A Hybrid Model Proposal for Hand Sign Recognition

Syed Afreen Akther
20201CAI0061 dept of Computer Science
Presidency university
Itgalapura, Bangalore, Karnataka
SYED.20201CAI0061@presidencyuniversity

Dr. Madhura K
Assistant professor: dept of Computer Science
Presidency university
Itgalapura, Bangalore, Karnataka
madhura@presidencyuniversity

ABSTRACT

A hybrid Model for hand gesture recognition in sign language is an exigent task that involves the strengths of multiple machine learning and artificial intelligence methods to achieve improved accuracy and robustness. This paper contours the steps involved in designing a hybrid model, including data collection, feature extraction, model selection, model combination, training and evaluation, and deployment. The hand gesture recognition model will be trained on a large data set of hand gestures and it will be evaluated using metrics such as precision, recall, and F1 score. The model combines the output of many machine learning models and algorithms such as decision trees, random forest, support vector machines, and deep neural networks, to increase their strengths and minimize their weaknesses. The final model is implemented on the system or application for sign language interpretation, providing a valuable tool for the hard of hearing and the deaf community.

I. INTRODUCTION

Approximately 430 million people globally, or 20% of the world's population, have a hearing loss that is considered "disabling," meaning a loss greater than 35 decibels in their better ear. A large majority of these individuals, approximately 80%, reside in low and middle-income countries. The risk of hearing loss increases with age, and among those over 60 years old, over a quarter are affected by disabling hearing loss.

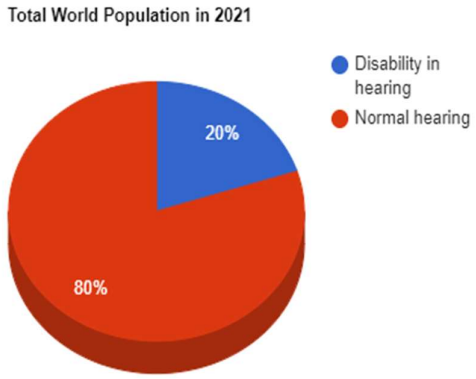

Figure 1: Representation of statistics

At crucial life stages such as pregnancy, the maternal period, adolescence and early adulthood, and the elder age, hearing loss and deafness can develop.

So visual language is used by deaf and hard-of-hearing individuals to communicate with each other and with those who can hear is called sign-language. It consists of hand gestures, facial expressions, and body language that convey meaning and grammar. A two-way communication process, sign language requires both the capacity to comprehend the gestures being made (responsive abilities) as well as the capacity to make the signs oneself (expressionistic abilities) Translation and recognition of sign language is an important research area because the integration of hearing-impaired people into the general public and the provision of equality of opportunity for the entire community make this a crucial research field. The creation of human-machine interfaces with the potential to improve communication between hearing-impaired and healthy people is a critically important issue aimed at eliminating the third human element. (Reference Sign Language Translation Using Deep Convolutional Neural Networks)

The advancements in technology have had a profound impact on the way humans interact with each other. some of the key ways it has advanced include is increased Connectivity, improved Accessibility, Enhanced Communication, Virtual and Augmented Reality, and Increased personalization.

In this paper, we are going to implement with the help of the Hybrid model. A hybrid model in machine learning is a combination of two or more models that work together to address the limitations of any single model. The idea behind hybrid models is to merge the strengths of different models to achieve improved performance, accuracy, and robustness. This concept is particularly useful in complex prediction tasks where a single model may not be enough to capture all the intricacies of the problem.

II. LITERATURE SURVEY

"Sign Language Translation using Deep Convolutional Neural Networks" (Rahib.H. Abiyey, Murat Arslan and Jhon Bhush Idoko), in their paper, they have suggested that American Sign Language Translator creates a vision-based system by combing SSD, Inception, and SVM. Cross-validation was used to train the hybrid model and a Monte Carlo estimator was used to validate it. The findings showed that cross-validation had greater accuracy. The system, which combines object detection feature extraction, and classification within a deep learning network, employs the ASL fingerspelling data set and has a straightforward structure. It has an average accuracy rate of 99% and an RMSE of 0.0126.

