SIGN LANGUAGE RECOGNITION USING TEMPLATE MATCHING TECHNIQUE

Soma Shrenika
Bachelor Student, Computer Science and Engineering
Institute of Aeronautical Engineering,
Hyderabad, Telangana, India
somashrenika@gmail.com

Myneni Madhu Bala
Professor, Computer Science and Engineering
Institute of Aeronautical Engineering,
Hyderabad, Telangana, India
baladandamudi@gmail.com

***Abstract*—There is an absence of communication with deaf people in our society. To overcome this barrier the introduction of Sign Language (SL) took place. To convey meaning to normal people, sign language makes use of patterns that are visually transmitted sign patterns. Sign language is also useful for people suffering with Autism Spectrum Disorder (ASD). Normal people cannot understand the signs used by deaf, as they do not know the meaning of a particular sign. The system proposed here aims at solving this problem. This system uses a camera, which captures various gestures of the hand. Then, processing of the image takes place by using various algorithms. First, pre-processing of the image takes place. Then, determination of edges occurs by using an edge detection algorithm. Finally, a template-matching algorithm identifies the sign and display the text. As the output is text, one can easily interpret the meaning of a particular sign. This also curtails the difficulty to communicate with the deaf. The implementation of the system is by using OpenCV-Python. The system uses various libraries.**

Keywords— Sign Language, Hand Gesture, Edge Detection, American Sign Language, Indian Sign Language.

I. INTRODUCTION

In recent years, technologies like gesture recognition, facial recognition have received huge importance under the branch of sign language. Gestures are various movements used in the process of communication. Either hand or body makes gestures. Sign language makes use of gestures, which usually make use of visually transmitted patterns. All over the world, the people suffering from hearing problems have a count of 4,94,93,50,000 approximately. Some of the existing sign language conversion systems consider hand orientation, hand shape and hand movement. In sign language, every sign has a meaning assigned to it, so that it becomes easy to understand and interpret by the people. The people, based on their language and the place in which they live, develop discrete and non-identical sign languages. There is no sign language accepted universally. People use various sign languages across the world.

Indian Sign Language presents various hand movements by using both right and left hands. The proposed work focuses on hand shape and orientation and works on American Sign Language. When using ASL, only one hand is used. Therefore, it becomes easy for implementing the system. ASL does not depend on any of the spoken languages and it has its own path of development.

The process in brief includes acquiring images using a camera. Then performing pre-processing steps on the image, that is, convert the acquired image, which is in RGB model to gray scale image. Later, track the edges by using canny edge detection algorithm. Finally, detecting the pattern using template-matching algorithm, this outputs the result as text.

This system bridges the imbalances between deaf people and normal people without any requirement of an intermediate translator. It achieves the objective of conversion of gestures to text. Before going to the methodology, have a glance at literature survey.

II. LITERATURE SURVEY

(Kamal Preet Kour et. al.) Presented a novel design for sign language detection system with set of identified features and extraction methods of hand movements by using an algorithm named Speeded Up Robust Features (SURF) using image processing. [5]

(Zhi-hua Chen et. al.) Presented a framework in which acquiring the hand region from the background by using segmentation. Finally, a rule classifier is applied to predict the results. [6]

(Neelam K. Gilorkar et. al.) Proposed a vision based system for hand gesture recognition. In segmentation process, they converted rgb to gray scale. To overcome the probability of useless detections, they proposed this method. They put forth a technique, improvised Scale Invariant Feature Transform (SIFT), using MATLAB. [7]

(T. Ayshee et. al) Put forth an approach where, an intelligent system is developed which interprets Bengali sign language to spoken languages by using fuzzy rule based system. Here, angles are measured for identifying the rules. [8]

(B. Bauer et. al.) Describe an approach for continuous sign language recognition system. It is a system, which depends on videos of continuous hidden Markov models (HMM). It makes use of German sign language (GSL).The input to the system is feature vectors that reflect manual signs. [9]

III. DATA DESCRIPTION

The proposed system uses American Sign Language (ASL) data set to identify the sign made by a gesture. The data set has numbers labeled from zero to nine and alphabets from a to z. This data set has 70 samples for each of the 36 symbols. There are 70 samples for each symbol. These 70 samples of each symbol cover all the hand shapes and movements. The features are all of right hand. The entire data set is available in both compressed and uncompressed format. Every sample of the data set is, characterized with its equivalent sign. A unique sign letter corresponds to every sample.

A. American Sign Language(ASL)

It originated two centuries ago. It has many features that are similar to the languages people speak. It is different from the English that people use for communication daily. The semantics of ASL also deviates from that of English. ASL makes use of gestures of the hand. They also include facial gestures. ASL consists of 26 symbols, known as American

manual alphabet. They are in Fig 1. ASL contains vowels and consonants.

