Sign Language Interpreter Using CNN and SVM

1st Dr. Dattaprasad Torse
HOD Electronics and Communication Department
Dr. M. S. Sheshgiri Belgavi Campus

2nd Bhuvan Budavi
Electronics and Communication Department
Dr. M. S. Sheshgiri Belgavi Campus

3rd Darshan Gadade
Electronics and Communication Department
Dr. M. S. Sheshgiri Belgavi Campus

4th Rakshit Patil
Electronics and Communication Department
Dr. M. S. Sheshgiri Belgavi Campus

5th Bhuvaneshwari R
Electronics and Communication Department
Dr. M. S. Sheshgiri Belgavi Campus

***Abstract*—— Sign Language is used by the deaf and voiceless community to be able to communicate with others, but the most commonly faced problem here is that everyone around may not be able to understand sign language. The main motive behind this system is to bridge the communication gap between the communities, therefore, establish the interaction between the speechless community to communicate with others. Hand gestures differ from one person to another person in shape and orientation, therefore, a problem of linearity arises. Recent systems have come up with various ways and algorithms to accomplish the problem and build this system. Algorithms such as KNearest neighbors (KNN), Multi-class Super Vector Machine (SVM), and experiments using hand gloves were used to decode the hand gesture movements before. In this paper, a comparison between KNN, SVM, and CNN algorithms is done to determine which algorithm would provide the best accuracy among all. Approximately 29,000 images were split into test and train data and preprocessed to fit into the KNN, SVM, and CNN models to obtain an accuracy of 93.83**

***Index Terms*—ASL, CNN, KNN, SVM**

I. Introduction

Sign language is a language used by deaf or hearingimpaired communities or the speechless to establish communication with others. It is a visual language that involves moments of center body parts such as hands and facial parts to convey the message. Hand gesture recognition technology is becoming increasingly relevant to recent growth that facilitates communication and provides a natural means of interaction that can be used across a variety of operations. Hand sign recognition or hand gesture recognition is one of the most active areas in humancomputer interaction that gives the machine the ability to capture and translate hand gestures. It then understands the command and executes it as directed. It provides an effortless way to interact without the use of any sensors or external devices. The goal here is to implement HCI's ability to nearby human-human interaction by modeling a sign language recognition system that would aid in predicting the context of dialogue between a person to another with the help of an interlocutor, here it is the system. This system makes use of various classifiers for hand detection and uses skin segmentation for recognizing the gestures and empirical tracking method that can dynamically change according to the stage of action. A gesture allows an individual to convey information to another person irrespective of whether they understand the message or not. This system also provides the facility of learning hand sign language with speech recognition that helps all those who want to learn the language. The world is now becoming more disabled-friendly making them feel much more normal and the system will make them independent without the need of a third person as a translator. This report will focus on American Sign Language. American Sign Language (ASL) American Sign Language was created back in 1817 for the deaf students. It was an attempt to represent the syntax and structure of English language on the hands of students and was a hope that if deaf students had access to the structure of English, then they could acquire the language. By the year 1835, ASL was used as the language to instruct and communicate with the student in schools for the deaf Sooner the use of this language spread across the world to communicate with the speechless. In the present world, reliable estimates for American ASL users range from 250,000 - 500,000 persons, including a number of children of deaf adults.

American sign language is used in this project to understand the hand gestures and convert them into their corresponding meaning. It is a sign language used by the speech impaired for communication. This hand sign language is a natural language that has identical linguistic properties as the spoken languages, without grammar usage, and is widely expressed by movements of the hands and face. There are various sign languages used depending upon the geographical factor. American sign language is the most used sign language which uses gestures to represent English letters.

Identify applicable funding agency here. If none, delete this.

II. Problem Statement

The problem identified is the communication gap between those who cannot speak and others who can but do not know sign language. Our solution is a system that serves as a translator and helps in understanding sign language by converting it to text and speech. It can also help someone with a voice to communicate with those who only understand hand sign language. With the help of the embedded camera, microphone and speakers, the system will capture real-time gestures and convert them to speech and text as well as speech/text to gestures in real-time scenarios. Thus, this software can also be used by someone who wants to learn sign language.

