American Sign Language Fingerspelling Interpretation

Snehal Bende, Manasi Haldankar, Sweekritha Hegde
Guided by Prof. Meena Ugale

Abstract— The ASL (American Sign Language) Fingerspelling Interpreter is a system based on Image Processing Techniques where an image of hand gesture is captured and corresponding alphabet is displayed on the screen. A dataset is made which stores the hand gestures for each alphabet. The captured images are processed using MATLAB codes. Morphological operations are used to extract features of hand gestures. The features extracted are used to match the captured image with the image stored in the dataset and finally display the results.

Index Terms — Dataset, hand gestures, morphological operations, feature extraction, fingerspelling, Euclidean distance, classification.
.

1 Introduction

DEAF and dumb people communicate via sign language. Sign language is well structured language and each hand gesture has a meaning assigned to it. Each country has its own language. American Sign Language is a visual Language. It is used in United States of America. ASL has its own unique syntax and rules of grammar. This sign language is a combination of movements, orientation and shape of hand gesture.

It has been observed that it is difficult to communicate with deaf and dumb people with our normal language (spoken language). ASL fingerspelling translation into written English language has been an active area of research in computer vision for over a decade. To communicate with them one has to know the sign language. ASL Fingerspelling Interpreter, which acts as a translator, will help the common people understand at least basic gestures used in sign language and thus help in communication.

ASL Fingerspelling Interpretation is based on Image processing. A database will be made which will consist of training images and testing images. ASL Fingerspelling Interpreter will take RGB image as an input. This input image will be converted into gray scale image. Morphological operations such as thresholding, erosion, dilation, boundary extraction will be performed to get boundary of the image. Feature extraction is done to get boundary features. These features are used to classify images using KNN (K-nearest neighbor). ASL Fingerspelling Interpretation will be based on 24 alphabets as two alphabets j and z have motion which will add more complexity.

2 SYSTEM DESIGN

2.1 Android Client:

The first step is taking a picture. An android application will be set up which will act as the client. The client will take a picture and send it to the server.

2.2 PHP Script

The PHP script connects the Android application to the MATLAB via the HTTP protocol. The PHP script invokes the MATLAB code and the following image processing techniques will be carried out.

2.3 MATLAB Code

2.3.1 Thresholding

First the RGB input image will be converted into gray-level image using *rgb2gray*.

Thresholding is a type of image segmentation that converts grayscale images into binary images.

Thresholding of the image will be done using *imbinarize* whict uses Otsu's method [1].

2.3.2 Feature Extraction

In this step, morphological operations like dilation and erosion will be performed. For dilation *imdilate* will be used. For erosion *imerode* will be used. Boundary extraction will be performed by subtracting an eroded copy of the binary mask from its dilated copy. After boundary extraction the image will be cropped with the help of *boundingbox* using *regionprops.* These boundary features will be used to calculate area and centroid of the hand gesture.

2.4 Classification

Classification will be on the basis of Euclidean distance between features extracted from the training data set and testing data set.

- *Snehal Bende is currently pursuing bachelor's degree program in Information Technology engineering in Mumbai University, India, PH-9930550716. E-mail: snehabende@gmail.com*
- *Manasi Haldankar is currently pursuing bachelor's degree program in Information Technology engineering in Mumbai University, India, PH-9819054105. E-mail: manasihaldankar11@mail.com*
- *Sweekritha Hegde is currently pursuing bachelor's degree program in Information Technology engineering in Mumbai University, India, PH-9076963379. E-mail: sweekritha.hegde30@gmail.com*

IJSER

The results of classification will be sent to android device.

3 Equations

The formula for calculating the Euclidean distance:

$$dist((x, y),(a,b)) = \sqrt{(x-a) \wedge 2 + (y-b) \wedge 2} \qquad (1$$

where (x,y) and (a,b) are coordinates of two pixels.

