RESEARCH ARTICLE OPEN ACCESS

A Comparative Study on Indian Sign Language Recognition Techniques

[1]Shalaka Gaikwad
[1]Research Scholar, Department of Computer Science, Taywade College, Koradi, Nagpur, India

[2] Dr. Girish Katkar
[2]Assistant Professor, Department of Computer Science, Taywade College, Koradi, Nagpur, India

[3] Dr. Ajay Ramteke
[3]Assistant Professor, Department of Computer Science, Taywade College, Koradi, Nagpur, India

ABSTRACT:-

Sign language is a visual-gestural language used by deaf and hard-hearing people for communication. Sign language has emerged as the primary mode of communication for people with deaf and voice impairment disabilities. To overcome the problem faced by these community we have studied some ISL recognition techniques. We have studies various techniques of Indian Sign Language Recognition and from these we have compare convolution neural network and Knearest neighbor algorithm. Accuracy of KNN was 99.9872% and 90% of CNN. From this we have evaluated that KNN is best for ISL recognition as compare to CNN.

Keywords:— k-NN, CNN, double handed Indian Sign language, English alphabets.

I. INTRODUCTION

Sign language is a visual-gestural language used by deaf and hard-hearing people for communication. Deafness and voice impairment have been persistent disabilities throughout history, hindering individuals from engaging in verbal communication and leading to their isolation from the predominantly vocally communicating society. Sign language has emerged as the primary mode of communication for people with these disabilities. Hand gesture recognition, a crucial aspect of human-computer interaction, has garnered significant research attention. Two primary approaches, contact-based and vision based, differentiate the acquisition techniques of input data. While the former involves interfacing devices like motion sensors and data gloves, the latter leverages imaging devices such as cameras, eliminating the need for physical contact and discomfort to the signer [1]. Sample frames from hand gesture datasets. (source) Numerous methods employ artificial intelligence techniques for recognition. Neural networks, time-delay neural networks, and block-based histogram representations have been proposed. Techniques such as hidden Markov models, local binary pattern in spatiotemporal representation, and dynamic sign language recognition systems using bag-of-features have demonstrated varying recognition accuracies [2]. The application of sophisticated models, such as 3D convolutional neural networks (3DCNN), has shown promise in the spatiotemporal feature learning process. Despite extensive research, challenges persist. Some methods require specific attire or coloured gloves, limiting practicality [3]. Some are restricted in their operations, recognizing just a small set of motions, or depending on intricate technology. To get around these restrictions, vision-based techniques—discussed in this paper—offer signer-independent modes and do away with the requirement for specialist equipment. These approaches, employing 3DCNN architectures and exploring feature fusion techniques, contribute to enhancing the usability and accessibility of hand gesture recognition technology in diverse applications [4].

