ARTIFICIAL INTELLIGENCE MINI PROJECT

SIGN LANGUAGE RECOGNITION USING DEEP NEURAL NETWORK

Aishwarya Jayaraman RA1911026010102

Anshuman Pal RA1911026010117

Sankalp Jain RA1911026010119

ABSTRACT

- Sign language is one of the oldest and most natural form of language for communication. Since most people do not know sign language and interpreters are very difficult to come by, we have come up with a real time method using neural networks for detecting sign language. Our method detects hand gestures and predicts the sign language and gives the corresponding English alphabet.
- To provide an easy immersive augmented experience which is also gesture enabled, we employ a web camera which is integrated with OpenCV libraries through a compiler.
- With the advent of Artificial Neural Networks and Deep Learning, it is now possible to build a system that can recognize objects or even objects of various categories (like red vs green apple). Utilizing this, here we have an application that uses a deep learning model trained on the ASL Dataset to predict the sign from the sign language given an input image or frame from a video feed.
- In this method, the hand is first passed through a filter and after the filter is applied the hand is passed through a classifier which predicts the class of the hand gestures. This method provides 96.43 % accuracy for the 26 letters of the alphabet.

ISSUES IN EXISTING SYSTEM

- Sign language is used very commonly among differently abled people, but many people don’t know sign language and interpreters are difficult to come by. There’s no proper sign language detecting system available currently and the ones which exist have low predicting accuracy.

PROBLEM STATEMENT

- For interaction between normal people and D&M people a language barrier is created as sign language structure which is different from normal text. So they depend on vision based communication for interaction.
- If there is a common interface that converts the sign language to text the gestures can be easily understood by the other people. So research has been made for a vision based interface system where D&M people can enjoy communication without really knowing each other's language.
- The aim is to develop a user friendly human computer interfaces where the computer understands the human sign language.

OBJECTIVE

- To make a sign language detector which can detect English alphabets through hand gestures.
- To recognize objects using ANN and Deep learning.
- To capture frames from videofeed using opencv.
- To predict sign captured with a confidence level greater than 20%. If sign is predicted with a low confidence level i.e. between 20%-50% it is presented with a maybe sign, if it is with a high confidence level i.e. above 50% it is presented in blue colour with confidence percentage.

PROPOSED METHODOLOGY

- **Dataset**

We trained the network on kaggle dataset of ASL Alphabet. The dataset contains 87,000 images which are 200x200 pixels, divided into 27 classes (26 English Alphabets and NOTHING).

- **Data Augmentation**

To train the model for better real-world scenarios, we have augmented the data using brightness shift (ranging in 20% darker lighting conditions) and zoom shift (zooming out up to 120%).

- **Transfer Learning (Inception v3 as base model)**

We use Google's Inception v3 as the base model. The first 248 (out of 311) layers of the model (i.e. up to the third last inception block) are locked, leaving only the last 2 inception blocks for training and also remove the Fully Connected layers at the top of Inception network. We then create our own set of Fully Connected layers and add it after the inception network so as to conform the neural network for our application (consists of 2 Fully Connected layers, one consisting of 1024 ReLu units and the other of 29 Softmax units for the prediction of 29 classes). The model is then trained on the set of new images for the ASL Application.

