blobs that occur in proximate locations across frames. Approaches that utilize this type of blob tracking are mainly the ones that detect hands based on skin color, the blob being the correspondingly segmented image region. Blob-based approaches are able to retain tracking of hands even when there are great variations from frame to frame. Extending the above approach, 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.

**2.2.2 Optimal estimation techniques**

Feature tracking has been extensively studied in computer vision. In this context, the optimal estimation framework provided by the Kalman filter [28] 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. In [29], the target is retained against cases where hands occlude each other, or appear as a single blob in the image, based on a hypothesis formulation and validation/rejection scheme. The problem of multiple blob tracking was investigated, 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 the 3D 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. Robustness against background clutter is achieved, where the conventional image gradient is combined with optical flow to separate the foreground from the background.

In order to provide accurate initialization for the snake in the next frame, the work in [30], utilizes the optical flow to obtain estimations of the direction and magnitude of the target's motion. The success of combining optical flow is based on the accuracy of its computation and, thus, the approach is best suited for the case of static cameras.

**2.2.4 Particle filtering**

Particle filters have been utilized to track the position of hands and the configuration of fingers in dense visual clutter. In this approach, the belief of the system regarding the location of a hand is modeled with a set of particles. The approach exhibits advantages over Kalman filtering, because it is not limited by the uni-modal nature of Gaussian densities that cannot represent simultaneous alternative hypotheses. A disadvantage of particle filters is that for complex models (such as the human hand) many particles are required, a fact which makes the problem intractable especially for high-dimensional models. Therefore, other assumptions are often utilized to reduce the number of particles. For example, dimensionality is reduced by modeling commonly known constraints due to the anatomy of the hand. Additionally, motion capture data are integrated in the model. In [31] a simplified and application specific model of the human hand is utilized.

The CONDENSATION algorithm [32] which has been used to learn to track curves against cluttered backgrounds, exhibits better performance than Kalman filters, and operates in real-time. It uses “factored sampling”, previously applied to the interpretation of static images, in which the probability distribution of possible interpretations is represented by a randomly generated set. Condensation uses learned dynamical models, together with visual observations, to propagate this random set over time. The result is highly robust tracking of agile motion. The “partitioned sampling” technique is employed to avoid the high computational cost that particle filters exhibit when tracking more than one object. The state space is limited to 2D translation, planar rotation, scaling and the number of outstretched fingers. In general, contour tracking techniques, typically, allow only a small subset of possible movements to maintain continuous deformation of contours. This limitation was overcome to some extent in, who describe an adaptation of the CONDENSATION algorithm for tracking across discontinuities in contour shapes

Chapter 4

Discussion

In this report we have done two things. Firstly, we have tried to give a brief

overview over the different technologies which are applied in wearable HCI in

order to include gestures. Secondly, we have tried to give a brief overview of

computer vision-based gesture recognition.

We were surprised to see the great variety of different technologies used for

including gestures into HCI. One reason might be the fact that the current state of

the art in computer vision-based gesture recognition is not impressive.

Besides getting a good overview, reading through the literature also allowed us

to identify desirable characteristics of a technology used in wearable HCI. These

are listed below.

1. Robust initialization and reinitialization: The hand can be expected to

enter and exit from the view frequently. Therefore the tracker must be able

to quickly reinitialize itself, and a reliable estimation of whether the hand is

present or not must be obtainable.

2. Robustness to background clutter: Objects in the background should not

distract the tracker, not even if these objects are of skin color.

3. Independence of illumination: As the tracker is to be used in wearable

applications, it must be able to cope with changing and mixed lighting con-

ditions.

4. Computationally effective: Mobile processors tend to be signiﬁcantly less

powerful than their desktop counterparts. Algorithms requiring extensive

computational resources should therefore be avoided.

4.1 Analysis Parameters

In order to find out the performance and viability of the proposed gesture recognition system

following testing and analysis parameters could be considered

a) Robustness: In the real-world, visual information could be very rich, noisy, and incomplete,

due to changing illumination, clutter and dynamic backgrounds, occlusion, etc. Vision-based

systems should be user independent and robust against all these factors.

b) Scalability: The Vision-based interaction system should be easily adapted to different scales of

applications. For e.g. the core of Vision-based interaction should be the same for desktop

environments, Sign Language Recognition, robot navigation and also for VE.

c) Computational Efficiency: Generally, Vision based interaction often requires real-time systems.

The vision and learning techniques/algorithms used in Vision-based interaction should be

effective as well as cost efficient.

d) User’s Tolerance: The malfunctions or mistakes of Vision-based interaction should be

tolerated. When a mistake is made, it should not incur much loss. Users can be asked to repeat

some actions, instead of letting the computer make more wrong decisions.