II. RELATED WORK

In this paper authors focuses on recognizing American Sign Language (ASL) letters and numbers, addressing the evolving technology landscape and the growing demand for improved user experiences among those primarily using sign language for communication. Leveraging deep learning, particularly through transfer learning, this study aims to enhance ASL recognition technology. Various deep learning models, including VGG16, ResNet50, MobileNetV2, InceptionV3, and CNN, are evaluated using an ASL dataset sourced from the Modified National Institute of Standards and Technology (MNIST) database, featuring ASL alphabetic letters represented through hand gestures [5]. ISL uses a combination of one-handed and two-handed gestures, which makes it fundamentally different from other common sign languages like American Sign Language (ASL). This paper aims to address the communication gap between especially abled (deaf) people who can only express themselves through the Indian sign language and those who do not understand it, thereby improving accessibility and communication for sign language users. This is achieved by using and implementing Convolutional Neural Networks on our self-made dataset. This is a necessary step, as none of the existing datasets fulfils the need for real-world images [6]. In this paper they used vision -based approach, direct pixel value local histogram, and hierarchical centroid features were extracted from an input image. k-NN, Neural Network classifier, Naïve Bayes algorithm are used as classifier [7]. Adithya V, Vinod P. R, Usha Gopalakrishnan, presented in their work, Artificial Neural Network Based Method for Indian Sign Language Recognition. For segmentation RGB colour spaced are transformed into YCbCr color space, the pixel of skin colour in the input images are identified by applying a thresholding technique based on distribution of the skin colour in YCbCr colour space. The result of segmentation produces a binary image in which the skin pixels are white in colour and background in black colour. For feature extraction distance transformation, row and column projection applied on distance transformed image, Fourier descriptor is applied on row and column projected image [8]. In this paper the problem with two deep learning (DL) techniques, namely Convolutional Neural Networks (CNN) and stacked denoising autoencoder (SDAE) networks, to recognize 24 ASL alphabets. Their data source was the gesture recognition database curated by Thomas Moeslund [9]. Automatic Indian Sign Language Recognition for Continuous Video Sequence- Joyeeta Singha, Karen Das This paper describes a novel approach towards a system to recognize the different alphabets of Indian Sign Language in video sequence automatically. The proposed system comprises of four major modules: Data Acquisition, Preprocessing, Feature-Extraction and Classification. Preprocessing stage involves Skin Filtering and histogram matching after which Eigen vector-based Feature Extraction and Eigen value weighted Euclidean distance-based classification technique was used. 24 different alphabets were considered [10]. Dynamic Hand Gesture Recognition Using the Skeleton of the Hand- Bogdan Ionescu, Didier Coquin, Patrick Lambert, Vasile Buzuloiu. Hand gestures can be divided into two main categories: static gestures and dynamic gestures. In this paper, a novel dynamic hand gesture recognition technique is proposed [11]. In this paper author Ashok Kumar Sahoo proposed a system for automatic recognition of Indian Sign Language immobile numeric signs. Wearable devices were not used to capture images it was collected by digital camera. System was created for real time recognition of sign for this digit's database of ISL is created which contains 5000 images and 500 images for every digit (0-9). Hierarchical centroid feature vector technique is used for feature extraction, Naïve Bayes & k-NN used for classification, recognition rate of kNN is high as compared to Naïve bayes classifier [12]. This paper proposed a system which consist of training, testing and recognition phase. First step is pre-processing. Preprocessing involves segmentation and filtering; segmentation is performed using Global thresholding algorithm and median filter is used to remove noise from the image. Shape descriptors define the boundary of an image while the invariant moments are used to change in scale and position of an image. Multi-class Support Vector Machine classifier is used to appropriately classify the images [13].

III. DATA SET

We have taken double handed Indian sign language data set. We have created our own data set and we split the data into training set and training for classification.

IV. METHODOLOGY

A. CNN- We have created a deep learning architecture with the pretrained parameters. We can create our own CNN model by two means first we write a code in MATLAB script directly. And second step is to use built in GUI provided by MATLAB. The size of input image is

150*150*3 here 3 is for rgb band. First, we create a blank network then create image input layer. Next layer is convolution 2d layer followed by rectified linear unit and followed by pooling here we select max pooling then we add fully connected layer before it there is a flattened layer but in this case of images and especially if we are working in MATLAB this flattened layer is not require if we will work with video images then we flattened layer for sequence images. Then we add SoftMax Layer followed by classification layer. These layers are used for feature extraction. We connect all these layers after that first thing is to analyze the feature space and set the dimension according to our need then we export the network. Here data is splinted into two parts for training and testing after it features are extracted and model was trained with these properties Gradient Decay Factor, Squared Gradient Decay Factor, Epsilon, Initial Learn Rate, Learn Rate Schedule, Learn Rate Drop Factor, Learn Rate Drop Period, L2Regularization, Gradient Threshold Method, Gradient Threshold, Max Epochs, Mini Batch Size, Verbose, Verbose Frequency which are given below. Fig 1. Shows layers in the created network. Fig 2. Represent image of alphabet C and many such images are used for training the model and fig 3. is used to test the model and fig 4. shows the result. Table 1. represents during first iteration elapsed time is 11 seconds, mini-batch Accuracy is 9.09%, mini batch Loss is 13.8758, and Base Learning Rate is 0.0010. During 20 iteration elapsed time is 13 seconds, mini-batch Accuracy is 90.91%, mini-batch Loss is 1.4493, and Base Learning Rate is 8.0000e 06 which increases model accuracy.