- **Using the model for the application**

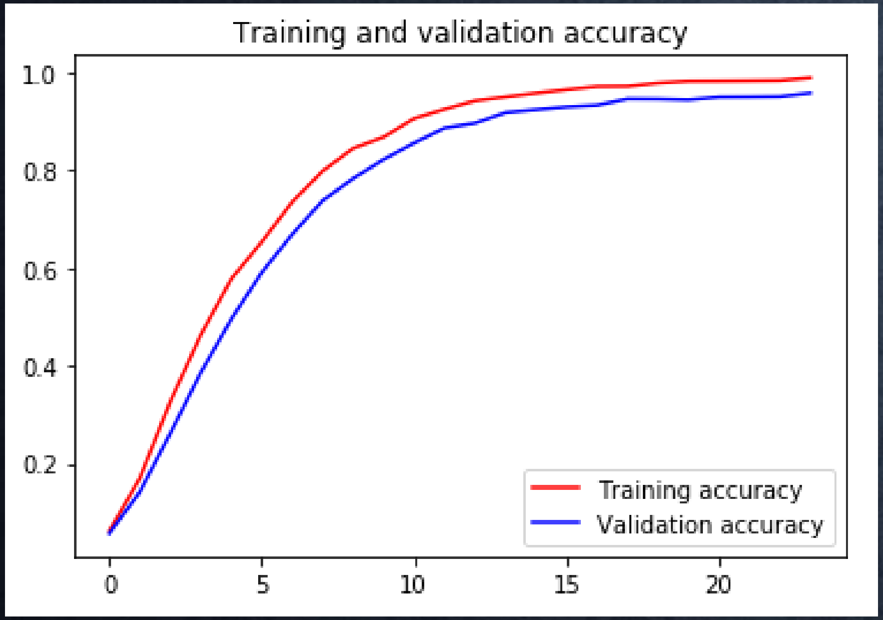
After the model is trained, it is then loaded in the application. OpenCV is used to capture frames from a video feed. The application provides an area (inside the green rectangle) where the signs are to be presented to be detected or recognized. The signs are then captured in frames, the frame is processed for the model and then fed to the model. Based on the sign made, the model predicts the sign captured. If the model predicts a sign with a confidence greater than 20%, the prediction is presented to the user (LOW confidence sign predictions are predictions
above 20% to 50% confidence which are presented with a Maybe [sign] - [confidence] output
and HIGH confidence sign predictions are above 50% confidence and presented with a [sign] - [confidence] output where [sign] is the model predicted sign and [confidence] is the model's confidence for that prediction). Else, the model displays nothing as output.

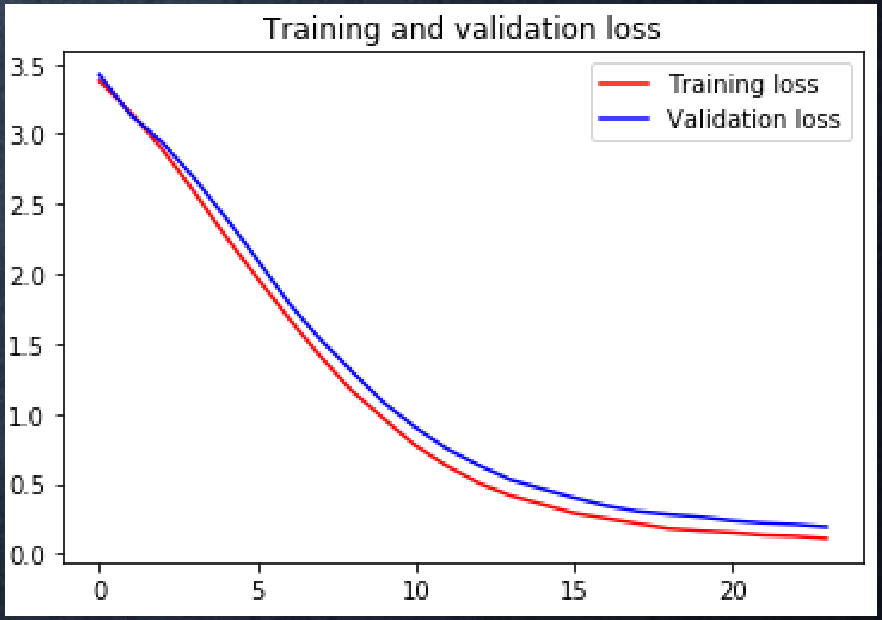
RESULTS

For training, Categorical Cross entropy was used to measure the loss along with Stochastic Gradient Descent optimizer (with learning rate of 0.0001 and momentum of 0.9) to optimize our model. The model is trained for 50 epochs and step per epochs is 100. The results are displayed below:

