Hand Gesture Recognition based on Shape Parameters

Meenakshi Panwar

Centre for Development of Advanced Computing
Noida, Uttar Pradesh , India
cimpi.panwar@gmail.com

Abstract—**Pattern recognition and Gesture recognition are the growing fields of research. Being a significant part in non verbal communication hand gestures are playing vital role in our daily life. Hand Gesture recognition system provides us an innovative, natural, user friendly way of interaction with the computer which is more familiar to the human beings. Gesture Recognition has a wide area of application including human machine interaction, sign language, immersive game technology etc. By keeping in mind the similarities of human hand shape with four fingers and one thumb, this paper aims to present a real time system for hand gesture recognition on the basis of detection of some meaningful shape based features like orientation, centre of mass (centroid), status of fingers, thumb in terms of raised or folded fingers of hand and their respective location in image. The approach introduced in this paper is totally depending on the shape parameters of the hand gesture. It does not consider any other mean of hand gesture recognition like skin color, texture because these image based features are extremely variant to different light conditions and other influences. To implement this approach we have utilized a simple web cam which is working on 20 fps with 7 mega pixel intensity. On having the input sequence of images through web cam it uses some pre-processing steps for removal of background noise and employs K-means clustering for segmenting the hand object from rest of the background, so that only segmented significant cluster or hand object is to be processed in order to calculate shape based features. This simple shape based approach to hand gesture recognition can identify around 45 different gestures on the bases of 5 bit binary string resulted as the output of this algorithm. This proposed implemented algorithm has been tested over 450 images and it gives approximate recognition rate of 94%.**

Keywords—Image processing, hand gesture recognition, Pattern recognition, Human computer interaction, K-means clustering.

I. INTRODUCTION

A. Motivation

Gesture and Gesture recognition terms are heavily encountered in human computer interaction. Gestures are the motion of the body or physical action form by the user in order to convey some meaningful information. Gesture recognition is the process by which gesture made by the user is made known to the system. Through the use of computer vision or machine eye, there is great emphasis on using hand gesture as a substitute of new input modality in broad range applications. With the development and realization of virtual environment, current user-machine interaction tools and methods including mouse, joystick, keyboard and electronic pen are not sufficient. Hand gesture has the natural ability to represents ideas and actions very easily, thus using these different hand shapes, being identified by gesture recognition system and interpreted to generate corresponding event, has the potential to provide a more natural interface to the computer system. This type of natural interaction is the core of immersive virtual environments. If we ignore the world of computers for a while and consider interaction among human beings, we can simply realize that we are utilizing a wide range of gestures in our daily personal communication. By the fact it is also shown that people gesticulate more when they are talking on telephone and are not able to see each other as in face to face communication. The gestures vary greatly among cultures and context still are intimately used in communication. The significant use of gestures in our daily life as a mode of interaction motivates the use of gestural interface and employs them in wide range of application through computer vision.

