Sign Language Recognition

Abstract:

Communication serves as a vital tool for sharing ideas, thoughts, and information for exchange knowledge, and convey ideas effectively. But there are some people with disabilities who cannot speak or hear (deaf). This Hand gesture is one of the methods used in sign language for communication. Sign language recognition is to track and recognize the emotion of humans made with hands and fingers, etc. The technique that has been implemented here converts the gestures from sign language to a spoken text language which is easily understood. The gestures that have been translated include alphabets from static images. The aim is to represent features that will be learned by a system known as convolutional neural networks (CNN), which contains four types of layers: convolution layers, pooling layers, nonlinear layers, and fully connected layers. The model is trained on the alphabets first and then tested whether the system is providing the correct output or not. Through the training and testing of a model trained on alphabet gestures, the system demonstrates its capability to recognize and interpret hand gestures with a high degree of accuracy.

1.Introduction

There are so many languages in the world in which a unique language is needed to express the thoughts and opinions of the disabled people, which will be understandable to ordinary people, and such a language is named sign language.

A sign language recognizer bridges the gap between the deaf and the general population. Sign language uses hand gestures to convey the messages or information. Sign language can vary from one part of the world to another. Due to this, people find difficulty in communicating with normal people because normal people cannot understand sign languages. The availability of sign language translators is limited, and these translators have many limitations. This led to the development of a sign language recognition system, which can automatically translate sign language into the text by effective pre-processing and accurate classification of the signs.

In the proposed system, Convolutional Neural Network (CNN) is used to classify images of sign language because convolutional networks are faster in feature extraction and classification of images over other classifiers and OpenCV library facilitates the feature of video capture for recognizing or classifying the hand gestures which represent

alphabets and uses the American sign language (ASL) images as the training data for model.

2.Existing System:

Sign language recognition by hand gestures has played a significant attention in recent years, driven by advancements in deep learning, particularly Convolutional Neural Networks (CNN), and computer vision technologies such as OpenCV. This research suggested the use of filters in the sign language recognition algorithm because the existing system has low accuracy as it faced issues with background environment.

3.Objectives:

The objective of this project is to translate information shared through hand gestures int to text.so that the communication problem between a normal and hearing-impaired person can be easily solved. The project seeks to create an accurate recognition system that can easily identify alphabets using hand gestures with high precision.

4.Proposed System

In this project, the masking and filtering of images plays an important role. It improves the accuracy of identifying the hand signs even in various light conditions. The image colors, brightness and grey scaling is done through adjusting the trackbars according to the lighting conditions. After recognizing the sign, the image is further processed and the final result, which is the alphabet, is obtained.

5.Requirements:

The software requirements of this system are python, Open-Source Computer Vision Library (OpenCV), TensorFlow, Keras and NumPy. TensorFlow is an open-source machine learning tool that is used to train the data which is images. This framework makes use of OpenCV an open-source python library for computer vision and machine learning. NumPy is a Python library that adds support for big, multidimensional arrays and matrices, as well as working with wide range of mathematical functions. The proposed system recognizes alphabets the words of sign language. Background elimination is done by selecting color ranges in the trackbar. Thus, this automatically recognizes the hand region.

6.Model for Recognition System:

6.1 Data Acquisition:

Data acquisition of various alphabet hand signs is the first step in the sign language recognition system. The data can be obtained in a variety of ways. The method of taking visual photographs, such as those of a physical scene. These visual photos should

undergo preprocessing to optimize the input image data by reducing unnecessary noise and to enhance essential image features for more processing.

6.2 Rescaling image size:

Image resizing is important to increase or decrease the total number of pixels. In the proposed system, the images are resized by decreasing the number of pixels to 64*64. As for size increases, the time it takes to process also increases. So, the reduction of size is being done as a step of preprocessing

6.3Grayscale Conversion:

Grayscale conversion is the process of converting a color image in to a grayscale image. In grayscale images, each pixel's value represents the brightness level which ranges from 0 (black) to 255 (white). Converting images to grayscale simplifies processing tasks, reduces computational complexity, and removes color information, focusing only on the image's structural details. In OpenCV, grayscale conversion is done by using the cv2.cvtColor() function. By specifying the cv2.COLOR_BGR2GRAY flag, the function converts a color image from the BGR (Blue-Green-Red) color space to grayscale.