"Gesture-Based Real-Time Indian Sign Language Interpreter" (Akshay Divkar, Rushikesh Bailkar, Dr Chhaya S. Pawar) in order to interpret isolated hand gestures from Indian sign language, the study suggests a vision-based system that uses CNN and RNN using the self-created data set of Indian sign language, it attained an accuracy of 54.2% when classifying spatial and temporal features using a pool layer technique. Future work will focus on enhancing continuous gesture recognition, merging CNN and RNN into a single model, and investigating attention-based approaches to make greater use of temporal dependencies

"Wearable on-device deep learning system for hand gesture recognition based on FPGA accelerator" (weibin Jiang1, Xuelin Ye2, Ruiqi Chen1,3, Feng Su3,4, Mengry Lin1, Yuhanxiao Ma5,
Yanxiang Zhu3 and Shizhen Huang1), in this paper, they have presented a low-cost wearable device with a neural network accelerator for gesture recognition. The device uses an IMU sensor and a software-hardware co-design to perform gesture recognition with low power and low latency. A new activation function was designed to improve recognition accuracy. The data is open-sourced for future use and the framework presents four limitations: a limited number of gestures tested, size, power consumption and design verification.

"A Comparative Review on Application of Different Sensors for Sign Language Recognition" (Muhammad Saad Amin, Syed Tahir Hussain Rizvi, and Md. Murad Hossain), The paper examines the various sensors utilized in Sign Language Recognition (SLR) technology with a specific emphasis on the glove sensor approach. It delves into the advantages, difficulties, and suggestions related to SLR, while also surveying the research of other experts in the field. The authors analyse glove types, data-capturing sensors, recognition methods, datasets, processing units, and output devices. The aim is to create a sophisticated SL-speech and text translation system to enhance communication.

"Recurrent Convolutional Neural Networks for Continuous Sign Language Recognition by Staged optimization" (Runpeng Cui, Hu Liu, Changshui Zhang) The paper proposes Recurrent convolution neural networks are used in deep architecture to continuously recognise sign language. The authors exploited the representational ability of the RCNN with tuning on significant gloss-level portions as part of a graduated optimisation procedure. They also create a unique detection net to guarantee uniformity among consecutive predictions.

"Perspective and Evolution of Gesture Recognition for Sign Language: A Review" (Jesus Galvan-Ruiz, Carlos M. Travieso-Gonzalez, Acaymo Tejera-Fettmilch, Alejandra Pinan-Roescher, Luis Esteban-Hernandez, and Luis Dominguez-Quintana)

To summarize, the sign language recognition technology field has seen substantial growth over the years and continues to evolve. Although much progress has been made, there remains room for improvement. Among the available devices, the Leap Motion stands out for its affordability, ease of use, compatibility with multiple platforms, and absence of active elements. Additionally, it comes with user-friendly software for processing information and has a large enough interaction area to accurately recognize sign language.

"Bangla Sign Digits: A Dataset for Real-Time Hand Gesture Recognition" (Dardina Tasmere, Boshir Ahmed, Md Mehedi Hasan) in the paper they have proposed a real-time hand gesture recognition system the methodology involves four convolution layers together with four pooling layers including four fully connected layers and a dropout layer with 40% dropout. The accuracy of the system was 97.63%.

"Detecting Hand-Palm Orientation and Hand Shapes For Sign Language Gesture Recognition Using 3D Images' (Lalit K. Phadtare, Raja S. Kushalnagar, Nathan D. Cahill) the paper proposed a system that implements a method that successfully extracts features to identify hand palm orientation and hand shape which are the two most important parameters of production out of five in ASL. The accurate results of palm orientation detection further were improved with shape sampling sizes and the number of bins.

III. Existing Methodology

There are numerous algorithms which we have been implementing in past years with an accuracy of 60-80%. The most used hand sign gesture recognition algorithm is K-Nearest neighbours(knn). Whereas this algorithm is a simple classification algorithm that works by finding the K- nearest training examples to the given data set and assigning the test example to the class that is most common among its nearest neighbours.

The accuracy of any model depends upon the specific hand signs that are being used and recognized based on the quality, variations, and quantity of the training data set. The knn shows the accuracy of all-around 70-80% with appropriate feature extraction and tuning of the hyperparameters.

Another algorithm with 60-70% accuracy is decision trees which are a type of machine learning model that works by recursively portioning the input space based on features of the input data.

A. Steps involved in developing the model:

Data Collection: This is the first step involved in the classification model. The Process of gathering raw data from various sources that are used for analysis, research, and other purposes. This is the crucial step in building the model.