B. Related Work

In some passed decades Gesture recognition becomes very influencing term. There were many gesture recognition techniques developed for tracking and recognizing various hand gestures. Each one of them has their pros and cons. The older one is wired technology, in which users need to tie up themselves with the help of wire in order to connect or interface with the computer system. In wired technology user can not freely move in the room as they connected with the computer system via wire and limited with the length of wire. Instrumented gloves also called electronics gloves or data gloves is the example of wired technology. These instrumented gloves made up of some sensors, provide the information related to hand location, finger position orientation etc through the use of sensors. These data gloves provide good results but they are extremely expensive to utilize in wide range of common application. Data gloves are then replaced by optical markers. These optical markers project Infra-Red light and reflect this light on screen to provide the information about the location of hand or tips of fingers wherever the markers are wear on hand, the corresponding portion will display on the screen. These systems also provide the good result but require very complex configuration. Later on some advanced techniques have been introduced like Image based techniques which requires processing of image features like texture, color etc. If we work with these features of the image for hand gesture recognition the result may vary and could be different as skin tones and texture changes very rapidly from person to person from one continent to other. And also under different illumination condition , color texture gets modified which leads to changes in observed results. For utilizing various hand gesture to promote real time application we choose vision based hand gesture recognition system that work on shape based features for hand gesture recognition. This is universal truth that every person poses almost same hand shape with one thumb and four fingers under normal condition. The success of approach discussed in paper [1] for hand gesture recognition based on shape features is highly influenced by some constraints like hand should be straight for orientation detection in image, if it will not be followed then result could be unexpected or wrong and also we fix the new parameter to detect the presence of thumb. In paper [2], the approach is based on calculation of three combined features of hand shape which are compactness, area and radial distance. Compactness is the ratio of squared perimeter to area of the shape. If compactness of two hand shapes are equal then they would be classified as same, in this way this approach limits the number of gesture pattern that can be classified using these three shape based descriptors and only 10 different patterns have been recognized[1]. The algorithm implemented in this paper is divided into four main steps. First one is image pre-processing and segmentation of hand in the image using k-means clustering. The second step includes orientation detection, which is done in order to categorize image sequence into vertical and horizontal class. In the third step it calculates some of the essential shape based features required for hand pattern detection and for generating the unique 5 bit sequence for 45 different hand shapes. Finally, these resulted bits are used for assigning different key press events to various hand gestures. This proposed approach is designed and implemented for working on single hand gesture with uniform background.

II. THE IMPLEMENTED ALGORITHM

The flowchart of the algorithm is shown in Fig. 1 and its main steps are discussed in the following.

A. Image Segmentation

Image Pre-processing is necessary for image enhancement and for getting good results. In this algorithm, the input sequence of RGB images gets converted in to YCbCr images as RGB color space is more sensitive to different light conditions so we need to encode the RGB information in to YCbCr. Image segmentation is typically performed to locate the hand object in image. The K-means clustering algorithm is an iterative technique which is used to segment the image in to K clusters. K-mean computes centroid of each cluster in order to minimizes the sum of distances from each object to its cluster. K-means iteratively minimizes the sum of distances from each object to its cluster centroid until the sum cannot be decreased further. The result of K-means clustering is a set of clusters that are well separated from other clusters and compacted in their own cluster. In this approach images is having uniform plain background consist of only one hand object, so we have two clusters, for representing the hand object and background of the image. Cluster '1' which represents hand, has all pixel values set as 1 and cluster '2' which represents background has 0 intensity pixels. To reduce the background noise we remove all small insignificant smudges or connected components from the image that has fewer than P pixel. And then apply filling of holes on binary image. After hand segmentation we need to calculate the boundary contours in order to locate the hand region in the image. It is performed by scanning the image from top to bottom and left to right, the first white pixel is encountered is set as left most point of hand. Then start scanning from right to left in top to bottom manner and the first white pixel thus found is set as right side of the hand. In this way we got the vertical bounds of hand in image .Within these vertical bounds we perform a horizontal scan from left to right and top to bottom. The first white pixel encountered is fixed as top-most point of the hand. The hand extends from the bottom-most part of the image, so no scanning is needed to locate the end of the hand.

