Hand Sign Recognition Using Machine Learning

This project report is submitted to

Yeshwantrao Chavan College of Engineering

(An Autonomous Institution Affiliated to Rashtrasant Tukdoji Maharaj Nagpur University)

In partial fulfilment of the requirement

For the award of the degree

Of

Bachelor Of Engineering in Computer Technology

By

Nikhil Dapkekar

Piyush Dabare

Ayush Raut

Jay Rathod

Under the guidance of

Prof. Pradnya Moon

Co- guide

Mr. Tanay Katkamwar

DEPARTMENT OF COMPUTER TECHNOLOGY

Nagar Yuwak Shikshan Sanstha's

YESHWANTRAO CHAVAN COLLEGE Of ENGINEERING

(An autonomous institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur)

NAGPUR – 441 110

2022-23

CERTIFICATE OF APPROVAL

Certified that the project report entitled Hand Sign Recognition Using Machine Learning has been successfully completed by Piyush Dabare, Ayush Raut, Nikhil Dapkekar, Jay Rathod under the guidance of Prof. Pradnya Moon in recognition to the partial fulfilment for the award of the degree of Bachelor of Technology in Computer Technology, **Yeshwantrao Chavan College of Engineering (An Autonomous Institution Affiliated to Rashtrasant Tukdoji Maharaj Nagpur University)**

Signature	Signature	Signature	Signature
Prof. P. S. Moon	**Mr. Tanay Katkamwar**	**Prof. S. Kapse**	**Dr. Rakhi Wajgi**
(Guide)	**(Co-Guide)**	**(Coordinator)**	**(HoD)**

Signature of

External:

Examiner Name:

Date of Examination:

DECLARATION

We certify that

a) The work contained in this project has been done by us under the guidance of my supervisor(s).
b) The work has not been submitted to any other Institute for any degree or diploma.
c) We have followed the guidelines provided by the Institute in preparing the project report.
d) We have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.
e) Whenever we have used materials (data, theoretical analysis, figures, and text) from other sources, we have given due credit to them by citing them in the text of the report and giving their details in the references. Further, we have taken permission from the copyright owners of the sources, whenever necessary.

Name of the Student	Signature of Student
1. Piyush Dabare	
2. Ayush Raut	
3. Nikhil Dapkekar	
4. Jay Rathod	

Acknowledgement

We would like to thank our guide Prof. Pradnya Moon for thorough guidance in the project. We are extremely grateful and indebted to them for their expert, sincere, valuable guidance and encouragement which was of immense help to us.

We would like to express our sincere gratitude to Dr. R. D. Wajgi, Head, Department of Computer Technology for her constant encouragement towards the successful completion of our work

We wish to express our sincere thanks to Dr. U.P. Waghe, the Principal of our college for providing us with all the necessary facilities and the infrastructure without which we would not have been able to complete our project successfully

We would also like to thank our Project Coordinator Prof. S. R. Kapse for their continuous guidance owing to which the project could take shape.

We would like to thank technical assistant Mrs. B. H. Kulkarni for providing the necessary technological support. Last, but the least, we would like to thank all the faculty members and non-teaching staff members who helped us despite their busy schedule.

Chapter Index

Contents **Page No.**

Figures Index

Table Index

CO-PO/PSO Articulation Matrix

CO's	Statement	PO's												PSO's	
		1	2	3	4	5	6	7	8	9	10	11	12	PSO 1	PSO 2
1	Acquire the domain knowledge and analyse the implemented model	3	3	3	3	3		3				2	3		2
2	Design and develop the solution using appropriate tools and techniques for betterment of society and industry	3		3	3	3	3	3	3	3	3	2	3	3	3
3	Communicate the work done through paper presentation or participation in competition as a team.						3				3	2	3		

ABSTRACT

Sign language is one of the oldest and most natural form of language for communication, but 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 fingerspelling based American sign language. In our 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.

There is a communication problem between the deaf community and the hearing majority which can be solved by using innovation ideas and techniques which can be used to automatically recognise sign language and convert it into text. Sign Language is used by the deaf and dumb community to communicate with others, but the most common problem here is that people around them cannot understand sign language. Written communication can be also be used but it is slow as compared to face-to-face communication. In case emergency situations its more efficient to use sign language than written communication therefore, many sign languages were created in different countries of world. To solve this problem, we can use a custom Convolution Neural Networks (CNN) model to recognize hand gestures.

Convolutional neural networks (CNNs), which can be trained on massive datasets of hand sign photos to precisely identify diverse movements, are a common technique for hand sign recognition. These networks can recognise hand signs with high degrees of accuracy because they often use numerous layers of filters and pooling to extract pertinent data from the input images.