III. System Design

To recognize hand gestures, we are opting for a vision-based technique that does not use any external hardware. A simple hand can do the sign language gestures in front of a realtime camera, and these gestures can be detected and displayed simultaneously on the screen. This technique provides a contact-less way of communication between humans and computers providing growth to the humancomputer interaction (HCI) sector[1]. The following Fig. 1 provides a general flow chart of the system flow

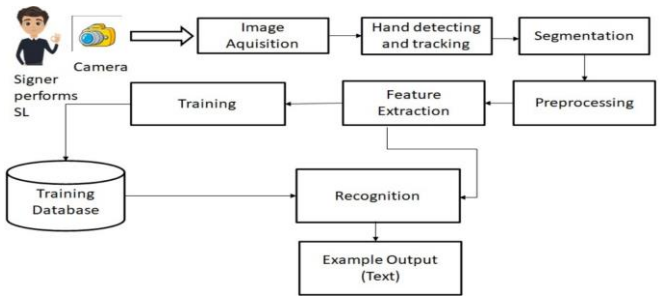

Fig. 1. Fig 1: System Flow Diagram

System Implementation The following section provides the description of the dataset and algorithm configuration that was used for hand sign recognition.

A. Input Data and Training Data

The images were collected with the help of the webcam. Images need to be trained and tested according to the partition made. The hand sign gestures were performed by 3 persons performed in front of the webcam. The hand signs were done using one hand as they would only require only one. Few hand signs were made using hand and finger accessories for the model to perform better in realtime. The recognition process will be less complex and more efficient if the background is less complex, and the contrast effect is comparatively more on the hand. So, it is assumed that the background of the images was less complex and uniform.

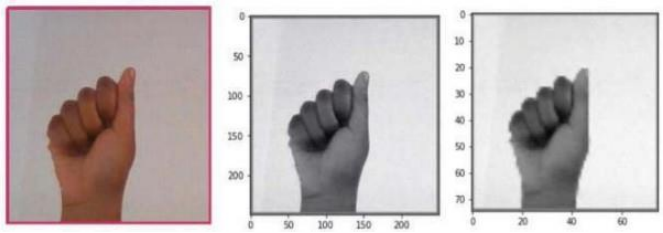

Fig. 2. Fig 2: Stage wise preprocessing of an image

B. Dataset

We are performing hand sign recognition for 26 alphabets, which are a part of ASL. For creating the dataset, we considered around and above 1000 images for each alphabet which sums up to approx of 29,000 images in total. The training and test split is 75class has about 800 images for training and 300 images for testing purposes. So, the total number of images is 21000 for training and 8000 for testing.The images are stored in the files that are named according to their translation. For reading the dataset, each folder is opened simultaneously, and the images are read in it. The labeling of the image is also done at the same time. The images go through the preprocessing stage and are stored in a pickle file along with their labels.

C. Preprocessor

A minimal preprocessing was applied over the dataset to reduce the computational complication and achieve better efficiency and accuracy. The image is initially converted into a grayscale image. The size of all images is made to 75x75 to maintain uniformity between all images. The resizing is done to speed up the process and level the image and avoid any memory errors. The image is then converted into an array of corresponding pixel values.

D. Choosing appropriate algorithm

1) CNN: Convolutional Neural Network or CNN is a type of artificial neural network widely used in computer vision. These networks are composed of several inputs, outputs, and hidden layers. CNN is formed by neurons that have parameters in the form of weights and biases that can be Conventional models for pattern recognition cannot process natural data in raw form. Therefore, the raw images have to pre-process and

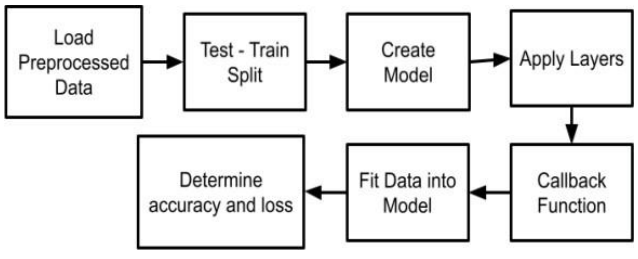

Fig. 3. CNN Flow Diagram

be stored in a particular format. For this project, we stored

the data in a 4-dimensional array. The arrays are appended to a pickle file along with its label. The dataset is loaded using the pickle model. After the dataset is loaded, the train and test split is done. The size of the test data is approximately 30next in this process is fitting the data in the model. The CNN model consists of many convolution layers. The first layer added is Conv2D. A total of 256 layers of size 3x3 are used in the first conv2D layer. It is then followed by ReLu for the activation function. ReLu or Rectified NonLinear is the most successful non-linearity used for CNN. It combats the vanishing gradient problem and is easier to compute and generates sparsity[9]. The next layer added here is MaxPooling2D. The pooling size used here is 2x2. MaxPooling takes the maximum value in the filter range. The next layer added again Conv2D with 64 layers of size 3x3 with the activation ReLu. Once again, the MaxPooling2D layer is added with the same filter range. The last two filters of Conv2D and MaxPooling2D are repeated for a better understanding of the image. The Flatten layer is added next. We flatten the output of the convolutional layers to create a single long feature vector[7].Flatten layer permits the completely connected layers to process the data achieved till this stage. The 1- dimensional array which is produced by the flatten layer is then passed through the dense layer. A Dense layer is where each unit or neuron gets connected to each neuron in the next layer. This layer is implemented with activation as ReLu first and then with Softmax activation. The softmax activation is used to normalize the output and gives a probability distribution. The model is then finally compiled with adam optimizer and computes by the loss with sparse categorical crossentropy. Adam combines the best properties of the AdaGrad and RMSProp algorithms to provide an optimization algorithm that can handle sparse gradients on noisy problems [8]. The following figure shows the summary of the model.