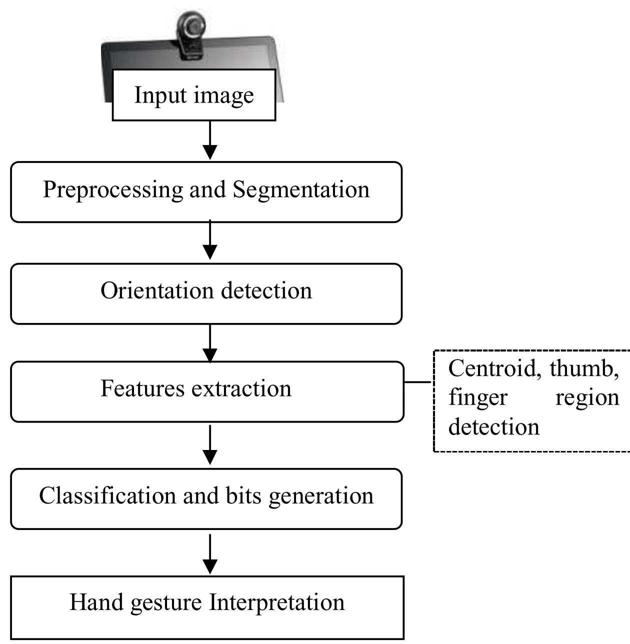


Figure 1: The flowchart of the implemented algorithm

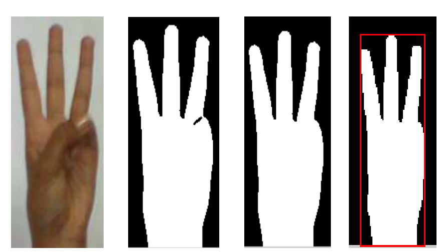


Figure 2: Input image, Cluster image, Enhanced image and localized hand object

B. *Orientation Detection*

For orientation detection of hand in image we employ dual approach because this step is very necessary to give accurate result as the success of this approach is heavily depend on this module. In this step we identify whether the hand is vertical or horizontal. Firstly we compute the ratio of length to width of bounding box with an assumption that if hand is vertical then length of the bounding box is greater than the width of bounding box and their ratio would be greater than 1. And if hand is horizontal then width of bounding box is greater than the length of bounding box and their ratio would be lesser than 1. Secondly we trace the boundary matrices or edges of hand in binary image. For horizontal hand whenever we get x-boundary is equal to 1 along with the increasing value of y-boundary for some time span we classify it as the horizontal hand and if we get y-boundary is equal to maximum of size of image with increasing value of x-boundary , it is set as vertical hand. To reduce the uncertainties discussed below and for proper orientation detection these two methods for orientation detection should give the same result. Uncertainties are, it may be possible that if any vertical image touch the left bottom corner of the image then it will show 'x=1' and 'y=max' which in turns gives an error or unexpected result or in rare condition it can also be possible that ratio of length to width for vertical image is less than 1. To avoid these kinds of errors we need to consider and compare the orientation results of both methods. In this way we categories the two categories of hand patterns, horizontal and vertical.

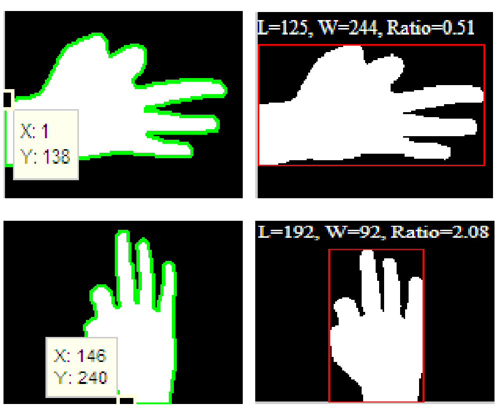


Figure 3: Horizontal image and Vertical image

C. *Features extraction*

1) Centroid: In this step, we calculate the centroid for partitioning the hand in to two halves, one which represents the finger portion and other which represents non finger region. Centroid is also called centre of mass and it divide the hand in to two halves at its geometric centre if the image is uniformely distributed.Centroid is calculated using image moment, which is the weighted average of pixel's intensities of the image. The centroid is calculated by first calculating the image moment using this formula [1]

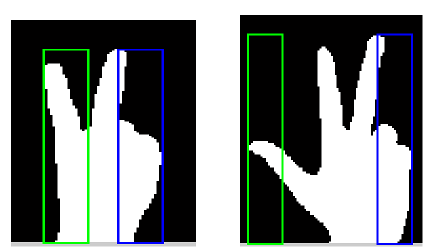
$$M_{ij}=\sum_{x}\sum_{y} x^i y^j I(x, y) \tag{1}$$

Where M_{ij} is image moment, I(x, y) is the intensity at coordinate (x, y).

$$\{\bar{x}\ \bar{y}\} = \{M_{10}/M_{00}, M_{01}/M_{00}\} \tag{2}$$

By using equation (2), we compute coordinates of centroid. $\bar{x}$, $\bar{y}$ are the coordinate of centroid and M_{00} is the area for binary image.

