ISSN No:-2456-2165

Sign Language Interpreter Using Computer Vision and LeNet-5 Convolutional Neural Network Architecture

Shreya Vishwanath
Department of CSE
JNTUH College of Engineering
Hyderabad, India

Shaik Sohail Yawer
Department of CSE
JNTUH College of Engineering
Hyderabad, India

Abstract:- A gesture is a form of sign language that incorporates the movement of the hands or face to indicate an idea, opinion, or emotion. Sign language is a way for deaf and mute persons to communicate with others by using gestures. Deaf and mute persons are familiar with sign language since it is widely used in their community, while the general public is less familiar. Hand gestures have been increasingly popular because they let deaf and mute people communicate with others. Many of these forms of communication, however, are still limited to specialized applications and costly hardware. As a result, we look at a simpler technique that uses fewer resources, such as a personal computer with a web camera that accomplishes our goal. The gestures are captured as images through a webcam and image processing is done to extract the hand shape. The interpretation of images is carried out using a LeNet-5 Convolutional Neural Network architecture.

Keywords:- *Gesture; Image Processing; Convolutional Neural Network; Numbers; Digits; OpenCV; LeNet-5; Parameters.*

I. INTRODUCTION

Communication is fundamental to a person's life. Dialogue allows a person to learn and grow. A person's ability to speak his thoughts and maintain pleasant social connections depends on his communication skills. Poor communication can ruin professional and personal relationships, making life difficult. A person's ability to interact is hampered when communication becomes a barrier for them, such as for the deaf or mute.

Communication among the deaf, mute, and the general public has become increasingly vital in everyday interactions. There are around 1.3 Million deaf and mute persons in India, yet there are only about 1000 qualified Sign Language Interpreters. However, learning and understanding a hand signal language are not easy for an average person. So, there is a need to bridge this communication gap using technology.

The paper discusses the flow of research process and explains each stage in the pipeline [Fig. 1]. It clearly explain how the data is collected and how the data is processed using image processing techniques. It also explains the usage of Convolutional network model building and validating it with new data.

II. REVIEW OF LITERATURE

Several researchers studied sign language interpretation using different techniques.

R Harini and colleagues employed computer vision to capture images and image processing to segment them [1]. They used a Convolutional Neural Network model to recognize gestures.

Pujianto Yugopuspito [2] used Convolutional Neural Network to recognize hand gestures in real-time using a mobile application and a Mobile Net algorithm to train images of 23 gestures.

In Ref. [3] Omkar Vedak proposed a system where hand gestures are processed to extract histograms of oriented gradients (HOG). Finally, an SVM classifier was used to recognize gestures.

Yann LeCun [4] uses multilayered networks trained with gradient descent to learn complex. high-dimensional and non-linear mappings from large collections of data named Convolutional Neural Networks. A typical convolutional neural network for recognizing characters dubbed LeNet-5 comprises 7 layers, not including the input, all of which contain trainable parameters.

Kanchan Dabre[5] used Haar Cascade Classifier to interpret hand signs. The handshape from continuous frames were identified by a variety of image processing methods. Before converting to audio, the video of commonly used full sentence gestures was turned into a text. Finally, a voice synthesizer translated the visible text into speech.

