Real Time Sign Language Gesture Translation Using Deep Learning

Abdulmateen Pitodiya[1], Mishty Singha[2], Srishti Shukla[3], Vikas Gupta[4], Sonal Dubal[5]

Department of Electronics and Telecommunication,
Vidyavardhini's College of Engineering and Technology, University of Mumbai, Vasai, India

[1]abdulpitodiya9@gmail.com, [2]miscap8820@gmail.com, [3]srishti.shukla99@gmail.com, [4]vikas.gupta@vcet.edu.in, [5]sonaldubal468@gmail.com

ABSTRACT:
Deaf and speech impaired persons utilise sign language as their major form of communication, however sign language is difficult for most others in society to understand. As a result, these people confront several problems every day. There are several models in both hardware and software; the former is expensive and difficult to use continuously, while some of the latter are already in use but have limits like low forecast accuracy, background condition restrictions, and many more. Our approach suggests a real-time gesture translation system that makes use of deep learning and image processing methods. With only a camera needed, the objective is to make it possible for signers and non-signers to converse without any issues. In order to anticipate a gesture with accuracy close to 99%, a massive dataset of 13000 photos was employed. The technique includes employing a camera to collect sign motions, picture pre-processing, and machine learning to translate the acquired gesture into text message and spoken form.

Keywords: Gesture Translation, VGG16, CNN, Image Processing, Deep learning, Text to Speech.

1 Introduction

When spoken language is difficult to understand, sign language is a way of communicating only via body language, namely by using the hands and arms. Around 466 million deaf or hard of hearing persons use sign language, including facial expressions, stances, and gestures, to communicate [1]. Deaf (hearing impaired) persons, hard-of-hearing people, people with different speech impairments, and deaf people all use sign language as their primary language. But regrettably, due to illiteracy in sign language comprehension, only few individuals are able to grasp it [2–5]. Because of this, it is extremely difficult for someone who knows sign language to communicate with someone who does not [6].

Speech interpreters are essential to speech impaired people's participation in medical, legal, educational, and training activities. For instance, if a patient visits a clinic and finds it difficult to explain their illness, this communication gap may make it more difficult for the doctor to identify the patient's condition and prescribe the right treatment. One popular remedy for the issue is to hire an interpreter who is fluent in sign language to help the disabled learn, although this is not always possible. The other options are sign language gesture translation software [11–15] and hardware [7–10].

This work introduces a real-time gesture translation system based on deep learning and image processing that just needs a webcam to enable signers to interact with non-signers. To obtain accuracy close to 99%, a massive data collection of 13000 photos was employed. The technology includes recording sign motions using a camera, analysing obtained images, and machine learning to translate collected gestures into text and voice. The structure of the paper is as follows. Section II addresses related work, Section III talks implementation and the proposed technique, Section IV shows the findings, and Section V wraps up the study.

2 Literature Survey

A glove's transmitter and receiver system [7] employs a flex sensor to collect data from hand gestures, which the Arduino Uno then processes to produce text and audio. A glove [8] that combines a microprocessor, accelerometer, and flex sensor to create a low-power, lightweight sign language translator. The sign gesture in [9] is sensed by an LED-LDR pair, which then passes the analogue signal it generates to the MSP430G2553 microcontroller, which further converts it into a digital sample signal. Wireless ASCII letter transmission is made possible by a zigbee module; [10] makes use of a leap motion controller. While [11–12] uses MATLAB to extract the features from the collected pictures and Support Vector Machine (SVM) as an image classifier. [13] asserts that using gear like a glove or a Microsoft Kinet Sensor is unnecessary since the camera processes its frames as it records sign language. The methods include colour segmentation, object stabilisation, and picture detection (sign language). To identify hand gestures, the grid-based feature extraction method is utilised, followed by the k-Nearest Neighbor (KNN) algorithm. To get precise skin detection results, the picture is detected using the HSV parameters in [14]. The method makes use of an algorithm for detecting skin tone. Convolutional neural networks are used in [15] to implement gesture detection with better accuracy and faster response times. There are several models available in both software and hardware formats. Hardware-based solutions can be expensive and inconvenient for people to use. Additionally, some software systems have drawbacks such poor prediction accuracy, constraints on backdrop conditions, and a lack of functionality like text-to-speech conversion.

3 Implementation and Methodology

3.1 Dataset Research

At first, a custom-labeled dataset was created with about 500 photos for each alphabet. Using Python's open CV package, 128 x 128 pixels of resolution photos were taken. Thus, a total of 26 alphabets were recorded, as shown in fig. 1, with about 500 photos being gathered for each alphabet. 13000 images in total will be used to train the model. Additionally, the dataset underwent pre-processing to make it more realistic.