Overall, machine learning for hand sign identification is a fast-expanding topic with a wide range of exciting applications, including sign language translation, HCI, and virtual reality.

Keywords— *Sign Language recognition (SLR), Convolution Neural Networks (CNN), American Sign Language (ASL)*

CHAPTER 1 INTRODUCTION

1.1 Overview

Sign language recognition (SLR) is a field of research in computer vision. SLR is necessary to address issues with video cutting, sign extraction, sign video background modelling, and sign classification. Because of the complexity and diversity of Sign Languages, Sign Language Recognition (SLR) is an extremely complicated issue. Every nation, for instance, has its own Sign Language and standards. Like how the American and Indian sign languages are distinct from one another. Indian Sign Language (ISL), for instance, employs two hands to represent the letter "a," but American Sign Language (ASL) only calls for one. In order to identify and translate American hand sign language into text, we will utilise convolution neural networks (CNN). Convolutional Neural Networks (CNN) are a subset of Artificial Neural Networks (ANN) that are frequently employed in image recognition, image categorization, object detection, etc**.**

This issue may be resolved by using a Convolution Neural Networks (CNN) model to identify hand gestures. Natural language processing, computer vision, pattern matching, and linguistics is all involved in the joint study area of sign language recognition. Its objective is to assemble numerous approaches and algorithms to recognise pre-existing indications and understand their significance. Sign language translation is among one of the most growing lines of research and it enables the maximum natural manner of communication for those with hearing impairments. A hand gesture recognition system offers an opportunity for deaf people to talk with vocal humans without the need of an interpreter. The system is built for the automated conversion of ASL into textual content and speech.

In our project we primarily focus on producing a model which can recognize Fingerspelling based hand gestures in order to form a complete word by combining each gesture. The gestures we aim to train are as given in the image below.

Sign language is a visual language and consists of 3 major components:

Fingerspelling	**Word level sign vocabulary**	**Non-manual features**
Used to spell words letter by letter .	Used for the majority of communication.	Facial expressions and tongue, mouth and body position.

Table 1.1 components of Sign language

American Sign Language was created in 1817 for deaf students. The main goal was to represent the letter and structure of the English language using hands, so that deaf students can use English. Until 1835, ASL was used as a language of instruction and student communication in schools for the deaf. Sooner the use of Sign Language spread throughout the world.

American Sign Language (ASL) is a complete, natural language that has the same linguistic properties as spoken languages, with grammar that differs from English. ASL is expressed by movements of the hands and face. It is the primary language of many North Americans who are deaf and hard of hearing, and is used by many hearing people as well.

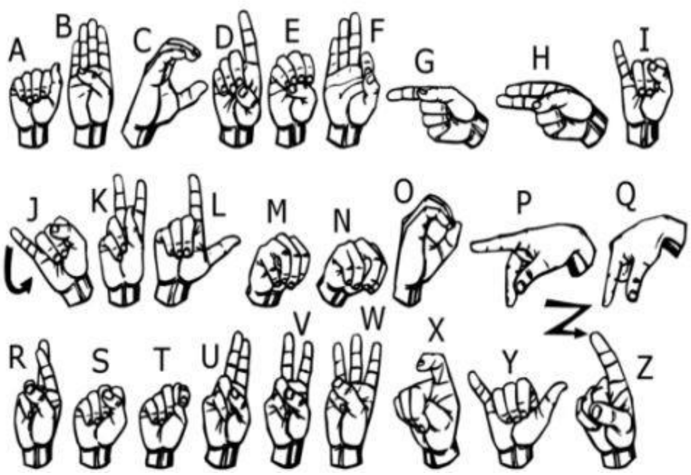

Fig 1.1 American Sign Language

1.2 Problem Statement

The deaf population and the hearing majority face an evident communication gap that can be closed by utilising novel concepts and methods that can automatically recognise sign language and translate it into text. The deaf and voiceless community uses sign language to interact with others, but the main difficulty they have is that not everyone can comprehend sign language. Although it can be employed, written communication is slower than face-to-face conversation. As sign language is more effective than textual communication in emergency circumstances, several sign languages have been developed across the world.

1.3 Importance of Hand Sign Language

It is a language that includes gestures made with the hands and other body parts, including facial expressions and postures of the body. It used primarily by people who are deaf and dumb. There are many different sign languages as, British, Indian and American sign languages. British sign language (BSL) is not easily intelligible to users of American sign Language (ASL) and vice versa.

A functioning signing recognition system could provide a chance for the inattentive communicate with non-signing people without the necessity for an interpreter. It might be wont to generate speech or text making the deaf more independent. Unfortunately, there has not been any system with these capabilities thus far. during this project our aim is to develop a system which may classify signing accurately.