2) Support Vector Machine: Support Vector Machines (SVM) is the supervised learning model and is used in classification, and regression analysis. The clustering algorithm provides a review to the support vector machines is call-back vector clustering and is regularly used in scientific applications as a substitute when data is not labeled. SVM is used for classification and regression. It is explicitly used to find the best separating line. The main goal of SVM is to design an optimal separating hyperplane such that it can classify training vectors. For the training purpose of the model, we load two files X and y using the pickle module. The X file will contain the array with pixels of the image and the y file consists of labels of the array from the X file. Once the dataset is loaded it is divided into training and testing data, where training data consists of 80remaining is used for testing purposes. SVC is the classifier used in SVM with gamma as the parameter of the RBF (Radial basis function kernel). Tunning of the gamma value decides if the model is overfitting, underfitting, or best fitting. In this case, the value of gamma is taken to be 0.001. The classification report and confusion matrix are generated with the help of the metrics module.

E. Displaying the Data

Once the gesture is recognized from the ROI, each gesture is classified into its text form and simultaneously the text is displayed on the screen. The data displayed is the output of the gesture shown on the screen

IV. Training and Testing the Model

1) Training: Training a model for sign language interpretation typically involves using a combination of Convolutional Neural Networks (CNNs) and Support Vector Machines (SVMs). CNNs are adept at extracting meaningful features from images, making them ideal for interpreting sign language gestures captured as image data. SVMs, on the other hand, excel in classification tasks by finding the best possible boundary between different classes.

The process begins by gathering a diverse dataset of sign language gestures, each associated with a specific meaning or word. These gestures are captured as images or video frames. Preprocessing steps, such as resizing, normalization, and potentially augmentation, are applied to enhance the dataset quality and variety.

The CNN is the first stage of the model. It's trained on these images to learn and extract hierarchical features from the gestures. Through multiple convolutional and pooling layers, the CNN identifies patterns, edges, and shapes within the sign language images.

Once the CNN extracts the relevant features, these features are passed to an SVM classifier. The SVM learns to classify these features into distinct categories corresponding to different sign language gestures. It learns to define decision boundaries that separate one sign from another in the feature space. The model is trained iteratively, adjusting its parameters to improve accuracy and reduce errors in classification. Techniques like cross-validation might be used to validate the model's performance and prevent overfitting.

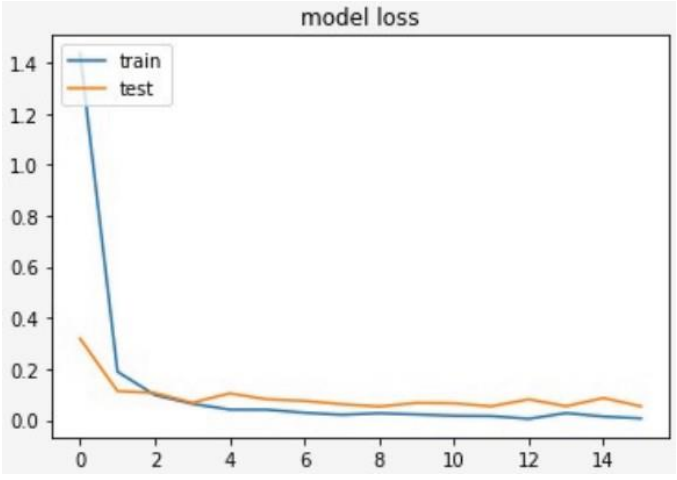

Fig. 4. Training the Model

After sufficient training, the combined CNN-SVM model becomes proficient in recognizing and interpreting sign language gestures. It can accurately predict the meaning of a

given sign by analyzing its visual features, making it a valuable tool for communication between individuals who use sign language and those who do not

2) Testing: Testing the sign language interpretation model involves evaluating its performance on a separate dataset that it hasn't seen before. This dataset contains sign language gestures, similar to the training data, and serves to assess how well the model generalizes to new, unseen examples.