3.2 Proposed Methodology

Sign language gestures are photographed using an Android smartphone or any camera, and their frames are sent for pre-processing. Fig. 2 displays a block schematic of the suggested technique. To eliminate any noise present, the image was pre-processed using a Gaussian and Median blur filter. Following the use of adaptive thresholding, a region of interest based on the skin's pixel value is produced. Pre-trained neural network VGG-16 is utilised to recognise gestures and outputs text messages. Python text to voice converter is used to convert text to speech (pyttsx3).

Data Pre-Processing: It is crucial that the webcam's raw photos are processed beforehand to reduce noise and exclude the area of interest. In order to do this, a series of operations are applied to the image, improving the picture's quality while also somewhat strengthening the system's resistance to external lightning conditions. First, median blur and gaussian blur are used to eliminate any noise. The picture is also reduced in size to 128x128 pixels to make computations easier and to shorten the time it takes for gesture translation. The region of interest is then determined using adaptive thresholding depending on the skin's pixel value. The threshold is determined depending on the values of the adjacent pixels using adaptive thresholding. The results of thresholding are displayed in Figure 3. Based on the skin threshold value, the hand's borders are properly captured. In order to make the collected pictures more realistic, Data Augmentation is used to flip, shear, zoom, and tilt the images. The model is then trained on these changes to improve accuracy. Even in dim lighting, the use of adaptive thresholding makes it possible to record the gesture's palm area clearly.

Convolutional Neural Network (CNN) / VGG-16: CNN is one of the primary categories used in neural networks to accomplish image identification and classification. The CNN image classification system accepts a picture as input, processes it, and assigns it to one of several categories. It is organised into three dimensions—width, height, and depth—depending on the resolution. In terms of technology, CNN models for deep learning are used for both training and testing; each input image is passed through a series of convolutional layers with filters, pooling, fully connected layer (FC), and the SoftMax function to classify an object with probabilistic values between 0 and 1.

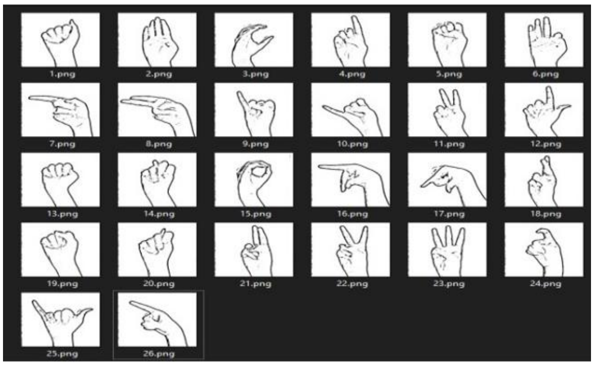


Fig. 1. Custom labelled dataset

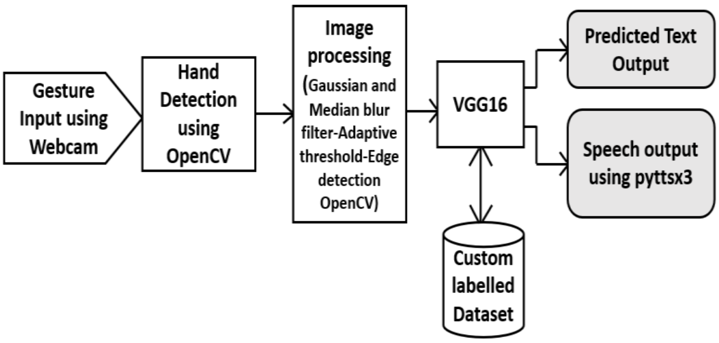

Gesture Input using Webcam
Hand Detection using OpenCV
Image processing (Gaussian and Median blur filter-Adaptive threshold-Edge detection OpenCV)
VGG16
Predicted Text Output
Speech output using pyttsx3
Custom labelled Dataset

Fig. 2. Block diagram of proposed system.