6.4Canny Edge Detection:

Canny edge detection is a widely used edge detection algorithm. It aims to identify the edges within an image while minimizing noise and preserving important edge features.

6.5Obscure The Image:

The proposed system used a masking process to hide some portions of an image to reveal some portions which help to identify the edges and features of images easily. Thus the greyscale image is converted into a binary image. The Convolution layer is often the first layer. The picture is received by it (a matrix of pixel values). Assume the input matrix is read from the image's top right corner. After that, the program selects a smaller matrix, known as a filter.

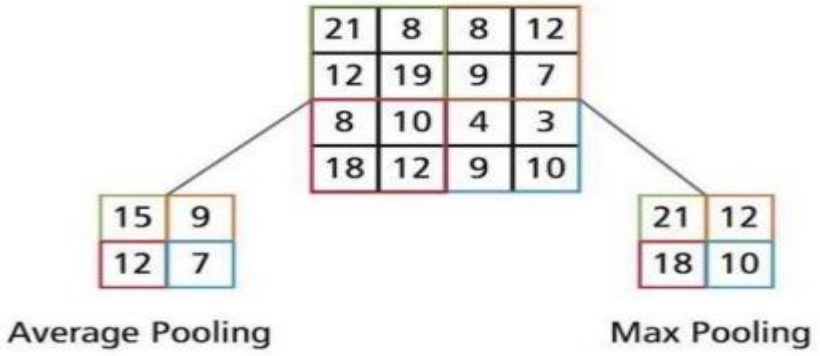
6.6 Convolution Layer:

Convolution layer extracts the low-level features such as edges and corners in the first step. Then upper-level layers extract functionality at a higher level. In CNNs, for the 3D convolution process. The input is N x N x D in dimension, and it is convolved with the H kernels, each of which is k x k x D in size. A mathematical procedure which takes two inputs, such as an image matrix and a filter or kernel, is known as convolution. The image matrix is a digital representation of image pixels, and the filter is another matrix that is used to process the image matrix. The kernel's size is much smaller than the image size, allowing us to process any aspect of the image. Apply a filter to the image matrix in this layer. Any number of convolution layers can be added based on the features to be

removed. The convolution function needs four arguments: the first is the number of filters, the second is the structure of each filter, the third is the input shape, and the fourth is the image form and 4 resolutions. The triggering function to be used is the fourth argument. The activation function determines which neuron can fire next.

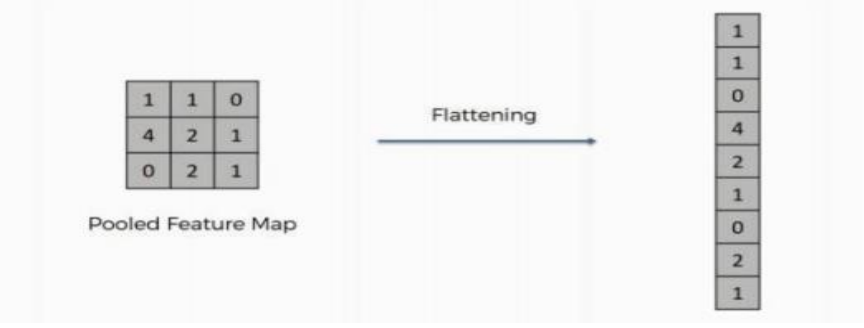
6.7 Pooling Layer:

A pooling layer is another building block of CNN. The main aim of pooling is to down sample the image matrix into a smaller matrix an to reduce the difficulty in computing the large image matrix obtained from the convolution operation. After the convolution operation, pooling the operation needs to be performed on the resultant matrix to reduce the complexity in processing these matrices.