The testing process begins by inputting new sign language images or video frames into the trained model. These unseen samples are processed through the CNN, which extracts relevant features from the images, similar to how it was done during training. The extracted features are then passed to the

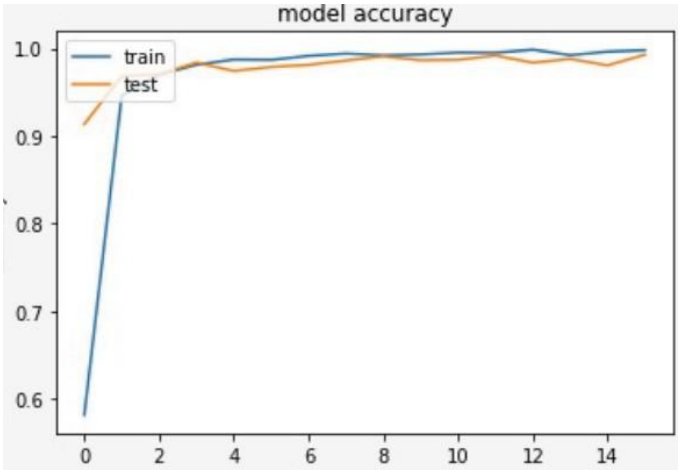

Fig. 5. Testing the Model

SVM classifier, which assigns a predicted label or meaning to each input gesture based on the learned decision boundaries from the training phase.

The model's performance during testing is assessed using various evaluation metrics such as accuracy, precision, recall, and F1-score. These metrics help gauge how well the model correctly identifies and classifies the sign language gestures compared to their actual labels in the test dataset.

Additionally, real-world testing might involve assessing the model's performance in different conditions, such as varying lighting, backgrounds, or hand orientations, to ensure its robustness and generalizability.

If the model performs well on the test dataset, accurately interpreting sign language gestures, it indicates that the model has effectively learned to recognize and classify these gestures and can reliably interpret sign language in real-world scenarios. If there are areas where the model doesn't perform as expected, further refinement or additional training with more diverse data may be necessary to improve its accuracy and robustness.

V. Conclusion

This paper examines which algorithm works best for hand sign recognition. It analyzes the effect of data augmentation on deep learning. In CNN applying the correct combination of layers for the model was to be taken care to obtain well defined nodes in the end. In the KNN algorithm, the value of k should be chosen according to the type of data and amount of noise tolerance the system can take. The parameter k determines the number of nearest neighbours by the majority of the voting process. Similarly in SVM, choosing the correct classifier value which is the gamma parameter should be decided in order to fit the model.The accuracy of each model is best determined by the size of the dataset. SVM works well if there are a large number of features because it is more likely that the data is linearly separable in high dimensional space[13].Whereas if there are less features then KNN works well when the features are less and performs accurately. CNN is also computationally efficient and does not have any dataset size constraint. This enables CNN models to run on any device, making them universally attractive. CNN has the ability to learn distinctive features for each class by itself. We can conclude after that CNN is comparatively the best among all the experimented algorithms used with good accuracy and minimum loss. CNN is a data-driven methodology and data augmentation has a huge effect on deep learning. The system can recognize gestures to a good extent with this algorithm.

VI. Future Scope

This system is an approach to help in communication between the speechless community and another person using hand signs with the use of machine learning. This system helps the communication between the speechless and the blind as well. Any person who is interested in learning sign language can also acquire it by using the speech-tohand sign gesture mode. People can now exchange thoughts, ideas, and messages irrespective of the person's ability to understand sign language or the community they belong to without any trouble.

VII. Refrences

References

[1] Hema B, Sania Anjum, Umme Hani, Vanaja P, Akshatha M "Survey on sign language and gesture recognition system" IRJET -V6-I3 - 2019
[2] Data Flair "Sign language recognition using python and OpenCV"
[3] Math Works "Convolutional neural networks"
[4] J. S. Raikwal, Kanak Saxena "Performance evaluation of SVM and K-Nearest Neighbor Algorithm over medical data set" International Journal of Computer Applications (0975 – 8887) Vol 50 No 14 July 2012
[5] B. Srinivas, G. Sasibhushana Rao "A Hybrid CNN-KNN Model for MRI brain Tumor Classification" International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-2, July 2019
[6] Jordi Torres.AI "Convolutional neural network for beginners" Towards Data Science 2018.
[7] Wikipedia "American sign language"
[8] Britannica "American sign language" Eric Drasgow
[9] Analytics Vidhya "Understanding support vector machine from examples" Sunil Ray 2017
[10] Analytics Vidhya "How to choose the value of K in KNN algorithm" 2015