HAND GESTURE RECOGNITION USING DIFFERENT TYPES OF NEURAL NETWORK

Priyanka Pawar
Department of Electronics Engineering
J.D. College of Engineering & Management, Nagpur, India

ABSTRACT
The problems tackled by hearing and speech impaired people while interacting with normal people can be easily overcome by construction of a communication system which allows communicate impaired people to other without a middle interpreter. To overcome the communication gap few systems are available which uses Data gloves, sensors wiring etc. But these methods are bulky and expensive. The proposed system is a cost-effective and possible to minimize the distance between hearing and speech impaired people with normal people. Proposed system captures the hand signs and compare with existing database and then accordingly converted into text. This system is using an Image Processing algorithm which processes the detection and extraction of the input hand gesture from the image stream. In this system we are using functions like skin color based thresholding, contour detection for detection of hands and identification of important points on the hand respectively. The distance between these contour points from the centroid of the hand becomes our feature vector against which we will train our neural network.

Keywords- component; matlab tool; Hand Gesture Recognition, Neural Network.

I. INTRODUCTION
The difficulties faced by hearing and speech impaired people or physically challenged people in communicating with others. I am designing real time communication system enables differently impaired people or physically challenged people to communicate among themselves without an intermediate human translator. One of the major challenges in hand gesture recognition is to segment the hand region effectively in varying background and changing lighting condition [1][6]. The aim of this work is to evaluate different segmentation processes specific to hand gesture recognition. Recently, there has been a surge in interest in recognizing human hand gestures. Hand gesture recognition has various applications like computer games, machinery control (e.g. crane), and thorough mouse replacement [11]. One of the most structured sets of gestures belongs to sign language. In sign language, each gesture has an assigned meaning.

II. GESTURE RECOGNITION
Gestures are expressive, meaningful body motions – i.e. physical movements of the fingers, hands, arms, head, face, or body with the intent to convey information or interact with the environment. [12]
There are several aspects of a gesture that may be relevant and therefore may need to be represented explicitly. Hummels and Stappers (1998) [2] describe four aspects of a gesture which may be important to its meaning:

- Spatial information – where it occurs, locations a gesture refers to.
- Pathic information – the path that a gesture takes.
- Symbolic information – the sign that a gesture makes.
- Affective information – the emotional quality of a gesture.

In order to infer these aspects of gesture, human position, configuration, and movement must be sensed. Gesture recognition is the process by which gestures made by the user are made known to the system. [13]
Gesture recognition is also important for developing alternative human computer interaction modalities [14]. It enables human to interface with machine in a more natural way.
Gesture recognition is a technique which used to make computers ‘see’ and interpret intelligently is becoming increasingly popular. Dynamic gesture recognition isn’t something entirely new. [15] Gestures are expressive, meaningful body motions – i.e., physical movements of the fingers, hands, arms, head, face, or body with the intent to convey information or interact with the environment. [12]

III. ALGORITHMS FOR HAND GESTURE RECOGNITION
MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. MATLAB provides the Toolboxes that allow you to learn and apply specialized technology. The
recognition of hand gesture using MATLAB is described as follows. We use the algorithms for hand gesture recognition using MATLAB as Edge detection algorithms.

A. EDGE DETECTION
Following steps are used for detecting the edges:

• Image capturing using a webcam or the front camera of the mobile phone.
• Converting the captured image into frames.
• Image pre-processing using Histogram Equalization.
• Edge detection of the hand by using an algorithm like Canny Edge Detection.
• Enlargement of the edges of regions of foreground pixels by using Dilation to get a continuous edge.
• Filling of the object enclosed by the edge.
• Storing the boundary of the object in a linear array.
• Vectorization operation performed for every pixel on the boundary.
• Detection of the fingertips.
• Tracking of the fingertips in consecutive frames to determine the motion.
• Identification of the gesture based the motion.
• Insertion of the input stream into the normal input path of the computing device.

IV. DESCRIPTION OF ALGORITHMS
Working of hand gesture recognition process using MATLAB is discussed as follows;

Video Capturing Using a Camera
The image will be captured with the help of a single web camera, which will then lead to the image pre-processing stage. In case of mobile phones, it will be captured by the front camera of the mobile phones.

A. Frame Separation
The frames of the captured video are saved as images. A MATLAB function is used for this purpose.

B. Object Tracking
We use a local image co-ordinate scheme for determining the fingertips. The co-ordinate system is determined by the position of the fingertip at the start of the gesture. In simple words, the co-ordinate system is established in the first frame of a sequence of gestures, and then is kept constant for the subsequent frames. Thus, the need of having a common co-ordinate system for all images is eliminated. This sort of a system emulates a human eye i.e. the brain perceives any gesture irrespective of the background.

B. IMAGE PREPROCESSING
This block will basically concentrate on Histogram Equalization. In this stage we aim to increase the contrast among neighboring pixels, as shown in Fig. 3(b). The lowest colored pixel value is reduced to zero and the greatest colored pixel value is made to value 255. The other neighboring pixel values are averaged and spaced out in a similar manner. This helps us to locate our object of interest from the background. Figure 3 shows the Histogram Equalization Process. Before Histogram Equalization After Histogram Equalization.

C. EDGE DETECTION
After converting this image into grayscale image edge detection is applied. Here we find the points of the image where there are sharp edges or discontinuities or

www.ijaert.org
where sharp change in brightness is encountered. We will apply the Canny Edge Detection Algorithm for the purpose of detecting points at which image brightness changes sharply or formally, there are more discontinuities. Figure 4 shows the edge-detected hand.

D. IMAGE FILLING AND BOUNDARY DETECTION
From the hand contour obtained from the preprocessing steps, the feature of interest is the set of fingertips, which, in turn, is a subset of the boundary of the hand. We use bwboundaries, a MATLAB function to store the boundary of the hand contour in a linear array, formed sequentially from the topmost and leftmost boundary pixel, which is on. bwboundaries detects boundaries of filled images or holes within filled objects. Thus, we fill the continuous edge of the hand contour with white pixels as shown in Figure 3. Further, we detect boundaries of all objects in a cell array, each cell corresponding to the boundary of one object, and each element in every cell corresponding to a pixel on the boundary of that object. Since the hand should ideally correspond to the largest object in the image, we detect the largest cell array for use in vectorization. This eliminates any adverse effects noisy background might have on fingertip detection.

E. VECTORIZATION
In order to reduce computing complexity we define the angle $C(i)$ between two vectors $[p(i-k), p(i)]$ and $[p(i), p(i+k)]$ as curvature, where k is a constant. The points along the edge where the curvature reached a local extreme, that is the local features, are then identified. Some of these local features are labeled as “peak” or “valley”. We use this algorithm to compute curvatures at every point, and thus detect positions of the fingertip in the boundary detected hand contour as shown in Figure 6. It shows Yellow vectors denote curvatures belonging to a fingertip; green vectors denote curvatures, which do not belong to a fingertip.

IV. CONCLUSION
Hand gesture recognition based man-machine interface is being developed vigorously in recent years. Gesture recognition is also important for developing alternative human computer interaction modalities. It enables human to interface with machine in a more natural way. MATLAB provides the better solution for hand gesture recognition. The Edge Detection Algorithm for the purpose of detecting points at which image brightness changes sharply or formally. We used neural network algorithm for gesture identification. NN has a fast computational ability. Edge detection and NN provides good and powerful solution for hand gesture recognition using MATLAB.

REFERENCES


[10] Face and Hand Gesture Recognition Algorithm Based on Wavelet transforms and Principal Component Analysis, IEEE 2013