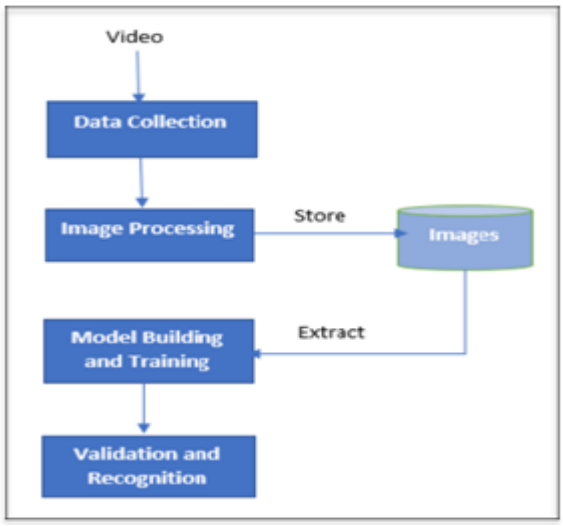

Fig. 1. Process Pipeline

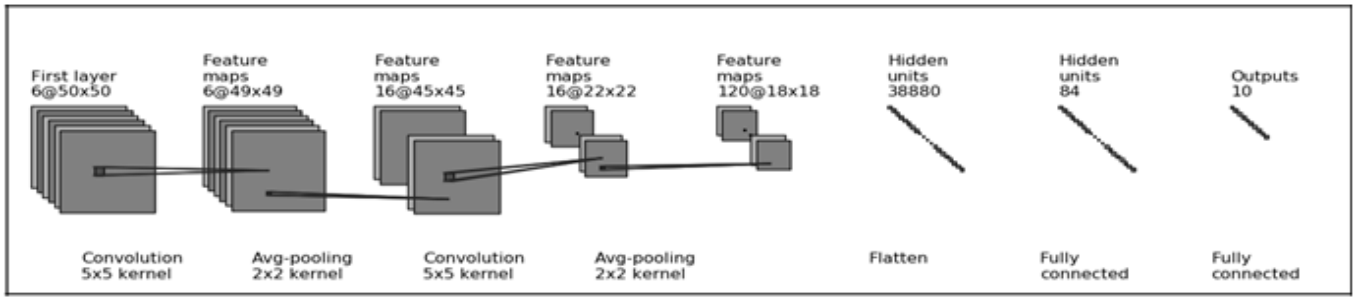

Fig. 2. LeNet-5 Architecture

III. METHODOLOGY

The proposed methodology has four stages [Fig. 1]. The first is the data collection stage, where the hand gestures are captured with a laptop's web camera using a computer vision library called OpenCV [1].

The second phase, image processing, is used to extract the important components of an image i.e., the shape of the hand. The unnecessary background components of the image are removed. In the next phase, a convolutional neural network model [1][2] is implemented using the Keras library of python. Finally, the new images are recognized by the validated CNN model.

A. Data Collection

Our model's primary data is a collection of hand gestures of ten digits. We capture 1500 images of each number ranging from 0 to 9 and augment them before storing them in the database. Firstly, we develop the histogram of the hand which is used for processing images. The hand boundary will be extracted from the images using the developed histogram. A python tool named OpenCV is used to record the video of the gestures presented to the camera [1]. The input video is split into images and sent to the image processing stage.

B. Image processing

Image processing involves extracting valuable information while ignoring distracting background and noise. For ease of processing, the photos are transformed from BGR to HSV. Backpropagation and morphological operations are performed using the histogram from the first phase. To remove noises, the image is smoothed and blurred. Finally, the hand border is derived using contours. The hand image appears as a white object on black background.[1][2][3][5]. After all the images are processed, the obtained images are augmented to get 3000 images of each number and stored in the database. The series of steps are shown in [Fig. 3].

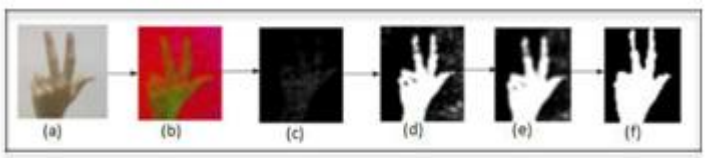

Fig. 3. Image processing: a) BGR, b) HSV, c) Backpropagation, d) noise removal, e) noise removal, f) Final Image

C. Model Building and training

For training our data, we chose to implement LeNet-5 architecture as it has a history of providing accurate results. Around 2500 images of each digit are trained by LeNet-5 Model. The remaining images are used for validation. The LeNet-5 model consists of three convolutional layers and two average pooling layers. [Fig. 2].

ISSN No:-2456-2165

Layer one has an image of 50x50 pixels as input and a convolutional layer with 6 feature maps. Each unit in each feature map is connected to a 5x5 neighborhood in the input. The size of the feature maps is 50x50 which prevents the connection from the input from falling off the boundary. Hyperbolic tangent function(tanh) is used as the activation function. Layer one contains 156 trainable parameters.

Layer two is an average pooling layer with 6 feature maps of size 49x49. Each unit in each feature map is connected to a 2x2 neighborhood in the corresponding feature map in layer one.

Layer three is a convolutional layer with 16 feature maps. Each unit in each feature map is connected to several 5x5 neighborhoods at identical locations in a subset of layer two's feature maps. Hyperbolic tangent function(tanh) is used as the activation function. Layer three contains 2,416 trainable parameters.

Layer four is an average pooling layer with 16 feature maps of size 22x22. Each unit in each feature map is connected to a 2x2 neighborhood in the corresponding feature map in layer three.