4 FIGURES & TABLES

4.1 Figures

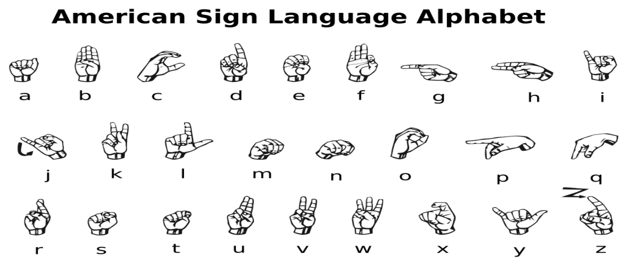

Fig 1 : American Sign Language alphabets

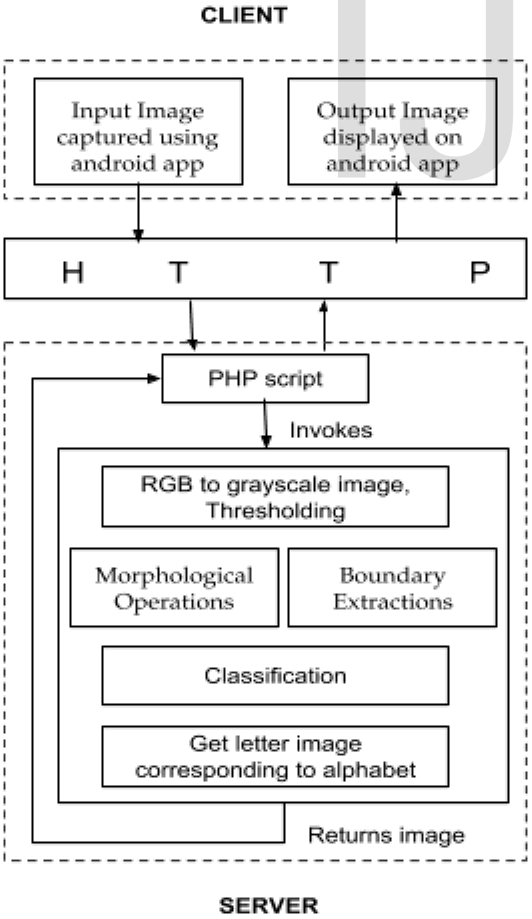
CLIENT
Input Image captured using android app
Output Image displayed on android app
H T T P
PHP script
Invokes
RGB to grayscale image, Thresholding
Morphological Operations
Boundary Extractions
Classification
Get letter image corresponding to alphabet
Returns image
SERVER

Fig 2: System Architecture

4.2 Algorithms

The algorithm used for classification is the KNN algorithm. The algorithm is as follows:

Let t be the number of training data samples. Let a be an unknown point.

- Store the training samples in an array of data points store[]. This means each element of this array represents a tuple (x, y).
- for j=0 to t:
- Calculate Euclidean distance d(store[j], a).
- Make set Q of L smallest distances obtained. Each of these distances correspond to an already classified data point.
- Return the majority label among Q. [10]

5 End Sections

5.1 Acknowledgement

We would like to thank Fr.Francis D'mello (Director of XIE) for providing us with such an environment so as to achieve goals of our project and supporting us constantly.We express our sincere gratitude to our Honorable Principal Mr. Y.D.Venkatesh for encouragement and facilities provided to us.We would like to place on record our deep sense of gratitude to Prof. Chhaya Narvekar, Head of Dept Of Information Technology, Xavier Institute of Engineering, Mahim, Mumbai, for her generous guidance help and useful suggestions.With deep sense of gratitude we acknowledge the guidance of our project guide Prof. Meena Ugale. The time-to-time assistance and encouragement by her has played an important role in the
development of our project.We would also like to thank our entire Information Technology staff who have willingly co-operated with us in resolving our queries and providing us all the required facilities on time.

5.2 Conclusion

ASL Fingerspelling Interpretation represents a solution that is useful for communicating with the dumb and deaf people. An Android application will be built which captures images of various gestures of hand. The captured images will be sent through PHP and processed using MATLAB. The corresponding alphabet will be displayed on the screen. Boundary extraction results are obtained by implementing morphological operations. This system is cost effective as only free softwares are used.

5.3 References

[1] H. M. Ewald, Ishan Patil and Shalini Ranmuthu, "ASL Fingerspelling Interpretation on Android", in Stanford University.

[2] N. Pugeault, R. Bowden, "Spelling It Out: RealTime ASL Fingerspelling Recognition" , IEEE International Conference on Computer Vision Workshops, IEEE, 2011.

IJSER

[3] N. Pugeault, ”ASL Finger Spelling Dataset,” in College of EMPS. [Online]. Available: http://empslocal.ex.ac.uk/people/staff/np331/index.php ?section=FingerSpellingDataset,[Accessed, Sept 20,2017].

[4] Barczak, A.L.C., Reyes, N.H., Abastillas, M., Piccio, A., Susnjak, T. (2011), “A new 2D static hand gesture color image dataset for ASL gestures”, Research Letters in the Information and Mathematical Sciences, 15, 12-20.

[5] “Tutorial on Client-Server Communications,” in Stanford University. [Online], http://web.stanford.edu/class/ee368/Android/Tutorial-3.pdf.

[6] Jayashree R. Pansare1, Shravan H. Gawande , Maya Ingle, “ Real-Time Static Hand Gesture Recognition for American Sign Language (ASL) in Complex Background”, https://uk.mathworks.com/matlabcentral/answers/uploaded_files/6161/Real-Time%20Static%20Hand%20Gesture%20Recognition%20for%20American%20Sign%20Language%20(ASL)%20in%20Complex%20Background.pdf

[7] K-Nearest Neighbours ,http://www.geeksforgeeks.org/k-nearest-neighbours/, [Accessed Sept 28, 2017].

IJSER