iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL
FOR RESEARCH
IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 Issue: V Month of publication: May 2023

DOI: https://doi.org/10.22214/ijraset.2023.52371

Real-Time Sign Language Recognition System for Hearing and Speech Impaired People

Sanchana H T[1], Yashaswini N[2], Vidhyakrishna[3], Thejashree P[4], Prof. Roopa Banakar[5]
[1, 2, 3, 4, 5]Department of Computer Science and Engineering, Sapthagiri College of Engineering#14/5, Chikkasandra, Hesaraghatta main road, Bangalore-57 India

Abstract: People with hearing loss frequently use sign language to interact with their community and communicate with others because it is largely a visual form of communication. It requires the use of manual gestures, nonverbal facial clues, and body motions, unlike spoken language, to express thoughts and convey meaning. The goal of Sign Language Recognition (SLR) is to identify, interpret, and translate these signs into the appropriate speech or text. For those with speech and hearing impairments, sign language is an essential tool for communicating with others and exchanging ideas. This paper proposes a novel method for recognising individual alphabet signals in sign language so that words can be formed from them. This is accomplished through the application of a deep learning network, which can detect the signs and output the corresponding text. Additionally, the recognized individual characters can be sequentially utilized to form words, which can then be converted into voice output.

I. INTRODUCTION

Sign language is a crucial means of communication for tens of millions of individuals worldwide who experience hearing disabilities and an estimated 28 to 32 million people globally, and around 28 to 32 million people uses American Sign Language (ASL)in the United States for communication alone. Unfortunately, the vast majority of non-hearing-impaired individuals have limited familiarity or understanding of sign language, leading communication barrier. While sign Although several sign language recognition (SLR) systems have been created, effectively identifying sign language is challenging due to its complex structure, involving both delicate finger movements and broad arm gestures. Many SLR systems are vision, acoustics, radio frequency (RF), and inertial measurement unit (IMU) sensors based , but most of these systems are unable to continuously recognize sign language, which results in issues with accuracy. They divide sentences into gestures and perform isolated recognition, which is imprecise due to the challenge of detecting gesture boundaries in continuous signals. While certain vision-based systems can continuously recognize entire sentences, they are not effective in capturing delicate finger movements and are vulnerable to background noise and texture. To tackle these obstacles, we have produced a real-time end-to-end SLR system that employs an innovative approach for detecting manual or hand gestures.

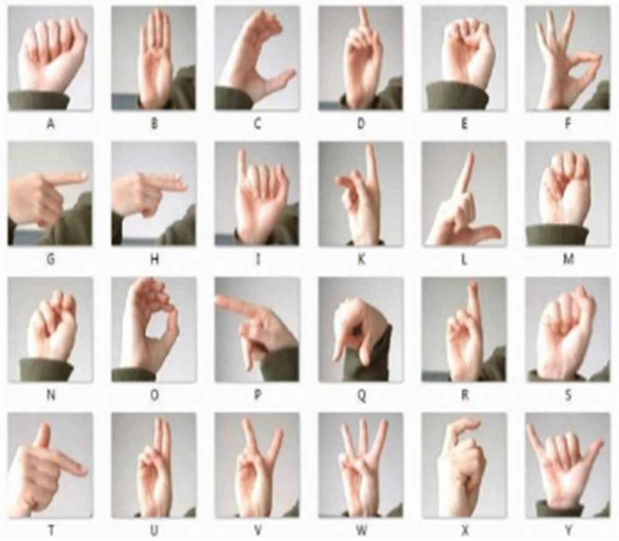

Fig. 1. Sign Language Hand Gestures

Sign language utilizes hand and body gestures to convey meaningful messages, and there are between 138 and 300 various kinds of sign languages used globally. In India, only around 250 certified sign language interpreters exist to serve a deaf population of approximately 7 million individuals, presenting a challenge in teaching sign language to this population. Sign Language Recognition aims to recognize these hand gestures and convert them into corresponding text or speech, employing deep learning techniques or methods., we can classify these hand gestures and produce corresponding text. For instance, recognizing the "A" alphabet in sign language can translate to "A" text or speech in English. The most popular neural network approach in deep learning is thought to be Convolutional Neural Networks (CNNs) and are widely used for Image/Video tasks. This system has the potential to greatly or substantially aid hearing and speech impaired people in communicating effectively.

