

Live Hand Gesture Recognition using an Android Device

Mr. Yogesh B. Dongare

Department of Computer Engineering.

G.H.Raisoni College of Engineering and Management, Ahmednagar.

Email- yogesh.dongare05@gmail.com

Abstract—In the field of image processing it is very interesting to recognize the human gesture for general life applications. Gesture recognition is a growing field of research among various human computer interactions; hand gesture recognition is very popular for interacting between human and machines. It is nonverbal way of communication and this research area is full of innovative approaches. This paper aims at recognizing 40 basic hand gestures. The main features used are centroid in the hand, presence of thumb and number of peaks in the hand gesture. That is the algorithm is based on shape based features by keeping in mind that shape of human hand is same for all human beings except in some situations. The recognition approach used in this paper is artificial neural network among back propagation algorithm. This approach can be adapted to real time system very easily. In this paper for image acquisition android camera is used, after that frames are send to the server and edge detection of the video is done which is followed by thinning that reduce the noise, tokens are being created from thinning image after tokens are fetched. The paper briefly describes the schemes of capturing the image from android device, image detection, processing the image to recognize the gestures as well as few results. Keywords: Edge Detection, Sobel algorithm, Android, token detection, gesture recognition, neural network..

I. INTRODUCTION

Among the set of gestures intuitively performed by humans when communicating with each other, pointing gestures are especially interesting for communication and is perhaps the most intuitive interface for selection. They open up the possibility of intuitively indicating objects and locations, e.g., to make a robot change direction of its movement or to simply mark some object. This is particularly useful in combination with speech recognition as pointing gestures can be used to specify parameters of location in verbal statements. Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Gestures can originate from any bodily motion or state but commonly originate from the face or hand. Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans. It enables humans to communicate with the machine (HMI) and interact naturally without any mechanical devices. There has been always considered a challenge in the development of a natural interaction interface, where people interact with technology as they are used to interact with the real world. A hand free interface, based only on human gestures, where no devices are attached to the user, will naturally immerse the user from the real world to the virtual environment.

Android device brings the long-expected technology to interact with graphical interfaces to the masses. Android device captures the users movements without the need of a controller.

II. PROPOSED ALGORITHM

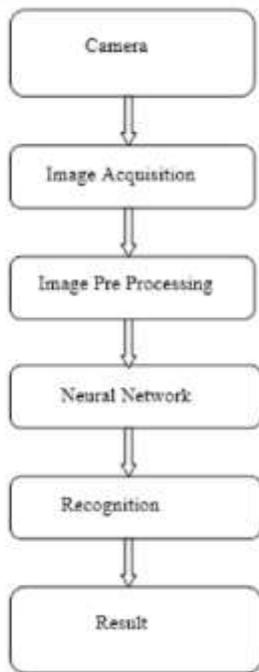


Fig. 1. Block Diagram of Hand Gesture Recognition Model

III. BACKGROUND

A. Hand Gesture Recognition

Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Gestures can originate from any bodily motion or state but commonly originate from the face or hand. Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans. Gesture recognition enables humans to communicate with the machine (HMI) and interact naturally without any mechanical devices. Gesture recognition can be conducted with techniques from computer vision and image processing. Gestures of the hand are read by an input sensing device such as an android device. It reads the movements of the human body and communicates with computer that uses these gestures as an input. These gestures are then interpreted using algorithm either based on statistical analysis or artificial intelligence techniques. The primary goal of gesture recognition research is to create a system which can identify specific human hand gestures and use them to convey information. By recognizing the hand symbols of a man it can help in communication with deaf and dumb people. It helps in taking prompt action at that time.

B. Edge Detection

Edge Detection [4] is the early processing stage in image processing and computer vision, aimed at detecting and characterizing discontinuities in the image domain. It aims at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. The points at which image brightness changes sharply are typically organized into a set of curved line segments termed edges. The same problem of finding discontinuities in 1D signals is known as step detection and the problem of finding signal discontinuities over time is known as change detection. Edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly in the areas of feature detection and feature extraction [1]. Some of the different types of edge detection techniques are:

1.Sobel Edge Detector

2.Canny Edge Detector

C. Sobel Edge Detector

The Sobel operator is used in image processing to detect edges of an image. The operator calculates the gradient of the image intensity at each point, giving the direction of the largest possible increase from light to dark and the rate of change in that direction. The result therefore shows how "abruptly" or "smoothly" the image changes at that point, and therefore how likely it is that, that part of the image represents an edge, as well as how that edge is likely to be oriented [7].