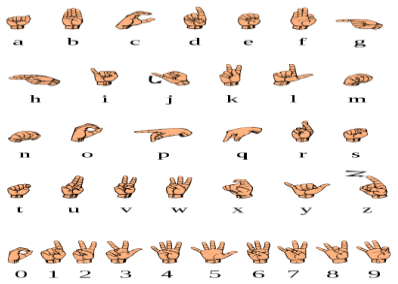

Fig. 1. American manual alphabets

IV. Proposed Methodology

The schematic diagram for sign language recognition is given below in Fig 2. It includes,

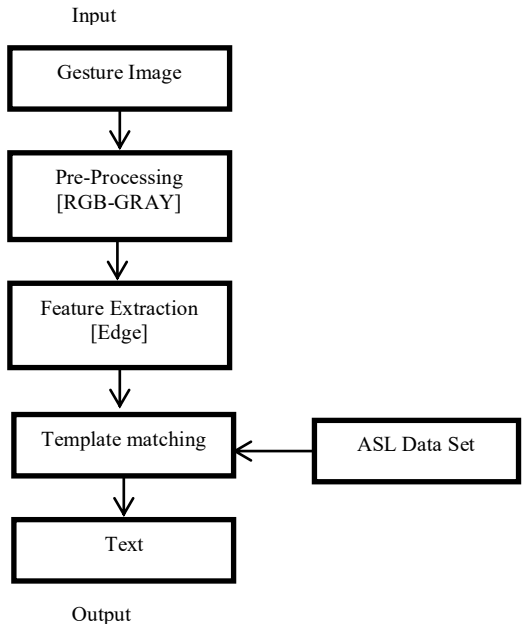

Fig. 2. Schematic diagram for sign language recognition

The schematic structure has the following steps:

A. Gesture Image

Acquire gestures of hand by using any type of camera. The captured image is in RGB color format.

1. RGB Color Format:

RGB color format is composed of three primitive colors namely: red(R), green (G), blue (B). Many image-processing techniques use RGB color model as a prerequisite. RGB is a device dependent model. Different systems produce different values. In order to get rid of distortions caused by light and shadows we can normalize the RGB values. The normalization process proceeds using the following three equations (1), (2), (3)

$$r = \frac{R}{R+G+B} \quad (1)$$

$$g = \frac{G}{R+G+B} \quad (2)$$

$$b = \frac{B}{R+G+B} \quad (3)$$

The sum of the normalized components of RGB is unity (r + g + b = 1) [10].

B. Pre-Processing

In this step, transformation of RGB image takes place, which is 3D to 1D. This transformation to one-dimensional image is to ease the process. The one-dimensional format is the gray scale image. There are various methods to convert an image from RGB color space to gray image. Here, Average method is used. Average method is the most basic and simple method that is used for converting image to gray scale image. It works as shown in (4),

$$\text{Gray} = \frac{R+G+B}{3} \quad (4)$$

This method takes 33% of RED, 33% of GREEN and 33% of BLUE i.e. the contribution is equal from all the three colors. This method does not turn out to be the best, but for this work, this method works well and satisfies the requirement.

C. Feature Extraction

In image analysis, determining the edges is important. Edge detection is useful to extract boundaries, corners, lines and curves. It removes the data that is not useful. According to [11], canny edge detection algorithm works better than many other edge detection algorithms. The figure below shows output of various edge detection algorithms.

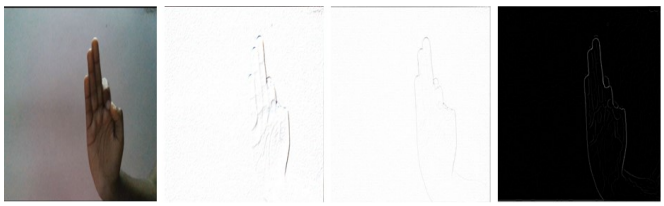

Fig.3. Fig.4. Fig .5. Fig.6.

Fig. 3. Input image (RGB format)

Fig. 4. Roberts cross edge detection

Fig. 5. Laplacian edge detection

Fig. 6. Canny edge detection

Canny Edge Detection:

Canny edge detection algorithm has multiple stages. Precisely, there are five stages to detect the edges using this algorithm.