Data Pre-processing: This involves cleaning and preparing the data by removing missing or duplicate data handling outliers, scaling, normalizing, and transforming the data for use in a model.

Feature extraction: Identifying the relevant information and extracting the most from the raw data to facilitate the subsequent analysis. Widely used in computer vision, feature extraction can be used to identify specific patterns, and objects in images such as edges, corners, or textures.

Model Training: The process of developing and refining a model to generate precise predictions or judgements based on incoming information. It entails giving training sets to a model or algorithm and revising the model's inputs until it can effectively predict outcomes from current data. This involves various techniques such as gradient descent and random forest.

Model Evaluation: After the model has been trained, the testing set is used to evaluate it. The performance of the model is analysed using evaluation measures like accuracy, precision, Confusion matrix recall and F1 score. It can generalise clean, unstudied data and spot any potential problems like over-fitting.

Model refinement: It is the process of improving a trained model's performance by changes to its hyperparameters, architecture, or training data is known as model refining.

The model is refined in many ways:

Hyperparameter tuning, Architecture modification, regularization Data augmentation, Transfer learning.

IV. Proposed Methodology

CNN was created expressly to handle image identification and computer vision tasks this is frequently the best model option for these tasks. The accuracy of a CNN algorithm for hand sign recognition will depend on various factors such as the size and quality of the dataset used for training the complexity of the model architecture, and the parameters used during training. A CNN model with a large and diverse dataset can achieve high accuracy in hand sign gesture recognition.

B. Benefits of CNN:

The ability to automatically learn and extract features from input photos using a sequence of convolutional layers, CNN are meant to automatically learn and extract features from input images. This means that the model will automatically learn the features it should search for in the input data where we do not have to manually specify them.

The ability to handle complex data and big datasets is a feature of CNNS that is frequently required for image recognition and computer vision jobs. Considering the model can recognize patterns and connections among the input data, allowing it to generalise successfully to fresh, unexplored material CNNs are flexible and customizable, allowing them to be used for many image identification and computer vision tasks. To increase productivity on a particular assignment, for instance, we can also add or delete layers, modify the size and the number of filters and alter the activation mechanism.

CNNs adapted and customized for specific image recognition and computer vision tasks also have the flexibility

of adding, deleting layers and changing the activation function accordingly to improve on a specific task. Good performance on image recognition tasks: CNNs have been shown to perform very well on image recognition tasks, achieving state-of-the-art results on benchmarks such as ImageNet. This is due to their ability to learn and extract relevant features from input images, as well as their ability to handle the large and complex datasets often encountered in these tasks.

Flexibility and adaptability: CNNs can be adapted and customized for specific image recognition and computer vision tasks. For example, you can add or remove layers, adjust the size and number of filters, and change the activation functions to improve performance on a specific task.

Overall, CNNs are a powerful and versatile tool for image recognition and computer vision tasks, and they have been used to achieve significant breakthroughs in areas such as object detection, facial recognition, and autonomous driving.

Convolutional Neural Networks (CNNs): these are frequently employed for a variety of image identification applications, including hand gesture recognition. They are especially good at spotting trends and details in picture and video analysis

C. **The basic working of a CNN can be broken down into these steps:**

- Convolutional Layer: The input image is placed through several convolutional layers, each of which applies a set of filters which can be referred to as kernels to the image. Every filter searches the image, looking for specific traits or patterns that are crucial for the job at hand. Each filter produces a feature map that emphasizes the presence of the pattern in the input as its output.
- ReLU Layer: This is an activation function which sets all negative values in the feature map to zero and leaves positive values unchanged. The output of the convolutional layer is passed through this function.
- Pooling Layer: By bypassing the output of the real layer through this layer, the spatial size of the feature map is reduced. This is accomplished by computing the maximum or average values of the pixels contained within a tiny window with this method the model's parameter count are decreased and overfitting is prevented.
- Fully connected layer: one or more layers that are completely linked are applied after the pooling layer's output is flattened into a 1d vector, these layers compute the model's final output is comparable to those found in conventional neural networks.
- SoftMax layer: It is the last layer of the CNN, and normalizes the model's output into a probability distribution over all conceivable classes. The output that the model predicts will be the class with the highest probability.