The computer vision techniques used in the application for manipulation of objects in virtual

environment have been implemented in C++ with the use of Open CV Library. The virtual

objects (front end) have been designed using OpenGLlibrary. The hardware requirements of the

application to be implemented include computer with1.99 GHz processor. The web cam used in

the experimental setup captures image sequences at the resolution of 320x240. Practical

experiments show that our application is implemented well in environments with little noises

(i.e., existence of objects whose color is similar to human skin) and with the balanced lightning

condition.

International Journal of UbiComp (IJU), Vol.3, No.1, January 2012

27

First, the user places his hand in front of the webcam. The webcam then detects the user’s hand

by creating a rectangle around it as shown in figure 5.

Figure 5. Hand Detected Once the hand has been detected the application further tracks different gestures of the user

performed by his hand and generates contour around it.

i. Move Left ii. Move Right

iii. Move Up iv. Move Down

Figure 6. Gestures for manipulating objects in virtual environment.

The application uses seven hand gestures defined within the application for manipulation with

objects in virtual environment. Figure 6 shows the different gestures along with their assigned

commands (functions) to manipulate the objects in virtual environment.

6. CONCLUSION

In present environment a number of facilities and various modes for providing input to any application are available. It is though unfortunate that with the ever increasing smart environments and corresponding input technologies still not many applications are available which are controlled using current and smart facility of providing input which is by hand gesture. The most important advantage of the usage of hand gesture based input modes is that using this method the user can interact with the application from a distance without using the keyboard or mouse. The application of manipulating objects through hand gestures in virtual environment is being proposed and implemented in the present paper provides a suitable efficient and user friendly human computer interface. With the help of this application the user can interact with the virtual objects using hand gesture instead of any other physical input devices .As the application provides the flexibility to the users and specifically physically challenged users to define the gesture according to their feasibility and ease of use.

The idea is to make computers understand human language and develop a user friendly human

computer interfaces (HCI). Making a computer understand speech, facial expressions and human

gestures are some steps towards it. Gestures are the non-verbally exchanged information. A

person can perform innumerable gestures at a time. Since human gestures are perceived through

vision, it is a subject of great interest for computer vision researchers. The project aims to

determine human gestures by creating an HCI. Coding of these gestures into machine language

demands a complex programming algorithm. An overview of gesture recognition system is

given to gain knowledge.

**Reference**

[1] S. S. Rautaray and A. Agrawal, Vision Based Hand Gesture Recognition for Human Computer Interaction: A survey, Springer Transaction on Artificial Intelligence Review, pp. 1–54, (2012).

[2] Ram Pratap Sharma and Gyanendra K. Verma, “Human Computer Interaction using Hand gesture “Science direct, ELSEVIER,Prcedia Computer Science 54 (2015) 721-727

[3] B. A. Myers, A Brief History of Human Computer Interaction, 1998

[4] Shigeo Abe, “Support Vector Machines for Pattern Classification, second edition”, Kobe University, Graduate School of Engineering, 2nd edition, Springer- Verlag London Limited 2005, 2010

[x1] Lalit Gupta and Suwei Ma “Gesture-Based Interaction and Communication: Automated Classification of Hand Gesture Contours”, IEEE transactions on systems, man, and cybernetics—part c: applications and reviews, vol. 31, no. 1, February 2001

[x2] Hall, E.T.: The Silent Language. Anchor Books. ISBN: 13 978-0385055499 (1973)

[x3] McNeill, D.: Hand and Mind: What Gestures Reveal About Thought. University of Chicago Press. ISBN: 9780226561325, 1992 (1992)

[5] V. I. Pavlovic, R. Sharma, and T. S. Huang. Visual interpretation of hand gestures for human-computer interaction: A review. IEEE Trans. Pattern Anal. Mach. Intell., 19:677{695, 1997.

[6] M. Cote, P. Payeur, and G. Comeau. Comparative study of adaptive segmentation techniques for gesture analysis in unconstrained environments. In IEEE Int. Workshop on Imagining Systems and Techniques, pages 28{33, 2006.

[7] A. A. Argyros and M. I. A. Lourakis. Real-time tracking of multiple skin-colored objects with a possibly moving camera. In Proc. European Conference on Computer Vision, pages 368{379, Prague, Chech Republic, May 2004.

[8] D. Chai and K. Ngan. Locating the facial region of a head and shoulders color image. In IEEE Int. Conference on Automatic Face and Gesture Recognition, pages 124{129, Piscataway, NJ, 1998.

[9] D. Saxe and R. Foulds. Toward robust skin identification in video images. In IEEE Int. Conf. on Automatic Face and Gesture Recognition, pages 379{384, 1996, 1996.

[10] G. Bradski. Real time face and object tracking as a component of a perceptual user interface. In IEEE Workshop on Applications of Computer Vision, pages 214{219, 1998.

[11] M. Krueger. Environmental technology: Making the real world virtual. Communications of the ACM, 36:36{37, 1993.