Sign language reduces the barrier for communicating with the humans having impaired of speech and hearing, on the other hand Sign language cannot be easily understood by common people. Therefore, a platform is necessary that is built using an algorithm to recognize various signs it is called as Sign Language Recognition (SLR).

1.4 Objectives

Our main objective is to capture images from camera, extract hand image from the input image and convert it into text. We are going to use CNN for classification of images. CNN are regularized versions of multilayer perceptron's. Multilayer perceptrons usually mean fully connected networks, that is, each neuron in one layer is connected to all neurons in the next layer. For that we need to implement a convolutional layer, pooling layer, and fully connected layer.

Convolution layer performs a dot product between two matrices, where one matrix is a filter and the other matrix is the restricted portion of the image.

At the end of the convolution process, we have a **featured matrix** which has lesser parameters(dimensions) than the actual image as well as more clear features than the actual one.

Pooling layer is applied to decrease the computational power required to process the data. It is done by decreasing the dimensions of the featured matrix even more. After pooling layer, we get a matrix containing main features of the image and this matrix has even lesser dimensions. After we have done highlighting some features in an image and reduces the dimensions of the image drastically. we are going to do the classification process and display the letter corresponding to that image.

CHAPTER 2 REVIEW OF LITERATURE

2.1 Patent Search

1. SYSTEM AND METHOD FOR AUTOMATED SIGN LANGUAGE RECOGNITION [Zhengyu Zhou, Tobias Menne, Kehuang Li, Kui Xu ; US20160307469A1 ; 2017 **]** [10]

A method for sign language recognition includes receiving. with an input device, an input based on a plurality of hand movements and postures of a user that correspond to a sequence of signs, extracting a plurality of features from the input corresponding to the plurality of hand movements and postures, identifying, a start of the sequence of signs in the input based on a first set of features in the plurality of features and a first Hidden Markov Model (HMM) stored in the memory and identifying a first sign in the input based on a second set of features in the plurality of features and a second HMM stored in the memory. The method also includes generating an output corresponding to the first sign from the input.

2. HAND GESTURE RECOGNITION SYSTEM AND METHOD [Anthony Vernon Walker Smith, Alistair Ian Sutherland, Arnaud Lemoine, Sean Mcgrath ; USOO6128003A ; 2018 **]** [11]

Noise problems in processing small images or large granularity images are reduced by representing hand images as rotational vectors calculated using a real-valued centroid. The hand region is therefore sectored independently of pixel quantization. Colour segmentation is used to identify hand colour regions, followed by region labelling to filter out noise regions based on region size. Principal component analysis is used to plot gesture models.

3. SIGN BASED HUMAN-MACHINE INTERACTION [Kikuo Fujimura, Xia Liu ; WO2005114556A2 ; 2005 **]** [8]

Communication is an important issue in man-to-robot interaction. Signs can be used to interact with machines by providing user instructions or commands. Embodiments of the present invention include human detection, human body parts detection, hand shape analysis, trajectory analysis, orientation determination, gesture matching, and the like. Many types of shapes and gestures are recognized in a non-intrusive manner based on computer vision. A number of applications become feasible by this sign-understanding technology, including remote control of home devices, mouse-less (and touch-less) operation of computer consoles, gaming, and man-robot communication to give instructions among others. Active sensing hardware is used to capture a stream of depth images at a video rate, which is consequently analyzed for information extraction.

4. A SYSTEM FOR TRANSLATING SIGN LANGUAGE INTO SPEECH AND VICE-VERSA [SHILPI KATARIA, AMAN SOHI, SHAMINDER SINGH THAKUR, NILESHSINGH V.**;** AU2021101012 ; 2021 **]** [7]

A system for translating sign language of a deaf/dumb person into speech comprises an image capturing device, an image processing unit, a first repository, a feature matching unit, a feature recognition unit, and an audio unit. The image capturing device is configured to capture video of the deaf/dumb person speaking sign language, and generate a corresponding live image data stream. The image processing unit is configured to receive the captured live image data stream from the image capturing device and extract features of gesture of a left and right hand of the deaf/dumb person. The feature matching unit is configured to cooperate with the image processing unit to receive the extracted features of the gesture and match extracted features of the gesture with the stored dataset. The feature recognition unit is configured to recognize the gesture information based on temporal and spatial hand gesture variation.