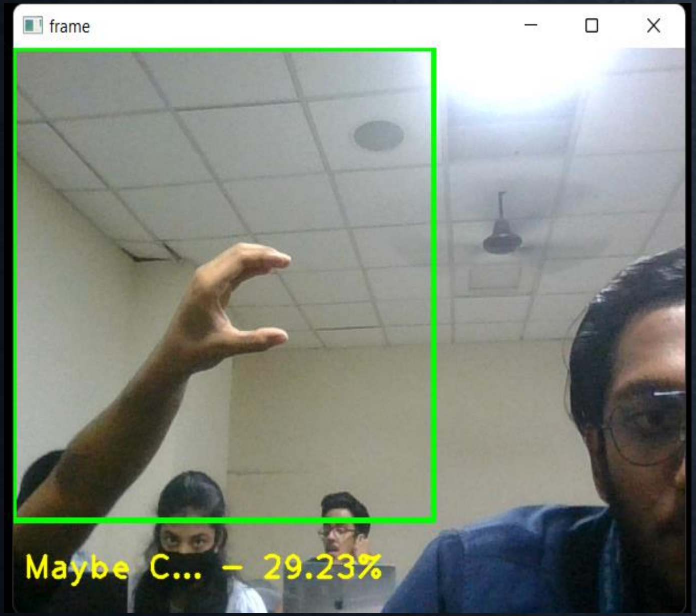
Metric	Value
Training Accuracy	98.83
Training Loss	0.1086
Validation Accuracy	96
Validation Loss	0.1961
Test Accuracy	96.4285