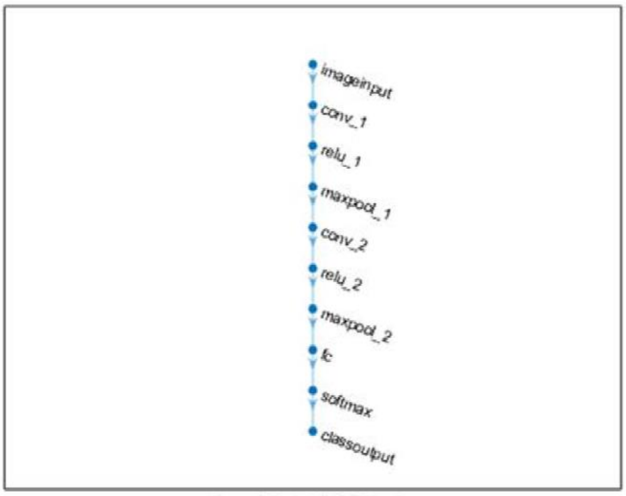

Figure 1: Created CNN Network

Figure 2: Training image

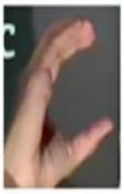

Figure 3: Test image

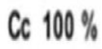

Figure 4: Recognized character

Epoch	Iteration	Time Elapsed (hh:mm:ss)	Mini-batch Accuracy	Mini-batch Loss	Base Learning Rate
1	1	00:00:11	9.09%	13.8758	0.0010
20	20	00:00:13	90.91%	1.4493	8.0000e-06

Table 1. Initializing input data normalization.

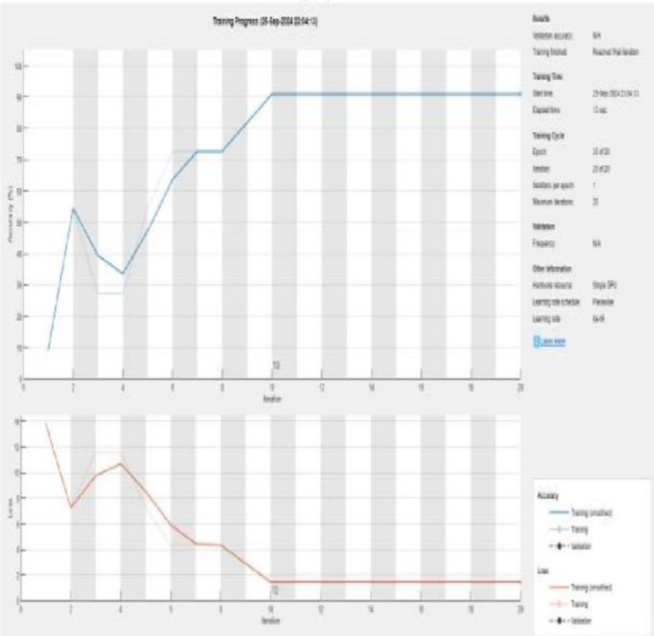

Figure 5: Training progress using CNN

B. KNN- This algorithm is designed for a sign language detection task which uses machine learning approach. Step 1: Load Images dataset. Image Datastore function is used loads images from the specified directory and it also includes subfolders and uses folder names as labels. The function disp(imds) displays information about the datastore, including the number of images and their labels. Step 2: Initialize Variables Then initializes empty arrays to store the extracted features and corresponding labels for each image. Step 3: Process Each Iterates each image in the datastore, then read the current image, resize it to 64*64 pixels for uniformity. Next step is to convert the RGB image to grayscale to simplify feature extraction. From the grayscale image extract histogram of oriented gradients (HOG)features. Then add the feature vector and its corresponding label to respective arrays. Step 4: Train a Classifier We have train a k-Nearesr Neighbors (k-NN) classifier using the extracted features and labels, with five neighbors considered for classification. Step 5: Predict on a New Image uigetfile is used to open a dialog box for the user to select a new image file for prediction. Step 6: Check User Input It checks if the user cancelled the file selection. If not, it constructs the file path, reads the selected image, and optionally displays it. Step 7: Prepare the Test Image Then resizes and converts the selected test image to grayscale, and then extracts its HOG features, just like for the training images. Step 8: Make Prediction Uses the trained classifier to predict the label of the new image based on the extracted features. Displays the predicted label for the test image.