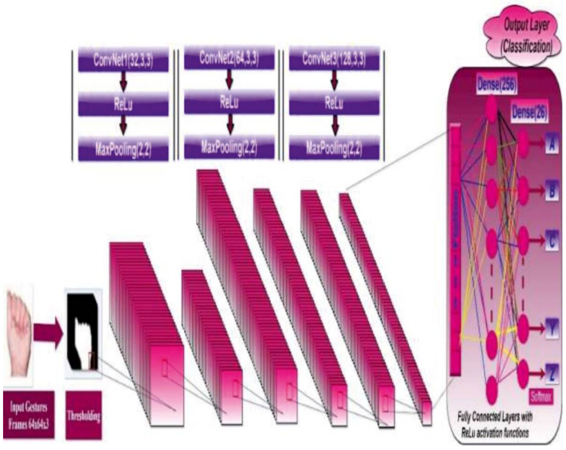

Fig. 2. Convolution Neural Networks

II. LITERATURE SURVEY

In [1], the article proposes an innovative approach for detecting manual or hand gestures in Argentinian sign language (LSA). The paper presents two significant contributions: the development of a database of handshapes specific to LSA and a supervised adaptation of self-organizing maps called ProbSom for image processing, descriptor extraction, and handshape classification. The ProbSom-based neural classifier, Utilizing the proposed descriptor, achieved an accuracy rate above 90%, which was compared to other state-of-the-art techniques such as SVM, Random Forests, and Neural Networks.

In [2], the proposed system comprises four main modules of the proposed system Data Acquisition, Preprocessing, Feature Extraction, and Classification. Histogram matching and Skin Filtering are both part of the preprocessing phase. Then, 24 different alphabets were recognized with a 96% recognition rate using the Eigenvalue weighted Euclidean distance-based Classification Technique and Eigenvector-based Feature Extraction.

In [3], The system concentrates on translating Sign to Speech with data rather than visuals, which is the more usual method. The method is comparable to image classification, but it also considers temporal factors. The method uses a Convolutional Neural Network to perform temporal learning and identify the frames of each image in a two-step process. The sign-to-speech module is shown by the resulting block diagram. The system gathers data from users' devices and sample frames at the right rate. A data preparation approach that eliminates noise and tracks important points then processes the frames.

In [4], The paper offers a system that can translate Indian Sign Language (ISL) into text in real-time. In order to categorise the symbol, the researchers present a deep learning methodology employing a CNN. In the first stage, a convolutional neural network implementation called Keras is used to create the classifier model. In phase two, a real-time system is added that finds the Region of Interest in the frame, which displays the bounding box, using skin segmentation. In order to identify the sign, the segmented region is then provided as an input to the classifier model. For the same subject, the system's accuracy was 99.56%, and in low light, it was 97.26%. The classifier was found to improve with different backgrounds and orientations of the gathered image. The RGB camera system is the primary focus of the researchers' approach.

In [5], A vision-based programme that converts sign language into text is being developed with the intention of enhancing communication between signers and non-signers.Video sequences are used in the proposed approach to extract temporal and spatial information. An RNN (Recurrent Neural Network) is trained on temporal features, and a CNN (Convolutional Neural Network) is utilized to recognize spatial features at inception. The study employs the American Sign Language Dataset.

III. METHODOLOGY

A. System Architecture

The proposed method concentrates on translating Sign language to Speech with data rather than visuals, which is the more usual method. The method is comparable to image classification, but it also considers temporal factors. The method uses a Convolutional Neural Network to perform temporal learning and identify the sign in a two-step process. The sign-to-text module is shown by the resulting block diagram. The system gathers data from captured sign image . A data preparation approach that eliminates noise and tracks important points then processes the image.