D. Artificial Neural Network

An artificial neuron is a computational model inspired in the natural neurons. These networks consist of inputs (like synapses), which are multiplied by weights (strength of the respective signals), and then computed by a mathematical function which determines the activation of the neuron. Another function (which may be the identity) computes the output of the artificial neuron (sometimes in dependence of a certain threshold). Artificial Neural Networks (ANN) combine artificial neurons in order to process information. The higher a weight of an artificial neuron is, the stronger the input which is multiplied by it will be. Depending on the weights, the computation of the neuron will be different. We can adjust the weights of the ANN in order to obtain the desired output from the network. This process of adjusting the weights is called learning or training.

The function of ANNs is to process information, they are used mainly in fields related with it. There are a wide variety of ANNs that are used to model real neural networks, and study behavior and control in animals and machines, but also there are ANNs which are used for engineering purposes, such as pattern recognition, forecasting, and data compression

[5].

E. Backpropagation Algorithm

The backpropagation algorithm [6] is used in layered feed-forward ANNs. This means that the artificial neurons are organized in layers, and send their signals forward, and then the errors are propagated backwards. The network receives inputs by neurons in the input layer, and the output of the network is given by the neurons on an output layer. There may be one or more intermediate hidden layers. The backpropagation algorithm uses supervised learning, which means that we provide the algorithm with examples of the inputs and outputs we want the network to compute, and then the error (difference between actual and expected results) is calculated. The idea of the backpropagation algorithm is to reduce this error, until the ANN learns the training data. The training begins with random weights, and the goal is to adjust them so that the error will be minimal [5].

IV. SYSTEM DESIGN

A. IMAGE ACQUISITION

Image acquisition is the first step in any vision system, only after this process you can go forward with the image processing. In this application it is done by using IPWebCam android application. The application uses the camera present in the phone for continuous image capturing and a simultaneous display on the screen. The image captured by the application is streamed over its Wi-Fi connection (or WLAN without internet as used here) for remote viewing. The program access the image by logging to the devices IP, which is then showed in the GUI.



Fig. 2. Original Image captured from Android device

B. IMAGE PRE-PROCESSING: EDGE DETECTION

In this program the edge detection technique used is sobel edge detector. The image captured is then passed through sobel filter.

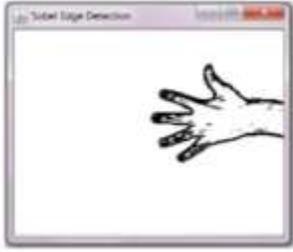


Fig. 3. Sobel Edge Filtered Image

C. THINNING

Thinning is a morphological operation that is used to remove selected foreground pixels from binary images, somewhat like erosion or opening. It can be used for several applications, but is particularly useful for skeletonization. In this mode it is commonly used to tidy up the output of edge detectors by reducing all lines to single pixel thickness. Thinning is normally only applied to binary images, and produces another binary image as output. After the edge detection, thinning has to be performed. Thinning is applied to reduce the width of an edge to single line.

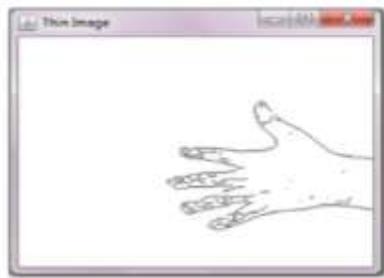


Fig. 4. Image after thinning

D. HAND TOKEN

The idea here is to make the image into a neuronal network usable form, so that the cosine and sine angles of the shape represents the criterias of a recognition pattern. Each square represents a point on the shape of the hand image from which a line to the next square is drawn.

On zooming a part of figure 5 it shows a right-angled triangle between the 2 consecutive squares, as shown in figure 6. This and the summary of all triangles of a hand image are the representation of the tokens of a hand from which we can start the neuronal network calculations.

The right-angled triangle in figure 5 represents a token of a single hand image. The angles A and B are the two necessary

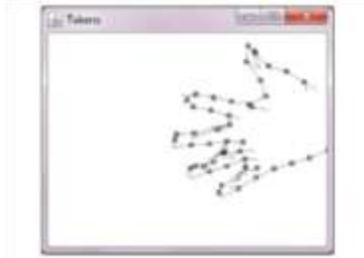


Fig. 5. Generated token of the original image

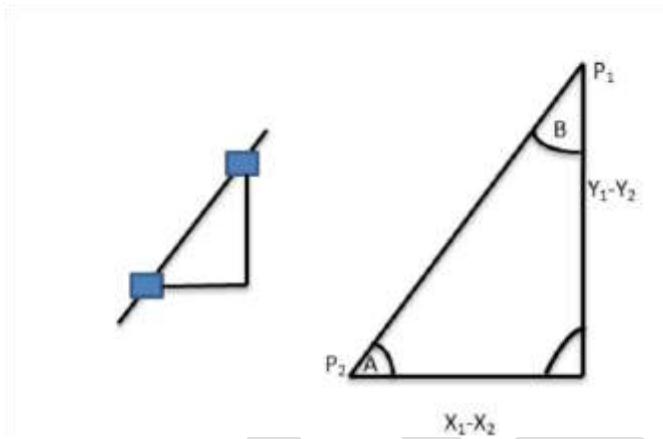


Fig. 6. Zoomed image of image tokens & effective right angled triangle

parts which will be fit into the neuronal network layers. With the two angles we can exactly represent the direction of the hypotenuse from point P1 to P2 which represents the direction of a hand image.