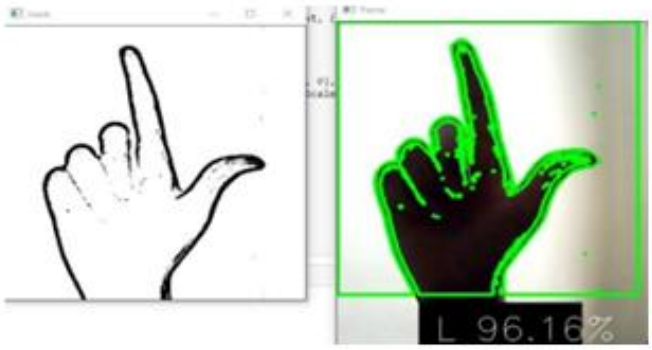


Fig. 3. Adaptive thresholding

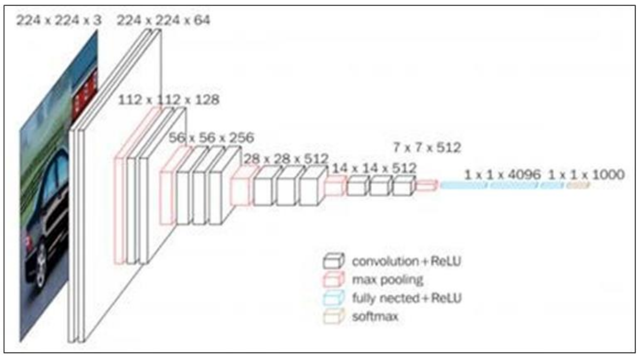


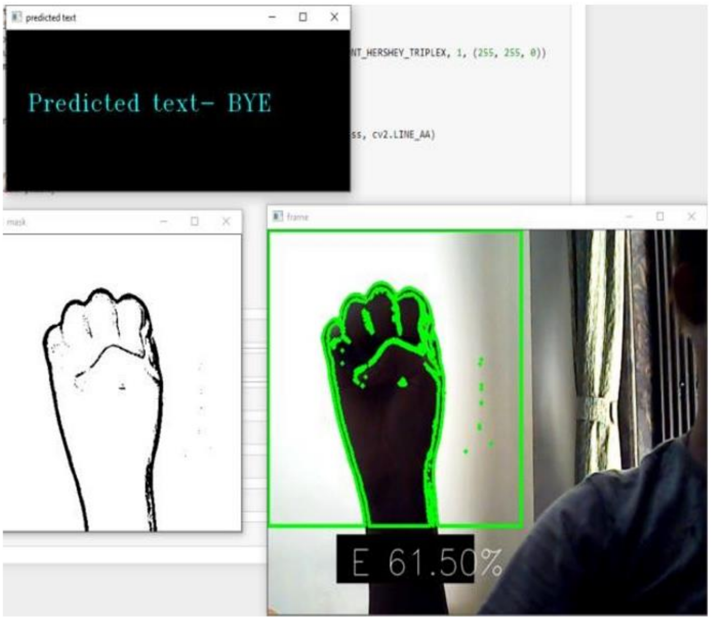
Fig. 4. VGG-16 architecture

Fig. 3 [16] depicts the architecture of the VGG-16. A RGB picture with a defined size of 224 by 224 pixels serves as layer-1's input. The picture is run through a convolutional layer stack (conv. Layer), using filters that are 3x3 in size. The spatial padding of the convolution layer input is such that the spatial resolution is kept after convolution; the convolution stride is fixed at 1 pixel. Five max-pooling layers, which come after some of the conv. layers, do spatial pooling. Over a 22 pixel frame, max-pooling is carried out using stride 2. After a stack of conventional layers come three FC layers. Each of the first two systems has 4096 channels, whereas the third system does categorization and has 1000 channels (one for each class). The soft-max layer is the last one. In every network, the FC layers are set up the same way. Rectification (ReLU) non-linearity is a feature that all hidden layers have. Our sign language motions are translated into English using the VGG16 neural network model. Due to its complexity and reliable results, the VGG16 architecture is regarded as one of the finest models for image categorization. To get the best results for our application, the VGG16 model is modified. For quicker prediction, the input image is scaled down to 128x128 pixels. The accuracy is higher compared to traditional neural networks, and it takes less time to predict pictures. The Python TensorFlow and Keras libraries are used to implement the model.

Text to Speech: Using Python's text-to-speech converter (pyttsx3), the conversion of text to speech is carried out [17]. The pyttsx3 software receives the anticipated words as input, and as an output it generates audible human speech. Two levels of headers should be numbered to indicate the text-to-speech outcome, which is a straightforward workaround but a vital feature because it conveys the feeling of a real vocal dialogue. Lower level headers are styled as run-in headings and are left unnumbered.

4 Experimental Results

The accuracy of using computer vision and machine learning to recognise sign language gestures is much improved, giving it an edge over other existing systems. Figure 5-7 illustrates how the model can predict all gestures with 99% training accuracy and a 0.1 second prediction time. The system becomes more robust to background circumstances thanks to the picture pre-processing. The model may be trained over a large number of im-ages thanks to the usage of a sizable bespoke dataset, increasing its confidence in predicting the right gesture. The number of erroneous predictions is greatly decreased by using a cutting-edge image processing neural network for this assignment.