Graphical Results

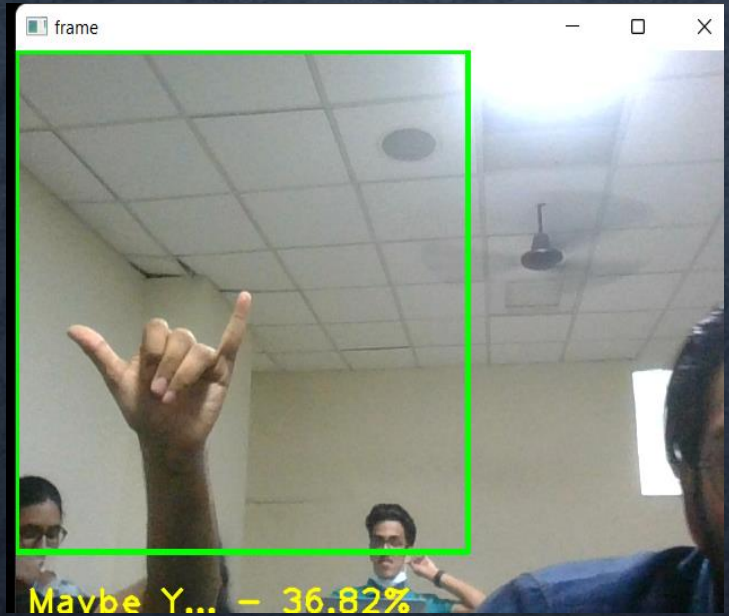




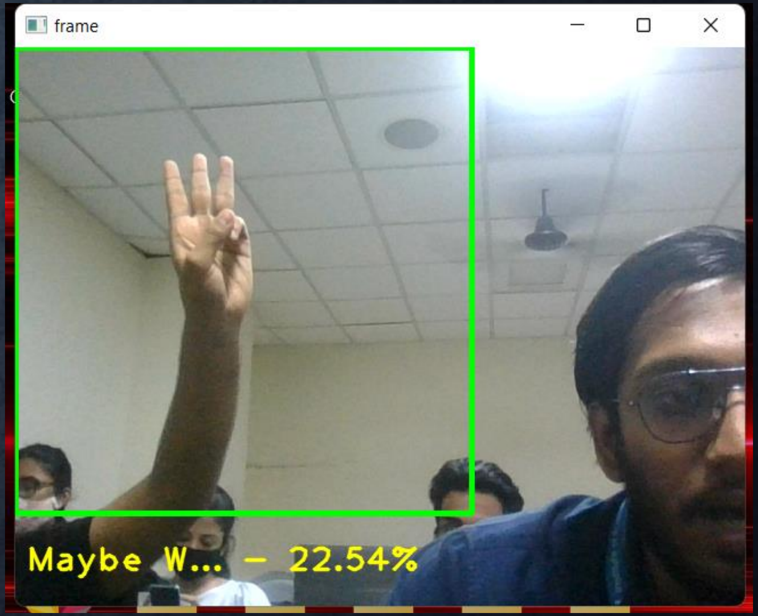
DEMO AND SCREENSHOTS



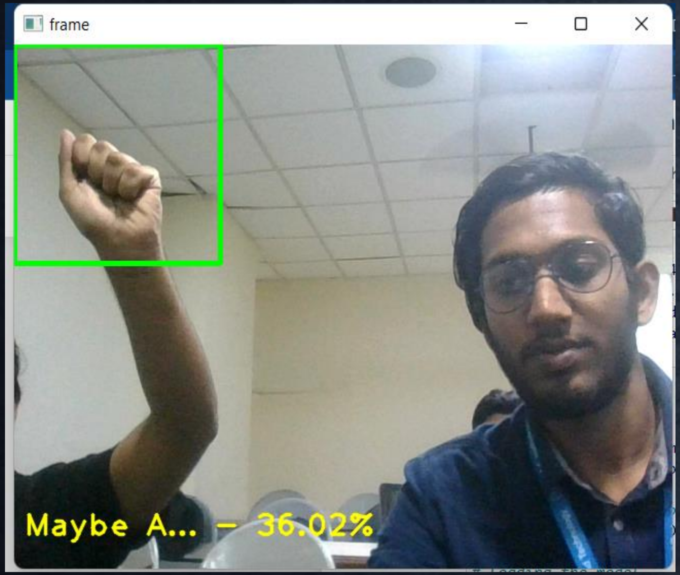
Maybe C... – 29.23%



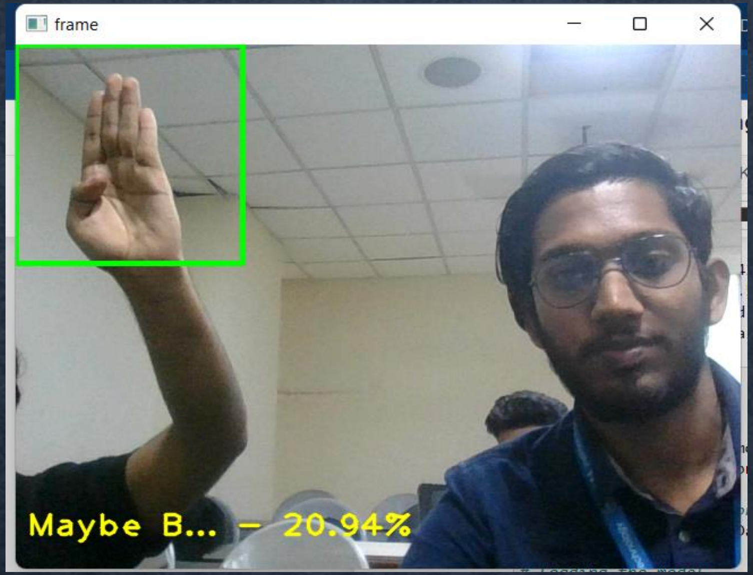
Maybe Y... – 36.82%



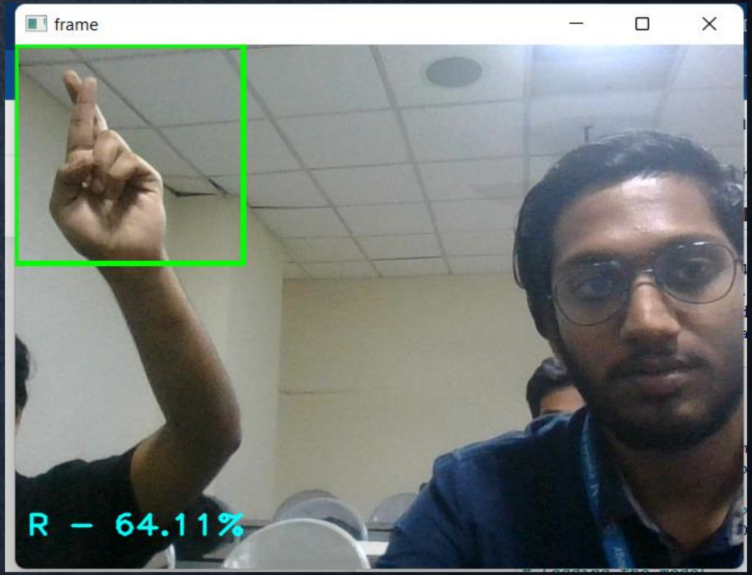
Maybe W... – 22.54%



Maybe A... – 36.02%



Maybe B... – 20.94%



R – 64.11%

THANK YOU!!