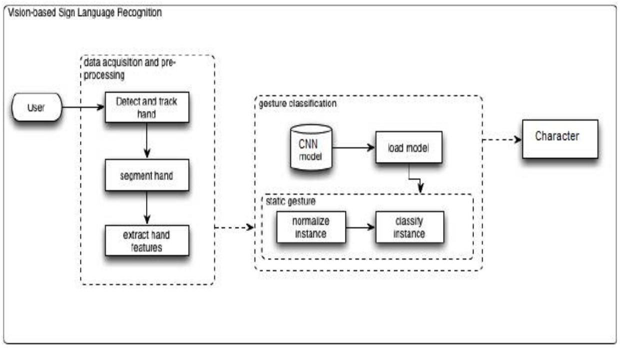

Fig 11: Block diagram of a sign-to-text translation

This subsection focuses on translating text into 3D sign language motion. The module works by identifying each alphabets in text and running text-to-sign over the same.

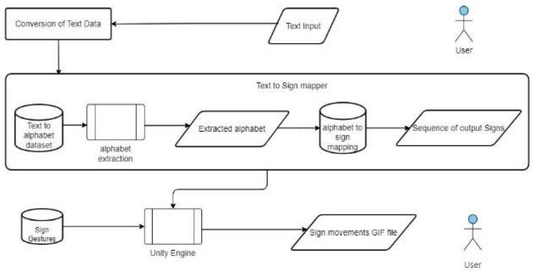

Fig.12 Block diagram of text-to-sign translation.

In the proposed model, the audio output is also supported in sign-to-text section by using Google Text-To-Speech API.

B. Dataset

We have used multiple datasets and trained multiple models to achieve good accuracy.

1) ASL Alphabet

The data is a collection of images of the alphabet from American Sign Language, separated into 27 folders that represent the various classes. The training dataset consists of 216 images which are 200x200 pixels. There are 27 classes of which 26 are English alphabets A-Z and SPACE.

2) Sign Language Gesture Images Dataset

The dataset with 27 different hand sign gestures which include A-Z alphabet gestures, also a gesture for space which means how the deaf (hard hearing) and dumb people represent space between two letters or two words while communicating. Each gesture has 8 images which are 50x50 pixels. CNN is well suited for this dataset for model training purposes and gesture prediction.

C. Data Pre-processing

An image is nothing more than a 2-dimensional array of numbers or pixels which are ranging from 0 to 255. Typically, 0 means black, and 255 means white. Image is defined by mathematical function f (x, y) where 'x' represents horizontal and 'y' represents vertical in a coordinate plane. The value of f (x, y) at any point is giving the pixel value at that point of an image. Image Pre-processing is the use of algorithms to perform operations on images. It is important to Preprocess the images before sending the images for model training. For example, all the images should have the same size of 200x200 pixels. If not, the model cannot be trained.

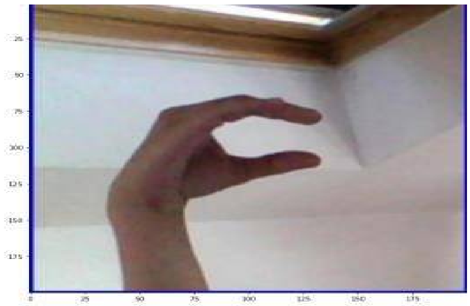

Fig. 3. Sample Image without Pre-processing

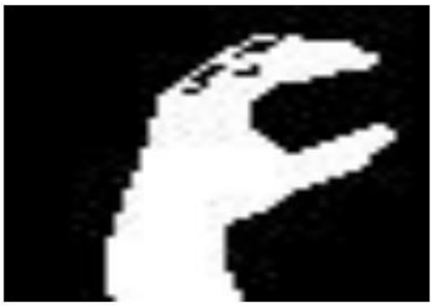

Fig. 4. Pre-Processed Image

D. Convolution Neural Networks (CNN)

Computer Vision is a field of Artificial Intelligence that focuses on problems related to images and videos. CNN combined with Computer vision is capable of performing complex problems.

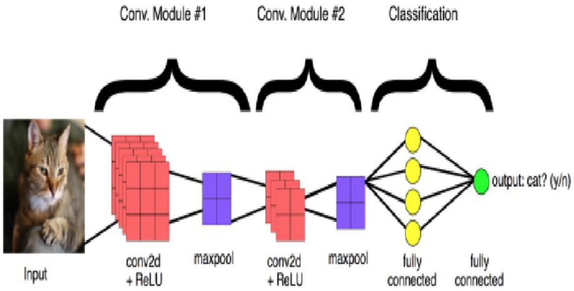

Fig. 5. Working of CNN

Feature extraction and classification are two important phases in CNN. A series of convolution and pooling operations are performed to extract the image features. The output matrix size decreases as we keep on applying the filters. In the last layer, the probability of the class will be predicted.