E. TRAINING DATA

Another main part of this work is the integration of a feed-forward backpropagation neural network. As described earlier the inputs for this neuronal network are the individual tokens of a hand image, and as a token normally consists of a cosine and sine angle, the amount of input layers for this network are the amount of tokens multiplied by two. The implemented network just has one input, hidden and output layer to simplify and speed-up the calculations on that java implementation. For training purpose the database of images located on the disk is used. It contains 6 different types of predefined gestures. These gestures are shown in figure 7. These are basic hand gestures indicating numbers zero to five. The implemented network just has one input, hidden and output layer to simplify and speed-up the calculations on that java implementation. For training purpose the database of images located on the disk is used. It contains 6 different types of predefined gestures. These gestures are first processed and then the tokens generated are passed to the network for training purpose. This process of training network from set images is done automatically when the application is initialized.

Orientation. The statistical summary of the results is as follows



Fig. 7. Database of Gestures

V. RECOGNITION

Recognition is the final step of the application. To fill the input neurons of the trained network, the previous calculated tokens discussed in section D are used. The number of output neurons is normally specified by the amount of different type of gestures, in this case it is fixed to 6. All other behavior of the network is specified by the normal mathematical principals of a backpropagation network as discussed in section E. It gives percentage of recognition to each gesture with highest percentage closely matching and lowest to the farthest matching and the closest match is considered as the result.

VI. TESTING

Figure 8 shows the screenshot of the screen when the application is started. Figure 9 shows the screen during the process of gesture recognition.

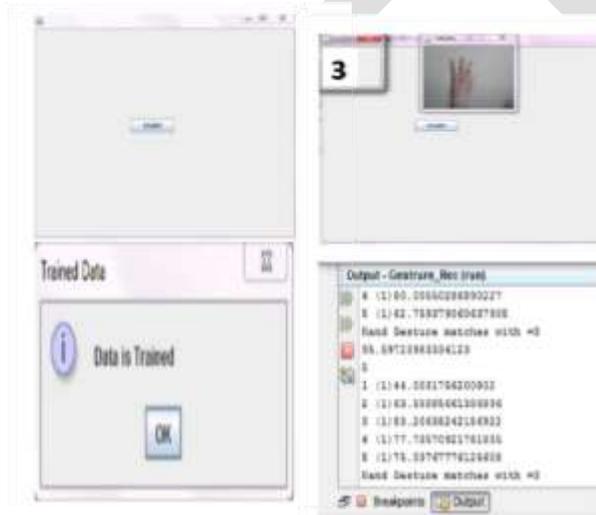


Fig. 8. Initialization of the pro

VII. RESULTS

To test the application, gestures are made by three different people. Some of the gestures are closed or have different

VIII. CONCLUSION

This system is useful for communicating with a deaf and dumb person. The system database has sign gestures of size of 176X144 pixels so that it takes less time and memory space during pattern recognition. The recognition rate of all gestures is between 70-80% which is an acceptable range. Overall accuracy of this system is 77% (approx).

In future, we can use a custom camera instead of the IpWebCam app which will further enhance the success rate of the system. Other different type of gestures can also be made part of the database. Try to eliminate noises from background to improve the accuracy rate.

REFERENCES;

- [1] Lindeberg, Tony (2001), "Edge detection", in Hazewinkel, Michiel, Encyclopedia of Mathematics, Springer, ISBN 978-1-55608-010-4.
- [2] H. Farid and E. P. Simoncelli, Differentiation of discrete multidimensional signals, IEEE Trans Image Processing, vol.13(4), pp. 496 – 508, Apr 2004.
- [3] D. Kroon, 2009, Short Paper University Twente, Numerical Optimization of Kernel Based Image Derivatives.
- [4] Lindeberg, Tony (2001), "Edge detection", in Hazewinkel, Michiel, Encyclopedia of Mathematics, Springer, ISBN 978-1-55608-010-4.
- [5] A. Bosch, A. Zisserman, and X. Munoz, Scene classification using a hybrid generative /discriminative approach, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 4, pp. 712-727, 2008 , 993-1007.
- [6] JagdishLalRaheja, Umesh Kumar, Human Facial Expression Detection Image Using Back Propagation Neural Network., International Journal of Computer Science and Information Technology (IJCSIT); Vol. 2, No. 1 , Feb 2010, pp. 116-112.
- [7] J. Matthews. An introduction to edge detection: The sobel edge detector, at www.generation5.org/content/2002/im01.asp, 2002