2) Thumb detection: Thumb detection step is performed in order to detect the presence or absence of thumb in hand gesture. Thumb is consider as a significant shape feature to classify various hand gestures in this approach. We know that thumb can either be reside at right most side of all finger of the hand or at left most side of the hand in general . To detect the presence of thumb in hand, we proceed with the previously calculated bounding box and consider the left side and right side of this bounding box. By taking 30 pixels width from each side of the bounding box we crop this bounding box in to two region, one which is represented by green boundary is left box and another is right box represented by blue boundaries in the image shown below. After having these two boxes we count the total number of white pixels presents in binary image which represent the hand object.Then we count number of white pixels present in each box i.e green and blue box. If there is less than 7% of total white pixels exist in any of the right box or left box, we consider that thumb is present in that box only. if both boxes having more than 7% percent of total white pixels in the image, then thumb is not present in any of the box . And If both boxes having less than 7% of total number of white pixels exist in image then thumb is not present in any of the box because thumb is only one and it can not be detected at both side of the bounding box for the same hand gesture. The percentage of white pixels set as 7% is choosen experimentaly on testing more than 400 images. In our previous approach discussed in paper[1], the parameter was taken as 0.69%, but for getting accurate result we set the percentage of white pixel for thumb detection as 7%. This method is applicable to both categories of hand.The results will be highly influenced by variation in orientation. Fig. 4 shows the partition of bounding box in two boxes represented by green and blue box. In Fig. 4(b), It detect thumb at the left hand side and in the green box , in which percentage of white pixels is counted less than 7 % of total white pixels.



(a) (b)

Figure 4: Thumb detection

3) Finger region detection: In this step we denote tip of the finger as peak. For getting the total number of finger raised in hand gesture we need to process only finger region of the hand that we have got in previous step by computing centroid. To proceed this task we trace the entire boundry matrices of hand. Vertical hand image and horizontal hand image have been processed in different manner. For vertical hand image, we only consider the y coordinates of the boundary matrices. When we get the values of y coordinates of boundaries starts increasing after the sharp decrement in the y-boundaries value. We consider this indication as tip of the finger and we fix it as a peak value or a peak. Similarly for horizontal image, we consider the x coordinate of the boundary matrices. This time only the x coordinates of the boundary matrices is traced. When we get the x coordinate of boundaries starts decreasing after the continous increment we mark this point as a tip of the finger in horizontal hand and set it as peak. In this way we found the tip of all raised and folded fingers in the image, but we need to classify significant peaks and insignificant peaks among them. For this we need to proceed to the next step to calculate the euclidean distance.

4) Euclidean distance: After marking the detected peaks or tip of the fingers in the hand we must find out the highest peak in the hand image. For this we calculate the distance between all tip of the fingers (detected peaks) and centroid using euclidean distance formula that is mentioned below

$$\text{E.D (a, b)} = \sqrt{(x_a - x_b)^2 + (y_a - y_b)^2} \qquad (3)$$

here 'a' represents all the boundary points and 'b' represent the reference point that is taken as centroid itself. On the basis of this distance formula we can find out the length of each raised or folded finger taking centroid as a reference point, this is done in order to extract the exact number of finger raised in the image. There may be some peaks detected which do not actually represent the tip of the raised fingers, but the tip of folded fingers. These peaks are considered as insignificant peaks as shown in Fig. 5. We can get rid of these kinds of insignificant peaks by computing the maximum peak. Putting the threshold at 75% of the maximum peak value, we can choose only those significant peaks whose values are more than this threshold value as these peaks represents the raised finger in hand gesture. Other peaks that are detected but do not intersect or fall above this threshold line would be treated as insignificant peak or folded fingers.