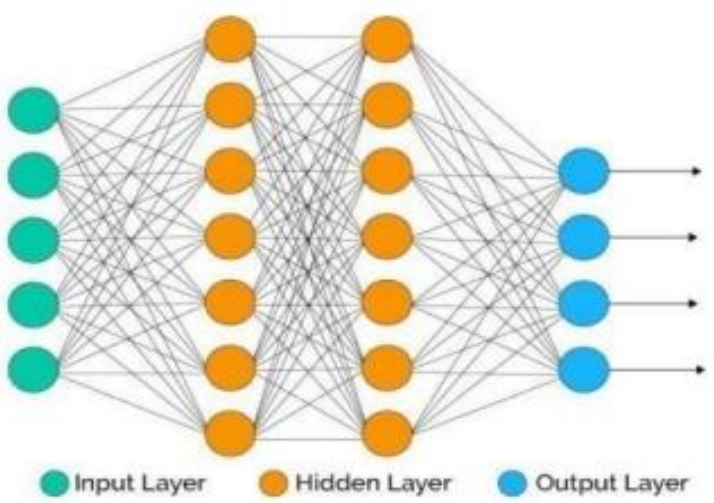
6.8 Flattening Layer:

Flattening is the method of transforming a pooled function map into a single column that can be transferred to a completely connected layer. Flattening is the process of transforming data into a one-dimensional sequence so that it can be passed on to the next layer. To make a single long function vector, flatten the contribution of the convolutional layers. It is attached to a totally connected sheet, which is the final classification model.



6.9 Fully Connected Layer:

The final pooling or convolutional layer's contribution, which is flattened before being fed into the completely connected layer. is the entry to the fully connected layer. In this layer, each neuron corresponds to weight and these weights are chosen randomly. To calculate the most reliable weights, the CNN network's completely linked component goes through its backpropagation process. Weights are allocated to each neuron that prioritizes the most suitable mark. Finally, the neurons "vote" for one of the marks. The final layer is used to get the probability of the input being in a specific class after going through the Entirely Connected Layer connected layers.



7.Video Capture and Recognition:

7.1 video capture:

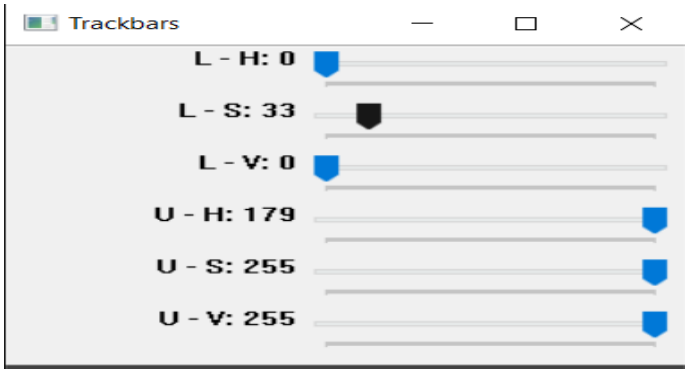
Video Capture is done by OpenCV using the cv2.VideoCapture class. This allows you to capture video from a camera or a video file. Then imshow class is used to show the current frame in window and a cv2.release class is used to release the captured image which is used by model to recognize the sign and a class in OpenCV called destroyAllWindows() to close OpenCV’s all windows

7.2 Creating trackbars for changing image color:

A trackbar is an element of OpenCV which is a graphical user interface element that allows users to select a value in a range by sliding a knob along a bar. It is used to adjust values dynamically in an application without restarting the program. Trackbars

are particularly useful for tasks such as adjusting the threshold level, changing the brightness of an image, or selecting a specific color range. specify the name of the

trackbar and the name of the window where it will appear. A pointer to an integer variable which stores the current position of the trackbar and define the maximum value of the trackbar. Then create a callback function that will be called whenever the trackbar's position is changed.