- Image Smoothing

Every image will have some noise associated with it. Smoothing filters reduce image noise. Gaussian filters are smoothing filters that reduce this noise by using a Gaussian kernel. As an input, provide the values for height, width and standard deviation in both the directions. If only σ_x value is

given, the value of σ_y will be taken as σ_x. If they are zero, the kernel size is used for calculation of standard deviation.

$$\sigma = 0.3 * \left((\text{ksz} - 1) * 0.5 - 1\right) + 0.8 \quad (5)$$

Where, ksz is the kernel size and ksz%2 ==1 i.e. odd
The size of kernel is taken as (5, 5), which means height=5 and width=5. The value of σ is obtained from height and width.

- Gradient Magnitude

Calculate the gradient magnitude. It calculates the intensity values of the edges. Apply sobel filter, that uses a convolution mask of size 3x3 to detect the changes in intensities and gradient values in horizontal and vertical directions, i.e. (x, y). It returns the first order derivatives for both the directions. From this, magnitude and slope of the gradient are gathered. The sobel masks in both the directions are,

$$G_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * \text{image} \quad (6)$$

$$G_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * \text{image} \quad (7)$$

$$G = \sqrt{G_x^2 + G_y^2} \quad (8)$$

$$\theta = \tan^{-1}\frac{G_x}{G_y} \quad (9)$$

The directions are assigned one among any of the four directions, 0, 45, 90, 135, based on the following criteria,

If the value, $0 \leq \theta \leq 22.5$ or $157.5 \leq \theta \leq 202.5$ or $337.5 \leq \theta \leq 360$ then, θ=0

If the value, $22.5 \leq \theta \leq 67.5$ or $202.5 \leq \theta \leq 247.5$ then, θ=45

If the value, $67.5 \leq \theta \leq 122.5$ or $247.5 \leq \theta \leq 292.5$ then, θ=90

Otherwise, θ=135

- Non-maximum Suppression

This method reduces the thickness of the image and reduces the blurring effect. The edges determined in this process will be sharp. In this process, consider the neighboring pixels. If θ=0, comparison with its horizontal pixel is done. If the pixel value is less than the pixel value in its horizontal neighbors, suppress value to zero. If θ=45, comparison with positive direction diagonal pixels is done. If pixel value is less than the pixel value in its diagonal neighborhood, suppress its value to zero. If θ=90, comparison with vertical pixel is done. If pixel value is less than the pixel value in its vertical neighbors, suppress its value to zero. If value of a pixel is less than the values of negative diagonal pixels, suppress its value to zero.

- Double threshold

At this stage, only the pixels with maximum gradient values are useful. Define the values for high threshold and low threshold. The value of high threshold is 200.The value of low threshold is 100. Consider these values for comparison. Compare the gradient value of a pixel with previously defined threshold values. If its value is larger than high threshold value, the pixel value is made 255. These pixels are known as strong pixels. If its value is less than low threshold value, it will be reduced to 0. If gradient value is greater than low threshold, they are known as weak pixels.

- Hysteresis

Finally, detect the edges by using hysteresis process. Here, the weak pixels are converted to strong pixels. This process will consider the 8-neighbours of a pixel. If there is at least one strong pixel in the 8-neighbourhood of a weak pixel, it is converted to strong pixel.

D. Template Matching

After processing the image through various stages, the image obtained is, compared with the images in the data set. By using template-matching algorithm, identification of the pattern is possible. Here, the template image is the edge image obtained from edge tracking. The dimensions of the template image are mxn. The dimensions of the image in data set are mxn. This comparison of template image with the images present in the data set uses Sum of Absolute Difference (SAD) method. SAD method computes intensity values. This method computes the values by subtracting the pixel values of both the images and then adds the result, until no pixel is left. The image that produces least SAD value indicates the best match for a sign and its corresponding label is the output.

V. RESULTS

This section shows the results obtained after performing the experiment.

Fig. 7. Input image in RGB format

Fig 7 shows a sample input image of hand gesture. This image is a three-dimensional image.

Fig. 8. Gray scale image

Fig 8 shows the output after using average method on RGB image. This image is a one-dimensional image.

```
C:\Users\Shrenika\Desktop\hh1>python gry.py
[161 161 147]
157
```

Fig. 9. Sample pixel value after using average method

Fig 9 shows a sample pixel value at location [100,100]. First line shows RGB values. In the next line the value is in gray scale.

Fig. 10. Image after applying Gaussian filter for smoothing.

Fig 10 displays the image after applying Gaussian filter for smoothing. This filter removes image noise.

Fig. 11. Gradient magnitude image

Fig 11 image is a gradient magnitude image where intensity values are calculated by using filter.

Fig. 12. Image after non maximum suppression

Fig 12 is an image after non-maximum suppression process. The thickness of the image is reduced to obtain sharp edges. The values are suppressed based on angles.