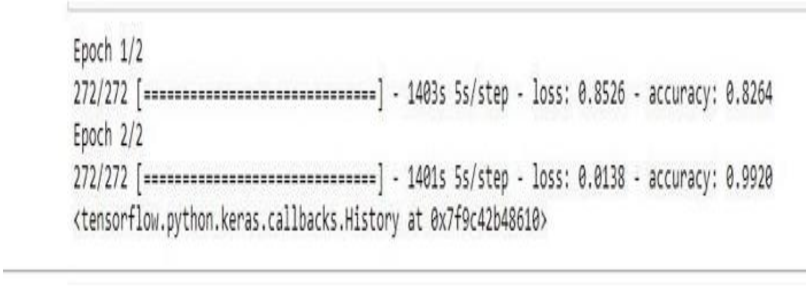


Predicted text- BYE

```
NT_HERSHEY_TRIPLEX, 1, (255, 255, 0))
ss, cv2.LINE_AA)
```

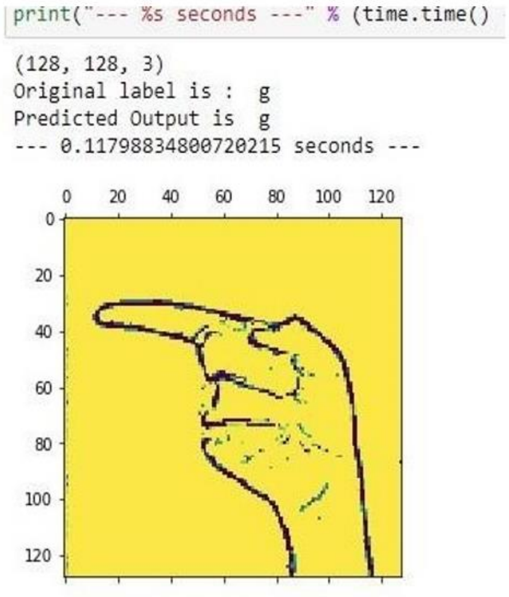
E 61.50%

Fig. 5. Gesture recognized to produce word “bye”.



```
Epoch 1/2
272/272 [==============================] - 1403s 5s/step - loss: 0.8526 - accuracy: 0.8264
Epoch 2/2
272/272 [==============================] - 1401s 5s/step - loss: 0.0138 - accuracy: 0.9920
<tensorflow.python.keras.callbacks.History at 0x7f9c42b48610>
```

Fig. 6. Average training accuracy (99%).



```
print("--- %s seconds ---" % (time.time()

(128, 128, 3)
Original label is :  g
Predicted Output is  g
--- 0.11798834800720215 seconds ---
```

Fig. 7. Average prediction time for gesture(0.1sec).

Table 1. Comparetive studies of results.

Parameters	**Hardware solution**	**Software solution**	**Proposed Deep Learning Model**
Papers Referred	[6], [4], [3], [10]	[8], [7], [2], [9], [1], [5]	**Current Paper**
Background and Lighting restrictions	None, To track gestures or information about gestures is gathered using sensors.	[2]Poor detection of palm area as whole frame is used;feature extraction and image pre-processing are carried out. [7]Whole frame is considered to clculate region of interest. Use skin segmentation and morphological operation for gesture detection. [8]MATLAB and 'bagOfFeatures' is used to predict gesture, accuracy drops when image	Eliminates background restrictions to a certain extent, as newer and advanced Image pre- processing algorithms offered by OpenCV library of python are used. Adaptive thresholding segments the skin (palm area) accurately along with Median and Gaussian blur which are used to filter the noise present in image.

		has face exposed in front of camera. [9] It only accepts static images and requires fixed background and proper lighting conditions. [1] accelerometer is used to gather gesture data, no restrictions.	
Drawbacks Observed	-Hardware solutions require specific hardware to be bought by the user which is expensive; time required to setup the apparatus of gesture recognition is complicated. User always needs to wear hardware for gesture recognition to work. Solutions are not practically feasible.	[2] Dataset size too small, only 4 gestures classified [7]Uses smartphone to send captured frame wirelessly to a server resulting in more delay. [8]Small dataset and poor Image pre-processing. [9]Requires user to keep palms in a fixed position. [1] Lesser accuracy obtained as accelerometer data is not accurate.	Only gestures involving use of alphabets are predicted, phrases are not included. Also, user requires a laptop with webcam functionality for system to work.
Accuracy (%) & Prediction Time	[3] Accuracy is 90%	[2] 97.5% [9] 83.3% [1] 53% [7] 97.2% and prediction time is 200ms [5] 98.6%	Testing **Accuracy obtained is 99% and average prediction time is 100ms.**