The main steps involved in convolution neural networks are:

1) Convolution

Convolution is nothing but a filter applied to an image to extract the features from it. We will use different filters to extract features like edges, and highlighted patterns in an image. The filters will be randomly generated. What this convolution does is, creates a filter of default size says 3x3 . After creating the filter, it starts performing the element wise multiplication starting from the top left corner of the image to the bottom right of the image. The results will be extracted feature.

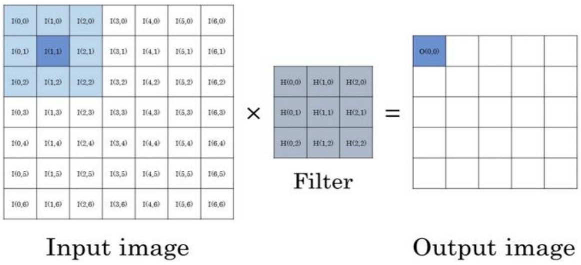
I(0,0) I(1,0) I(2,0) I(3,0) I(4,0) I(5,0) I(6,0)
I(0,1) I(1,1) I(2,1) I(3,1) I(4,1) I(5,1) I(6,1)
I(0,2) I(1,2) I(2,2) I(3,2) I(4,2) I(5,2) I(6,2)
I(0,3) I(1,3) I(2,3) I(3,3) I(4,3) I(5,3) I(6,3)
I(0,4) I(1,4) I(2,4) I(3,4) I(4,4) I(5,4) I(6,4)
I(0,5) I(1,5) I(2,5) I(3,5) I(4,5) I(5,5) I(6,5)
I(0,6) I(1,6) I(2,6) I(3,6) I(4,6) I(5,6) I(6,6)
×
H(0,0) H(1,0) H(2,0)
H(0,1) H(1,1) H(2,1)
H(0,2) H(1,2) H(2,2)
=
O(0,0)
Filter
Input image
Output image

Fig. 6. Convolution

The output matrix size decreases as we keep on applying the filters. Size of new matrix = (Size of old matrix — filter size) +1.

A frequent activation function in convolutional neural networks (CNNs) is the Rectified Linear Unit (ReLU). It is applied to each convolutional layer's output in a CNN.

The ReLU activation function is defined as:

f(x) = max(0, x)

where x is the input to the function.

The ReLU function is applied element-wise to each convolutional filter's output in the context of CNNs. The ReLU function's objective is to induce non-linearity into the network so that it can recognise more intricate patterns in the data.The ReLU function returns the value of the input when the input is positive. The function returns 0 when the input is negative. Due to this, the output of the ReLU function is a sparsely activated matrix with mostly zero and very few non-zero values. The network's capacity to learn valuable features is enhanced as a result of the sparsity property's ability to reduce the number of parameters in the network.

2) Pooling

The pooling layer will be used following the convolution process. The pooling layers will be used to decrease the image to a smaller size.

a) Max pooling

Max pooling is nothing but selecting the maximum pixel value from the matrix.

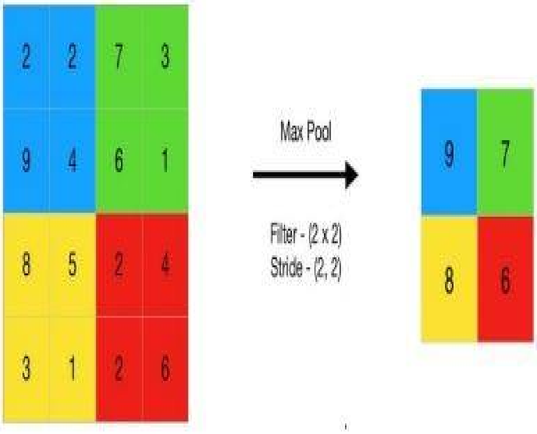

Fig. 7. Max Pooling

This method is helpful to extract the features with high importance or which are highlighted in the image.