Layer five is a convolutional layer with 120 feature maps. Each unit is connected to a 5x5 neighborhood on all 16 of layer four's feature maps. Hyperbolic tangent function(tanh) is used as the activation function. Layer five contains 48,120 trainable parameters.

Layer six and layer seven are fully connected layers containing 38,880 and 84 units, respectively. The reason for selecting these numbers comes from the output design [4].

Finally, we have the output layer consisting of 10 classes (10 digits) and uses the SoftMax activation function. We used the Categorical Cross-Entropy loss function and Stochastic gradient descent optimizer with a learning rate of 1e-2.

D. Validation and Recognition

To determine the model's accuracy and effectiveness in recognizing digits, the model is evaluated using validation data from 500 photos of each gesture. For better comprehension, the accuracy throughout the epochs is calculated and represented as a graph. The training and validation losses are both significant in the early epochs, but the accuracy is poor. However, in the following epochs, the loss has continuously decreased while accuracy has improved to the highest point [Fig. 4]. As a result, the model is preserved and utilized to recognize gestures in new images.

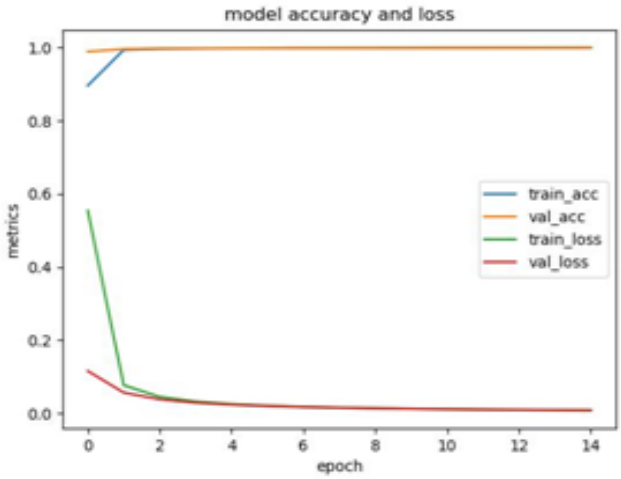

Fig. 4. Accuracy and Loss

In order to recognize the images in real time, new images are captured via webcam and are processed in similar fashion as we did in the second stage. The processed image is passed to the LeNet-5 trained model. The predict method available in Keras library is used to recognize the new image.

IV. CONCLUSION

In this paper, we have described how to use Convolutional Neural Networks to recognize hand motions in the presence of a simple background and acceptable lighting. We captured gestures through a webcam and trained a LeNet-5 model for 10 gestures. The results show that our best model, which is based on ten categories of Hand Signs, has a training accuracy of 99.8% and a validation accuracy of 90% for a total set of 30,000 images. The amount of data and the model's architecture have a significant impact on recognition accuracy. To improve the work, more signs, as well as signs in multiple languages, can be included. In the future, a mobile-based application for the convenience of use may be developed.

REFERENCES

[1]. R. Harini, R. Janani, S. Keerthana, S. Madhubala and S. Venkatasubramanian, "Sign Language Translation," 2020 6th International Conference on Advanced Computing and Communication Systems (ICACCS), 2020, pp. 883-886, doi: 10.1109/ICACCS48705.2020.9074370.

[2]. Pujianto Yugopuspito, I. Made Murwantara, and Jessica Sean. 2018. Mobile Sign Language Recognition for Bahasa Indonesia using Convolutional Neural Network. In Proceedings of the 16th International Conference on Advances in Mobile Computing and Multimedia (MoMM2018). Association for Computing Machinery, New York, NY, USA, 84–91. DOI: https://doi.org/10.1145/3282353.3282356

[3]. Omkar Vedak, Prasad Zavre, Abhijeet Todkar, Manoj Patil “Sign Language Interpreter using Image Processing and Machine Learning “, unpublished.
[4]. Y. Lecun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE, vol. 86, no. 11, pp. 2278-2324, Nov. 1998, doi: 10.1109/5.726791.
[5]. K. Dabre and S. Dholay, "Machine learning model for sign language interpretation using webcam images," 2014 International Conference on Circuits, Systems, Communication and Information Technology Applications (CSCITA), 2014, pp. 317-321, doi: 10.1109/CSCITA.2014. 6839279.