[12] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. IEEE Trans. Pattern Analysis and Machine Intelligence, 24(4):509{522, 2002.

[13] C. Maggioni. Gesture Computer - new ways of operating a computer. In Int. Workshop on Automatic Face and Gesture Recognition, pages 166{171, Zurich, Switzerland, 1995.

[14] J. Davis and M. Shah, Visual gesture recognition. Vision, Image, and Signal Processing, 141(2):101{106, 1994.

[15] Y. Cui, D. Swets, and J. Weng. Learning-based hand sign recognition using shoslf-m. In Int. Workshop on Automatic Face and Gesture Recognition, pages 201{206, Zurich, 1995.

[16] Y. Cui and J. Weng. Hand sign recognition from intensity image sequences with complex background. In Proc. IEEE Computer Vision and Pattern Recognition (CVPR), pages 88{93, 1996.

[17] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1):119{139, 1997.

[18] P. Viola and M. Jones. Robust real-time object detection. In IEEE Workshop on Statistical and Computational Theories of Vision, Vancouver, Canada, 2001.

[19] E. Ong and R. Bowden. A boosted classi¯er tree for hand shape detection. In Automatic Face and Gesture Recognition, pages 889{894, 2004.

[20] J. Rehg and T. Kanade. Model-based tracking of self-occluding articulated objects. In Proc. International Conference on Computer Vision (ICCV), pages 612{617, 1995.

[21] J. Lin, Y. Wu, and T. S. Huang. Capturing human hand motion in image sequences. In Proc. IEEE workshop on Motion and Video Computing, pages 99{104, 2002.

[22] J. Rehg and T. Kanade. Digiteyes: Vision-based hand tracking for human-computer interaction. In Workshop on Motion of Non-Rigid and Articulated Bodies, pages 16 {24, Austin Texas, November 1994.

[23] J. Lee and T. L. Kunii. Model-based analysis of hand posture. IEEE Computer Graphics and Applications, 15(5):77{86, 1995}.

[24] F. Quek. Eyes in the interface. Image and Vision Computing, 13(6):511{525, 1995.

[25] J. Martin, V. Devin, and J. Crowley. Active hand tracking. In IEEE Conference on Automatic Face and Gesture Recognition, pages 573{578, Nara, Japan, 1998.

[26] Q. Yuan, S. Sclaroff, and V. Athitsos. Automatic 2D hand tracking in video sequences. In IEEE Workshop on Applications of Computer Vision, pages 250 {256, 1995}.

[27] R. O'Hagan and A. Zelinsky. Finger Track - a robust and realtime gesture interface. In Australian Joint Conference on Arti¯cial Intelligence, pages 475{484, Perth, Australia, November 1997.

[28] R. E. Kalman. A new approach to linear °itering and prediction problems. Transactions of the ASME{Journal of Basic Engineering, 82:35{42, 1960

[29] A. A. Argyros and M. I. A. Lourakis. Real-time tracking of multiple skin-colored objects with a possibly moving camera. In Proc. European Conference on Computer Vision, pages 368{379, Prague, Chech Republic, May 2004.

[30] W. Kim and J. Lee. Visual tracking using snake for object's discrete motion. In IEEE Int. Conf. on Robotics and Automation, volume 3, pages 2608{2613, Seoul, Korea,, 2001.

[31] J. MacCormick and A. Blake. A probabilistic exclusion principle for tracking multiple objects. In Proc. International Conference on Computer Vision (ICCV), pages 572{578, Corfu, Greece, 1999.

[32] M. Isard and A. Blake. Icondensation: unifying low-level and high-level tracking in a stochastic framework. In Proc. European Conference on Computer Vision, pages 893 908, Berlin, Germany, 1998

[g1] Peter A. Andersen, Nonverbal Communication: Forms and Functions (Mountain View, CA: Mayfield, 1999), 36

[g2] Allan Pease and Barbara Pease, The Definitive Book of Body Language (New York, NY: Bantam, 2004), 121

**Contact based gesture**

[1] Henrik Birk and Thomas Baltzer Moeslund, “Recognizing Gestures From the Hand Alphabet Using Principal

Component Analysis”, Master’s Thesis, Laboratory of Image Analysis, Aalborg University, Denmark, 1996.

[2] Andrew Wilson and Aaron Bobick, “Learning visual behavior for gesture analysis,” In Proceedings of the

IEEE Symposium on Computer Vision, Coral Gables, Florida, pp. 19-21, November 1995.

[3] Thad Starner and Alex Pentland, “Real-time American sign language recognition from video using hidden

markov models”,Technical Report No. 375, M.I.T Media Laboratory Perceptual Computing Section, 1995.

[4] Jennifer Schlenzig, Edward Hunter, and Ramesh Jain, “Recursive spatio-temporal analysis: Understanding

Gestures”, Technical report, Visual Computing Laboratory, University of San Diego, California, 1995.