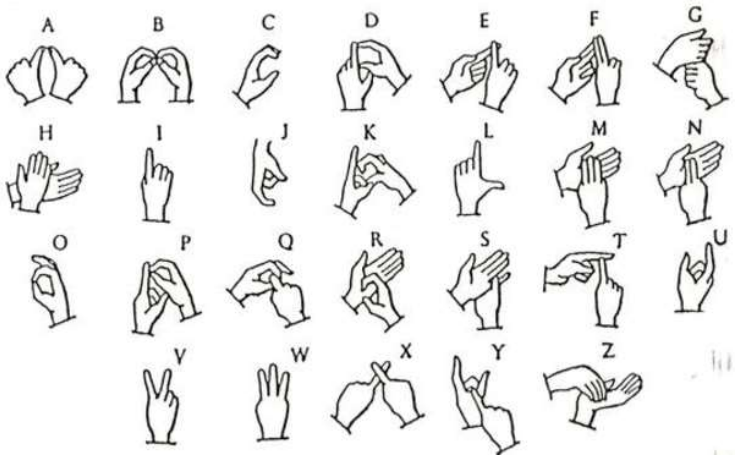

Figure 2: Indian Hand Sign language

https://d3i71xaburhd42.cloudfront.net/48d9c25a01bcffa54507fa99fb9159972cd7032c/2-Figure1-1.png

D. **Implementation of the proposed methodology:**

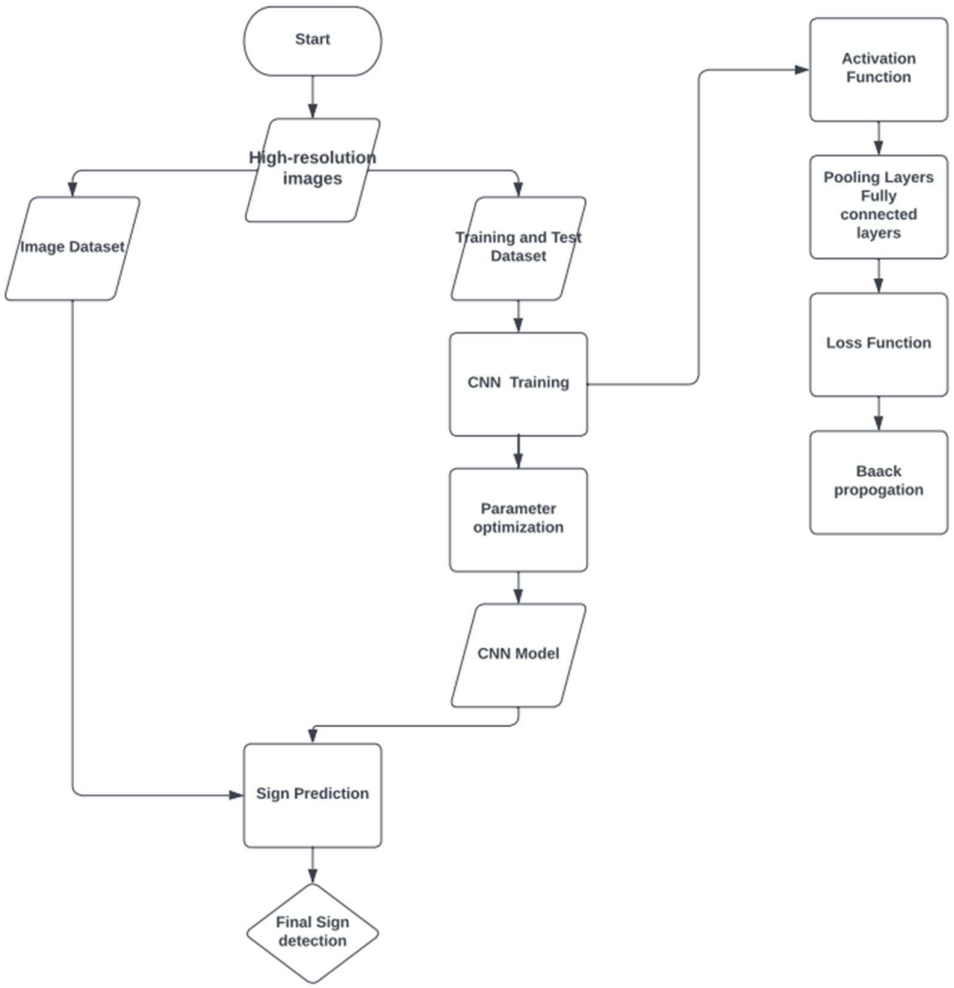

Figure 3: Flow chart for the complete model

E. **Pseudo Code for the Algorithm**

Step 1: Collect and pre-process data

X,y = load_data() # load hand sign gesture images and labels

X = pre_process(X) #resize and normalize the images

Step 2: Split data into training and testing sets

X_trian, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2)