Insignificant peak

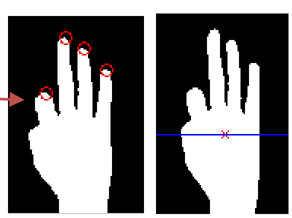


Figure 5: Detected peaks and Centroid of segmented hand

D. Classification and bits generation

Classification of various hand gestures is based on the features calculated in part II. The five bit binary sequence is thus generated to uniquely recognize and utilize these recognized hand gesture for supporting human computer interaction. Peak-Centroid plots are shown in Fig. 6. The significant peaks we identified in previous step is encoded as '1' and insignificant peaks is encoded as '0' based on the intersection status of various finger tips to threshold line. Leftmost bit in the 5 bit binary sequence is reserved for status of thumb in hand image. If thumb is present, leftmost bit will be 1 otherwise 0.

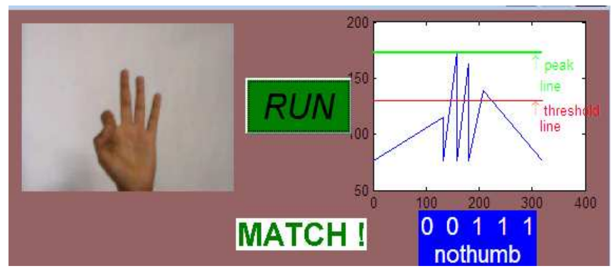
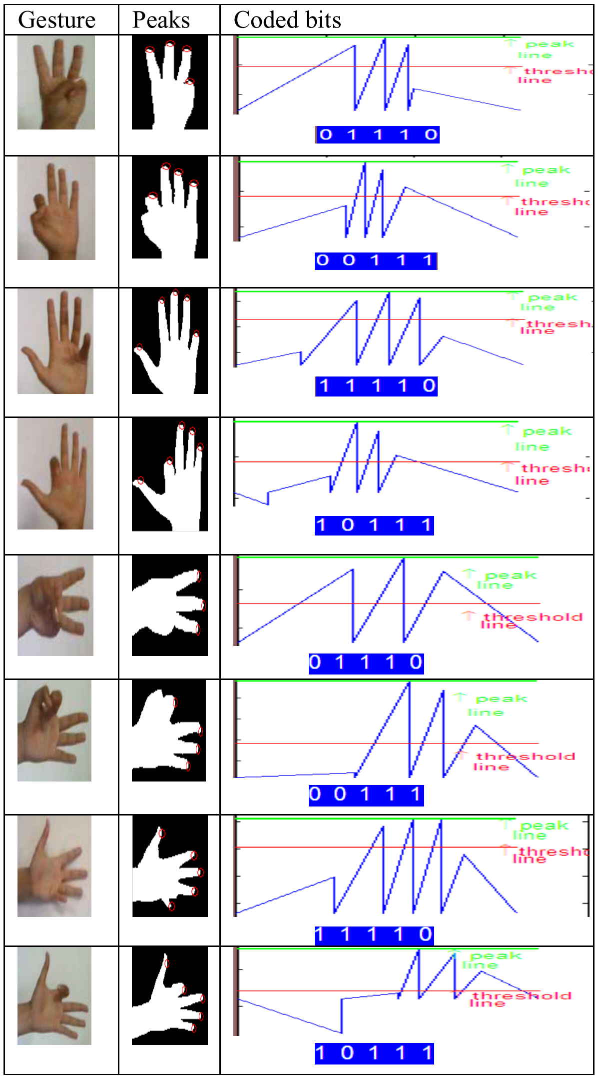