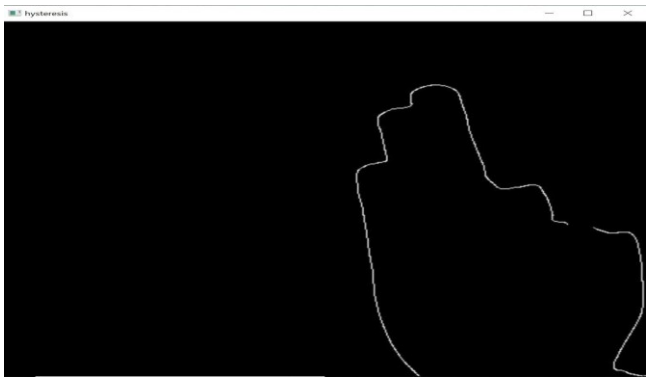

Fig. 13. Hysteresis image

Fig 13 is an image after hysteresis process. This is the final image obtained in the last step of canny edge detection.

Fig. 14. Output image with label

Fig 14 shows the label obtained after template matching using SAD and the matched image from the data set.

VI. Conclusion And Scope

The aim of this paper is to help and serve the deaf of our society to communicate with normal people. Here the implementation of the system is using image-processing techniques. This system is for people who cannot use gloves, sensors and other highly refined equipment. First, acquire image with a camera. Then convert it to gray scale image for further processing. Edge detection algorithm was used to detect the sign in the image. There, the process includes removal of noise and other less important data and applying smoothing algorithm to image, finding gradient magnitude followed by tracking the edges by hysteresis. The last step is displaying the sign alphabet. In future, we can develop a system that is two-way system where, conversion of sign to text and text to sign is possible. Developing a system, where interpretation involves dynamic gestures. Implementation can extend to mobile phones.

Acknowledgment

This research contribution is a part of undergraduate research and content development at Institute of Aeronautical Engineering.

REFERENCES

[1] Neha Poddar, Shrushti Rao, Shruti Sawant, Vrushali Somavanshi, Prof. Sumita Chandak "Study of Sign Language Translation using Gesture Recognition", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 2, 2015.

[2] Sawant Pramada, Deshpande Saylee, Nale Pranita, Nerkar Samiksha, Mrs.Archana S, Vaidya "Intelligent Sign Language Recognition Using Image Processing", IOSR Journal of Engineering, Vol. 3, Issue 2, 2013.

[3] Prof. Supriya Pawar, Sainath Muluk, Sourabh Koli "Real Time Sign Language Recognition using Python", International Journal of Innovative Research in Computer and Communication Engineering ,Vol. 6, Issue 3, 2018.

[4] P.V.V.Kishore, "Segment, Track, Extract, Recognize and Convert Sign Language Videos to Voice/Text", International Journal of Advanced Computer Science and Applications, Vol. 3, Issue 6, 2012.

[5] Kamal Preet Kour, Dr. Lini Mathew" Sign Language Recognition Using Image Processing", International Journals of Advanced Research in Computer Science and Software Engineering, Vol. 7, Issue 8, 2017.

[6] Zhi-hua Chen, Jung-Tae Kim, Jianning Liang, Jing Zhang and Yu-Bo Yuan "Real-Time Hand Gesture Recognition Using Finger Segmentation", The Scientific World Journal, Volume 2014, Article ID 267872, 9

[7] Neelam K. Gilorkar, Manisha M. Ingle, "Real Time Detection And Recognition Of Indian And American Sign Language Using Sift", International Journal of Electronics and Communication Engineering & Technology, Vol. 5, Issue 5, 2014.

[8] T. Ayshee, S. Raka, Q. Hasib, Md. Hossian, R. Rahman, "Fuzzy Rule-Based Hand Gesture Recognition for Bangali Characters", in IEEE International Advanced Computing Conference, 2014.

[9] B. Bauer,H. Hienz "Relevant features for video-based continuous sign language recognition", IEEE International Conference on Automatic Face and Gesture Recognition, 2002.

[10] Patil, Prajakta M., and Y. M. Patil, "Robust Skin Colour Detection and Tracking Algorithm", International Journal of Engineering Research and Technology. Vol.1, Issue8, 2012.

[11] Mamta Juneja , Parvinder Singh Sandhu," Performance Evaluation of Edge Detection Techniques for Images in Spatial Domain", International Journal of Computer Theory and Engineering, Vol. 1, Issue 5, 2009.

[12] Dataset- kaggle.com/ayuraj/asl-dataset

[13] ASL image- Ds13 at English Wikipedia [Public domain]. upload.wikimedia.org/wikipedia/commons/c/c8/Asl_alphabet_gallaudet.svg