5 Conclusion

CNN and image processing techniques are useful for interpreting sign language motions. Since the images are processed using efficient image processing methods before being input into the model, it is resilient enough to operate under a variety of lighting and backdrop conditions. The technology operates effectively regardless of the surrounding conditions because to the several picture processing processes. With 99% training accuracy, the system is able to anticipate all gestures, and the prediction time is 0.1 seconds. The suggested system made the claim that it could translate gestures with high accuracy, speed, and affordability, enabling deaf and speech-impaired persons to interact with the outside world. The system is more user-friendly since it doesn't need expensive hardware or portable gadgets that the user must wear. The system can be improved in the future in a number of ways, including by expanding the model database to find more typical gestures used in daily life, offering a straightforward GUI interface for a positive user experience, and making the software open source so that the community can add more features to it.

References

1. S. Chandragandhi, R. Akash Raj, Muhammed Shamil, S. Akhil, PT Prabhashankar, “Real Time Translation of Sign Language to Speech and Text,” IARJSET, vol. 8, Issue 4, pp. 56–58, April 2021Author, F., Author, S.: Title of a proceedings paper. In: Editor, F., Editor, S. (eds.) CONFERENCE 2016, LNCS, vol. 9999, pp. 1–13. Springer, Heidelberg (2016).
2. K. Assaleh, T. Shanableh, M. Zourob, “ Low complexity classification system for glove-based arabic sign language recognition neural information processing ” . Springer Ber-lin/Heidelberg, 2012, pp. 262-268.. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
3. Kalsh E A, Garewal N S.Sign, “Language Recognition System”, International journal of computational engineering research, vol. 3, no. 6, pp. 15-21,2013.
4. C. Preetham, C., Ramakrishnan, G., Kumar, S., Tamse, A. and Krishnapura, “Hand talk-implementation of a gesture recognizing glove,” Texas Instruments India Educa-tors Conference, pp. 328-331,2013.
5. D. Harish, N. and Poonguzhali, S., 2015, “Design and development of hand gesture recognition system for speech impaired people,” International Conference on Industrial In-strumentation and Control (ICIC), pp. 1129-1133,2015.
6. S. Shaoo, M. Gauri, R.Kiran Kumar, “Sign Language Recognition: State of the Art,” ARPN Journal of Engineering and Applied Sciences, vol. 9, pp. 116–134, 2014
7. A.Bagwari, S. Bisht, S. Kaushik, M. Devrari, “A Hardware Model for the Helping of Speech Impared People,” Design Engineering, Issue 8, pp. 16636-16651,2021.
8. A. Chougule, S. Sannakki, V. Rajpurohit, “Smart Glove for Hearing -Impaired,”International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN, vol. 8, pp. 2278-3075, April 2019.
9. Praveen, N., Naveen K.,Megha M., “Sign language interpreter using a smart glove,”International Conference on Advances in Electronics Computers and Communications , 2014.
10. M. U. Kakde, M. Nakrani, A. Rawate, “A Review Paper on Sign Language Recognition System for Deaf and Dumb People using Image Processing,” International Journal of Engineering Research & Technology (IJERT) Volume 05, Issue 03, March 2016
11. P. Sridevi, T. Islam, U. Debnath, N. A. Nazia, R. Chakraborty and C. Shahnaz, "Sign Language Recognition for Speech and Hearing Impaired by Image Processing in MATLAB," IEEE Region 10 Humanitarian Technology Conference (R10-HTC), Malambe, Sri Lanka, 2018
12. Raheja, J. L., A. Mishra and A. Chaudhary. “Sign language recognition using SVM.” Pattern Recognition and Image Analysis, Vol. 26 Springer 2016.

13. K. Shenoy, T. Dastane, V. Rao, D. Vyavaharkar, "Real-time Indian Sign Language (ISL) Recognition," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bangalore, 2018
14. Aznaveh M, Mirzaei H., Roshan E., Saraee M.H, "A New and Improved Skin Detection Method Using Mixed Color Space," Advances in Intelligent and Soft Computing, vol 60. Springer, Berlin, Heidelberg.
15. Sruthi C. J and Lijiya A, "Signet: A deep learning based Sign Language Recognition System." International conference on communication and signal processing, April 2019
16. Blier, L. A brief report of the heuritech deep learning meetup#5,2016. Available at: https://blog.heuritech.com/2016/02/29/a-brief-report-of-the-heuritech-deeplearning-meetup-5/.
17. pyttsx3 2.7 project, https://pypi.org/project/pyttsx3/2.7, last accessed 2019/04/10.