Figure 6: Hand gesture with bits code and distance plot

III. EXPERIMENTAL RESULTS

We have applied the above discussed algorithm and with this algorithm we have tested 450 images with 45 different patterns. By using these effective shapes based features and encoded bit sequence, we can recognize and classify 45 different hand gesture patterns. On the basis of generated binary bit sequences we can assign different-different task to support human computer interaction. Table I provide us with the experimental data result which shows some of the input gestures along with their peaks-centroid plot and resulted corresponding bits. Under the same category of orientation like vertical or horizontal these generated bit sequence will always be unique. However it may be possible that two hand images which are having same hand shape pattern but belong to different categories of orientation, have generate the same coded bits. This type of similarity is solved internally in the code on the basis of their orientation category so that these two same encoded bits which belong to different class of orientation can be assigned to perform different key press events. For example generated bit sequence of gesture 'j' and gesture '3' are same, but they are different in their orientation category as these hand gestures belongs to two different orientations, vertical and horizontal respectively. Table II shows the result of 450 images tested through this algorithm. Out of 450 images, it has correctly identified 423 images and falsely identified the remaining 27 cases, gives the success rate of 94% approximately with average computation time of 2 second for recognizing single image in image sequence. The algorithm is based on simple shape based feature calculation which provides us with the comfort of implementation.

TABLE I



Here, in table I, gestures and their corresponding finger tips(peaks) are plotted using centroid as a reference point for hand gesture recognition. If we divide the hand in to two regions using centroid, then it will partitioned the hand region in to the finger region and non-finger region, so for detecting the number of fingers in hand image we use this peak-centroid plot considering finger region as a significant portion of hand gesture. From other half of the hand which is non finger region, the only thing which matters is the presence or absence of thumb in the hand and for the detection of thumb we applied different approach on the basis of assumption. So we have only considered the finger region of the hand to plot the graph for detected peaks. Green line in the graph is plotted to show the highest finger tip or maximum distance of peak from the centroid and red line represents the threshold line which is plotted at the 75% of the maximum peak value or distance. This threshold line plays a significant role in classifying the detected peaks in to significant peaks and insignificant peaks., which is used to denote the raised finger and folded finger respectively. These detected peaks are then encoded in to binary bit sequence of 0 and 1 accordingly. Fig. 7 shows all the hand gesture with their corresponding key press events.

TABLE II
Hand Gesture Recognition Result

Gesture	Input Image	Successful Cases	Recognition Rate	Elapsed Time
1	10	10	100	1.72s
2	10	10	100	1.56s
3	10	10	100	1.74s
4	10	09	90	2.90s
5	10	10	100	1.57s
6	10	10	100	1.41s
7	10	10	100	2.45s
8	10	10	100	1.43s
9	10	10	100	1.65s
10	10	09	90	1.76s
11	10	10	100	2.56s
12	10	10	100	1.84s
13	10	10	100	2.41s
14	10	10	100	2.46s
15	10	09	90	1.50s
16	10	09	90	1.87s
17	10	10	100	1.72s
18	10	09	90	1.45s
19	10	10	100	2.74s
20	10	10	100	1.67s
21	10	09	90	1.53s
22	10	09	90	2.64s
23	10	10	100	1.82s
24	10	10	100	1.54s
25	10	10	100	1.70s
26	10	10	100	1.43s
27	10	10	100	1.63s
28	10	09	90	1.66s
29	10	10	100	1.79s
30	10	09	90	1.51s
31	10	10	100	2.46s
32	10	10	100	2.52s
33	10	10	100	2.64s
34	10	09	90	2.52s
35	10	10	100	2.54s
36	10	09	90	2.61s
37	10	09	90	1.45s
38	10	10	100	1.65s
39	10	10	100	1.98s
40	10	09	90	1.23s
41	10	09	90	1.45s
42	10	09	90	1.43s
43	10	10	100	1.62s
44	10	09	90	2.71s
45	10	09	90	1.54s
All	**450**	**423**	**94.0%**	**1.96s**

IV. HUMAN COMPUTER INTERACTION

This section introduces the application of this hand gesture recognition system. In our system, for making the interaction between gesture recognition software and MS Office/notepad, we have used the Abstract Windows Toolkit. It is used to control the mouse and keyboard remotely. For providing the human computer interaction through gesture recognition system, we have generated some of the key press events as shown in Table III .The sample hand gestures get converted in to corresponding key press events we have assigned.