3) Flatten

The obtained resultant matrix will be in multidimension. Flattening is converting the data into a 1- dimensional array for inputting the layer to the next layer. We flatten the convolution layers to create a single feature vector.

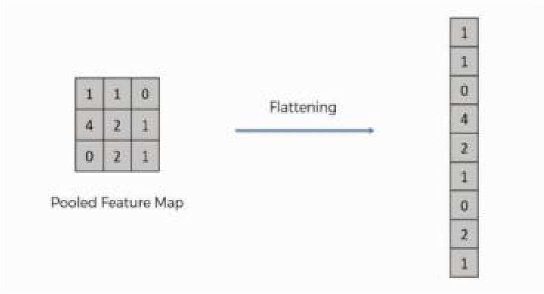

Fig. 9. Flatten

4) Full Connection

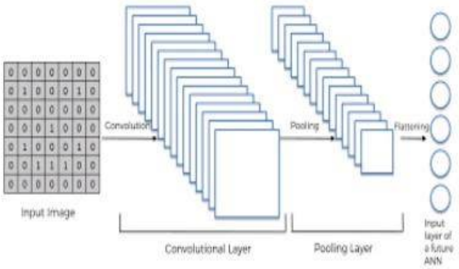

Fig. 10. Full Connection

A fully connected layer is simply a feed-forward neural network which makes use of two activation functions and they are ReLU and Softmax activation functions.

The softmax function is often used as the last activation function of a neural network to normalize the output of a network to a probability distribution over predicted output classes.

Softmax function formula:

$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

σ = softmax

$\vec{z}$ = input vector

e^{z_i} = standard exponential function for input vector

K = number of classes in the multi-class classifier

e^{z_j} = standard exponential function for output vector

e^{z_j} = standard exponential function for output vector

All the operations will be performed and prediction is obtained. The gradient descent backpropagation technique will be used to determine the loss and update the weights based on the ground truth.

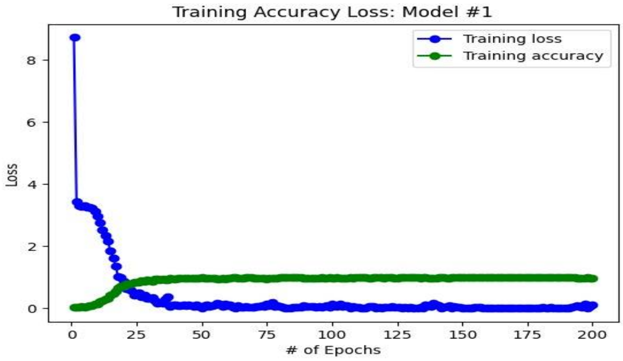

Fig.11 Training Accuracy Loss

A complete iteration through a given dataset is referred to as an epoch in machine learning when a model is being trained.
In the proposed model, epochs of 200 is used which means that the CNN will run the training data 200 times through the network, computing the loss function, using an optimizer to update the model parameters, and gradually improving its internal representations of the data.

IV. CONCLUSIONS

In conclusion, we have created a major and useful system that can understand sign language and translate it into text. However, due to its inability to recognise body motions or other dynamic gestures, our system is only able to recognise hand gestures for the alphabet from A to Z. We admit that there is opportunity for advancement and enhancement in subsequent iterations.

REFERENCES

[1] Ronchetti, Franco, Facundo Quiroga, César Armando Estrebou, and Laura Cristina Lanzarini. "Handshape recognition for Argentinian sign language using probsom." Journal of Computer Science & Technology 16 (2016).
[2] Singha, Joyeeta, and Karen Das. "Automatic Indian Sign Language Recognition for Continuous Video Sequence." ADBU Journal of Engineering Technology 2, no. 1 (2017).
[3] Sruthi Upendran, A Thamizharasi, “Embedded Sign Language Interpreter System for Deaf and Dumb People.” 2018 IEEE.
[4] TD SanjanaRaj, MV Beena, “. Indian Sign Language Numeral Recognition Using Region of Interest Convolutional Neural Network.” IEEE 2018.
[5] Kshitij Bantupalli, Ying Xie,” American Sign Language Recognition using Deep Learning and Computer Vision.” 2018 IEEE International.