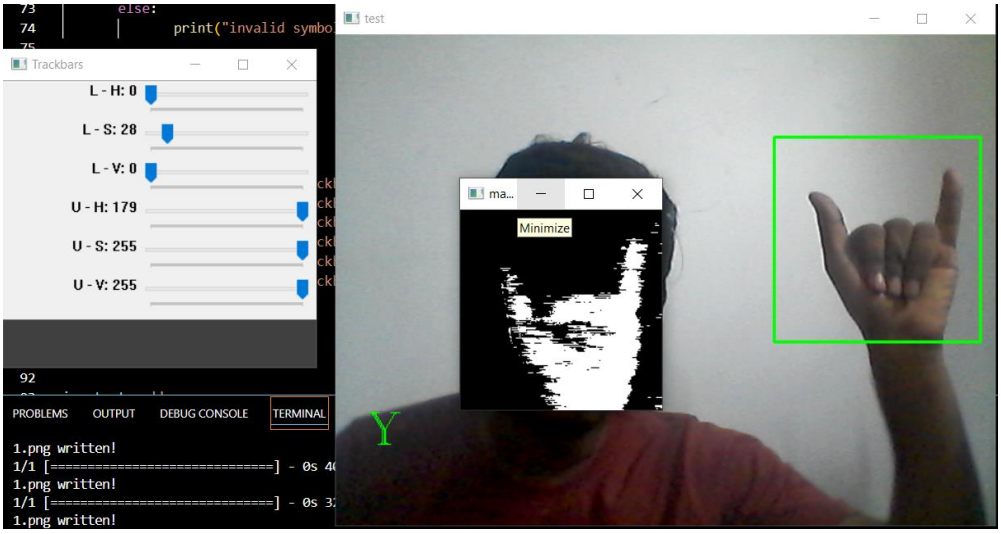


7.3 Image masking:

Masking is a technique used to focus on only certain parts of an image while ignoring the other part. This is done by creating a mask that specifies which parts of interest in an image, and then applying this mask to perform operations only on those regions.

7.4 Sign recognition:

The image which is in captured in the masked portion after adjusting the color and brightness through trackbars is loaded and the trained model is used to predict the class for the image containing a sign and the predicted alphabet is shown in the text format.



8.Result

The device was tested with real-time video feedback, and the results were much more accurate. In any light condition, the sign language recognition system's success is perfect. On all the signs, the highest accuracy is 93%. This means that the machine can understand multiple signs and is able to predict the alphabet.

9.Conclusion

The proposed project of sign language recognition system is successful in predicting the signs of alphabets in different lighting conditions. The dynamic adjustment of the color and brightness along with masking made the CNN model predict more accurately in difficult light conditions. The proposed system uses CNN for the training and classification of hand sign images. For training the model, a total of 1750 static images for each alphabet sign are used to train and get the accurate output. Finally, the output of the recognized sign is shown in the text format. The system is capable of recognizing 26 English alphabets. Thus, this is a user-friendly system which can be used by all physically impaired people.

References:

1. Anderson, R., Wiryana, F., Ariesta, M. C. & Kusuma, G. P. Sign language recognition application systems for deaf-mute people: A review based on input-processoutput. Proced. Comput. Sci. 116, 441– 448. https://doi.org/10.1016/j.procs.2017.10.028 (2017).

2. Mummadi, C. et al. Real-time and embedded detection of hand gestures with an IMUbased glove. Informatics 5(2), 28. https://doi.org/10.3390/informatics5020028 (2018).

3. Rivera-Acosta, M., Ortega-Cisneros, S., Rivera, J. & Sandoval-Ibarra, F. American sign language alphabet recognition using a neuromorphic sensor and an artificial neural network. Sensors 17(10), 2176. https://doi.org/10.3390/s17102176 (2017).

4. Ye, Y., Tian, Y., Huenerfauth, M., & Liu, J. Recognizing American Sign Language Gestures from Within Continuous Videos. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2145–214509 (IEEE, 2018). https://doi.org/10.1109/CVPRW.2018.00280.

5. Ameen, S. & Vadera, S. A convolutional neural network to classify American Sign Language fingerspelling from depth and colour images. Expert Syst. 34(3), e12197. https://doi.org/10.1111/exsy.12197 (2017).