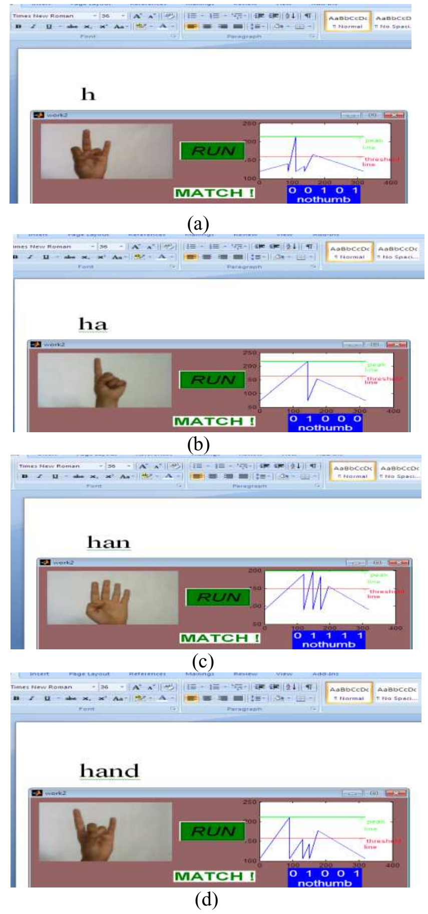
.

Figure 7: Sample hand gestures

V. CONCLUSION

We proposed a shape based approach for hand gesture recognition with several steps including smudges elimination orientation detection, thumb detection, finger counts etc. Visually Impaired people can make use of hand gestures for writing text on electronic document like MS Office, notepad etc. The strength of this approach includes its simplicity, ease of implementation, and it does not required any significant amount of training or post processing, it provide us with the higher recognition rate with minimum computation time. The weakness of this method is that we define certain parameters and threshold values experimentally since it does not follow any systematic approach for gesture recognition, and maximum parameters taken in this approach are based on assumption made after testing number of images. If we compare our approach with our previous approach described in paper [1].The success rate has been improved from 92.3% to 94% , the computation time decreased up to fraction of seconds. Also to make the system more robust, we have eliminated some of the constraints needed to be followed in our previous approach which makes it simpler. The proposed algorithm is simple and independent of user characteristics.

TABLE III
Comparison table for gestures and their corresponding Key Press events



REFERENCES

[1] Meenakshi Panwar and Pawan Singh Mehra , "Hand Gesture Recognition for Human Computer Interaction", in Proceedings of IEEE International Conference on Image Information Processing(ICIIP 2011), Waknaghat, India, November 2011.

[2] Amornched Jinda-apiraksa, Warong Pongstiensak, and Toshiaki Kondo, "A Simple Shape-Based Approach to Hand Gesture Recognition", in Proceedings of IEEE International Conference on Electrical Engineering/Electronics Computer Telecommunications and Information Technology (ECTI-CON), Pathumthani, Thailand , pages 851-855, May 2010

[3] A. Jinda-Apiraksa, W. Pongstiensak, and T. Kondo, "Shape-Based Finger Pattern Recognition using Compactness and Radial Distance," The 3rd International Conference on Embedded Systems and Intelligent Technology(ICESIT 2010), Chiang Mai, Thailand, February 2010.

[4] *Rajeshree Rokade , Dharmpal Doye, Manesh Kokare,* "Hand Gesture Recognition by Thinning Method", in Proceedings of IEEE International Conference on Digital Image Processing (ICDIP), *Nanded India, pages* 284 – 287, March 2009.