Step 3: Build and compile the model

Model = Sequential()

Model.add(Conv2d(filters=32,kernel_size= (3,3),activation='relu', input_shape=(X_train.shape[1:])))

Model.add(MaxPooling2d(pool_size=(2,2))

Model.add(Flatten())

Model.add(Dense(units=128, activation='relu'))

Model.add(Dense(units=len(classes), activation='softmax'))

Model.compile(loss='categorical_crossentropy' optimizer='admin',metrics=[accuracy])

Step 4: Train the model

Model.fit(X_train,y_train,epochs=10, batch_size=32)

Step 5: Evaluate the model

Loss, accuracy = model.evaluate(X_test, y_test)

Print('Accuracy: {:.2f}%.format(accuracy*100))

Step 6: Make predictions

New_image = load_new_image()

New_image = pre_process(new_image)

Prediction = model.predict(new_image)

Factors for better efficiency

Increase the size and diversity of the dataset: The greater amount of data gives the model to learn and recognize various hand signs accurately.

A. **Improving the quality of data sets**:

Ensuring that the data is clean, labelled correctly, and balanced to prevent bias in the model. Also, the implementation of argumentation techniques creates variations of the existing data to improve the model's robustness.

Optimize the model architecture: Experiment with different CNN architectures and hyperparameters to find the optimal model that can learn and generalize well to new data. We can also use pre-trained models and fine-tune them.

Regularization techniques: regularization techniques like dropout, L1 and L2 regularization, and batch normalization can help prevent overfitting and improve the model's accuracy.

Improving the training process: Techniques like learning rate schedule, early stopping, and gradient clipping to improve the training process and ensure that the model converges to the optimal solution.

Considering the target environment: Factors like Lighting conditions, background noise, and hand positioning can affect the model's accuracy. So, it is significant to consider them during the training and evaluation process.

RESULTS

The accuracy of the model depends on many factors, evaluated on a test set which is a subset of the data that was not used during training. The number of correctly classified instances in the test is the accuracy. We can achieve more than 90% on the benchmark datasets for image classification.

CONCLUSION

A potential method for precisely recognising and categorizing hand signs in real time is the hybrid model using Convolutional Neural Network (CNN). This model can efficiently collect the spatial and temporal information of hand signs, enabling it to identify them with high accuracy. This is done by leveraging the strengths of both classic machine learning approaches and deep learning algorithms.

And, the performance of the model can be further enhanced by utilising larger datasets and adjusting hyperparameters. The model can also be optimised to identify sign language with greater complexity.

REFERENCES

1. Koreascience.or.kr
2. "Detecting hand-palm orientation and hand shapes for sign language gesture recognition using 3D images".
3. "Hand gesture recognition for Bangla Sign Language Using Deep Convolution Conference Neural Network".
4. "A Comparative Review on Application of Different Sensors for Sign Language Recognition" Muhammad Saad Amin, Syed Tahir Hussain Rizvi, and Md. Murad Hossain
5. "Perspective and Evolution of Gesture Recognition for Sign Language: A Review" **J**esus Galvan-Ruiz, Carlos M. Travieso-Gonzalez, Acaymo Tejera-Fettmilch, Alejandra Pinan-Roescher, Luis Esteban-Hernandez, and Luis Dominguez-Quintana
6. "Detecting Hand-Palm Orientation and Hand Shapes For Sign Language Gesture Recognition Using 3D Images" Lalit K. Phadtare, Raja S.Kushalnagar, Nathan D. Cahill

7. “Wearable on-device deep learning system for hand gesture recognition based on FPGA accelerator” weibin Jiang1, Xuelin Ye2, Ruiqi Chen1,3, Feng Su3,4, Mengry Lin1, Yuhanxiao Ma5,Yanxiang7 Zhu3 and Shizhen Huang1
8. “Sign Language Translation using Deep Convolutional Neural Networks” Rahib.H. Abiyey, Murat Arslan and Jhon Bhush Idoko.