We have created a dataset from Double-Handed Indian Sign Language Characters image and preprocess data extract features from both the training and testing data set and used Convolution Neural Network algorithm for classification of data. In our experiment both training data set and test data gives good accuracy. Figure 5. shows training progress of our model with 90% accuracy and loss is 1.8%. The accuracy of the model k-nearest neighbor algorithm for classification of data was 99.9872%. We will study this problem in depth and will apply deep learning toolbox for recognition which will be helpful for deaf and dumb community and normal community which will remove communication barrier between two community which also increase recognition rate.

V. CONCLUSION

We have studies various techniques of Indian Sign Language Recognition and from these we have compare convolution neural network and K-nearest neighbor algorithm. Accuracy of KNN was 99.9872% and 90% of CNN. From this we have evaluated that KNN is best for ISL recognition as compare to CNN.

ACKNOWLEDGMENTS

This research is supported by Mahatma Jyotiba Phule Research Fellowship an Autonomous Institute of The Other Backward Class Bahujan Welfare Department, Govt. of Maharashtra.

REFERENCES

[1] M. Al-Qurishi, T. Khalid and R. Souissi, "Deep Learning for Sign Language Recognition: Current Techniques, Benchmarks, and Open Issues," in

IEEE Access, vol. 9, pp. 126917-126951, 2021, doi: 10.1109/ACCESS.2021.3110912.

[2] S. Yap, B. N. Panggiri, G. Darian, Y. Muliono and S. Y. Prasetyo, "Enhancing BISINDO Recognition Accuracy Through Comparative Analysis of Three CNN Architecture Models," 2023 International Conference on Information Management and Technology (ICIMTech), Malang, Indonesia, 2023, pp. 732-737, doi: 10.1109/ICIMTech59029.2023.10277780.

[3] D. Naveen, M. D Nirmala, T. J. T. Maladhkar, M. Serena and R. Mohis, "TechIt-Easy: An Application for Physically Impaired People Using Deep Learning," 2023 International Conference on Artificial Intelligence and Applications (ICAIA) Alliance Technology Conference (ATCON1), Bangalore, India, 2023, pp. 16, doi: 10.1109/ICAIA57370.2023.10169396.

[4] Sahoo, Ashok & Mishra, Gouri & Ravulakollu, Kiran. (2014). Sign language recognition: State of the art. ARPN Journal of Engineering and Applied Sciences. 9. 116- 134.

[5] Sanaa Mohsin, Baraa Wasfi Salim, Ashty Kamal Mohamedsaeed, Banar Fareed Ibrahim and Subhi R. M. Zeebaree, "American Sign Language Recognition Based on Transfer Learning Algorithms", International Journal of Intelligent Systems And Applications In Engineering, pp.390-399, 2024.

[6] Harsh Kumar Vashisth, Tuhin Tarafder, Rehan Aziz, Mamta Arora and Alpana, "Hand Gesture Recognition in Indian Sign Language Using Deep Learning", Engineering Proceedings, 2023.

[7] A. K. Sahoo1, P. K. Sarangi, P. Goyal, "Indian Sign Language Recognition Using Soft Computing Techniques", Machine Vision Inspection Systems, vol.1, ch2, 2020.

[8] Adithya, V., P. R. Vinod, and Usha Gopalakrishnan. "Artificial neural network-based method for Indian sign language recognition" IEEE Conference on Information & Communication Technologies (ICT), IEEE, 2013.

[9] Aly, Walaa & Aly, Saleh & Almotairi, Sultan, "User-Independent American Sign Language Alphabet Recognition Based on Depth Image and PCANet Features", IEEE Access. PP. 1-1. 10.1109/ACCESS.2019.2938829, 2019.

[10] M. M. Hasan, P. K. Misra, "Brightness Factor Matching for Gesture Recognition System Using Scaled Normalization", International Journal of Computer Science Information Technology (IJCSIT), Vol. 3.

[11] J. Singha and K. Das, "Indian Sign Language Recognition Using Eigen Value Weighted Euclidean Distance Based Classification Technique", International Journal of Advanced Computer Science and Applications (IJACSA), vol. 4.

[12] Ashok Kumar Sahoo, "Indian Sign Language Recognition Using Machine Learning Techniques", Macromolecular Symposia, pp.397, 2021.

[13] Karishma Dixit and Anand Singh Jalal, "Automatic Indian Sign Language Recognition System," 2013 3rd IEEE International Advance Computing Conference (IACC).