2.2 Literature Review

1. Machine Learning-based Hand Sign Recognition [G. Pala, J. B. Jethwani, S. S. Kumbhar and S. D. Patil ; IEEE ; 2021 **]**

Greeshma Pala et. al. [1], has done a comparison study between K-Nearest neighbor (KNN), Multi-class Super Vector Machine (SVM), and Convolutional Neural Networks (CNN) algorithms to determine which algorithm will be able to recognized hand signs with more accuracy among three of them. Algorithms such as K-Nearest neighbor (KNN), Multi-class Super Vector Machine (SVM), and also experiments using hand gloves were used to decode the hand gesture movements before to find best way with high accuracy. Approximately 29,000 images were used and split into test and train dataset, after that all the images were pre-processed to fit into the KNN, SVM, and CNN models. After implementation and test of KNN, SVM, and CNN algorithms accuracy of 93.83%, 88.89%, and 98.49% was obtained respectively. It was concluded after that CNN is comparatively the best among KNN, SVM, and CNN algorithms with a good accuracy and minimum loss.

2. A Comprehensive Study on Deep Learning-based Methods for Sign Language Recognition [N. Adaloglou, Theocharis Chatzis, Ilias Papastratis; IEEE ; 2022 **]**

N. Adaloglou et. al. [3], has done a comparative experimental assessment of computer vision-based methods for SLR by implementing and evaluating multiple datasets which available to public. They have done an in-depth analysis of the most characteristic Deep Neural Network based SLR model architectures. Continuous Sign Language Recognition (CSLR) is a task very similar to the one of continuous human action recognition. There are sequence of glosses (instead of actions) needs to be identified in a continuous stream of video data. But glosses are involved in very small number of frames. The authors also find out that 3D CNN based architectures are more efficient in isolated SLR than 2D CNN based model.

3. Sign Language Recognition Using Convolutional Neural Networks [Suharjito, H. Gunawan, N. Thiracitta and A. Nugroho; INAPR ; 2018 **]**

Herman Gunawan et. al. [4], try to implement one of model which is i3d inception which is also a new Action Recognition model with very high accuracy. They try to find out if it is possible to adopt Action Recognition behaviour into Sign Language Recognition. The goal of this paper is to implement the i3d inception model to Sign Language Recognition model.

4. Sign Language Recognition Systems: A Decade Systematic Literature Review [Ankita Wadhawan and Parteek Kumar; CIMNE; 2019 **]**

Ankita Wadhawan and Parteek Kumar [6], has done a Systematic Literature Review and a classification scheme in the paper in between 2007 to 2017. One hundred and seventeen research articles were selected by the authors and reviewed. Each of 117 selected papers were compared with each other on the basis of the method and process they used to acquire data, single or double signs language used, classification technique used and accuracy. It was found that most of the major research on sign language recognition has been performed in isolated and using single handed signs language while taking input from camera. The main aim was to provide readers and researchers a roadmap to guide future research and facilitate knowledge accumulation in sign language recognition.

5. Deep Convolutional Neural Networks for Sign Language Recognition [G. Anantha Rao, K. Syamala, P. V. Kishore; IEEE; 2018 **]**

G. Anantha Rao et. al. [2], has created a mobile Indian sign language gestures recognition app using Convolutional Neural Networks (CNN) in which a video is captured as input so that the other can capture the video and text is displayed on the screen. The 200 signs were performed by five separate participants, each of whom took up 60 frames of film. They performed the signs while being viewed from five different perspectives. To improve hand sign recognition accuracy, various CNN architectures were developed and tested. In the end, the authors outperformed other classification models on the same dataset with accuracy of 92.88%.

6. The Efficiency of Sign Language Recognition using 3D Convolutional Neural Networks [N. Soodtoetong and E. Gedkhaw; IEEE; 2018 **]**

N. Soodtoetong and E. Gedkhaw [16], has studied process and method which related with the recognition of sign language using deep learning. 3D-CNN algorithm was used for recognized process and input was taken by capturing images from the Kinect Sensor. The Kinect Sensor was used to collect RGB image, which were then classified using 3D CNN. The result show that the 3D-CNN algorithm could recognize the gesture motion and have the accuracy of 91.23%.

7. Sign language recognition: State of the art [Sahoo Ashok, Mishra Gouri and Kiran Ravulakollu; ARPN ; 2016 **]** [20]

Sign language is used by deaf and hard hearing people to exchange information between their own community and with other people. Computer recognition of sign language deals from sign gesture acquisition and continues till text/speech generation. Sign gestures can be classified as static and dynamic. However static gesture recognition is simpler than dynamic gesture recognition but both recognition systems are important to the human community. The sign language recognition steps are described in this survey. The data acquisition, data pre-processing and transformation, feature extraction, classification and results obtained are examined. Some future directions for research in this area also suggested.

8. Real Time Sign Language Recognition and Speech Generation [Amrita Thakur, Pujan Budhathoki, Sarmila Upreti, Shirish Shrestha, Subarna Shakya; JIIP; 2020 **]** [12]

The commercialization of an economical and accurate recognition system is today's concern of researchers all over the world. Thus, sign language recognition systems based on Image processing and neural networks are preferred over gadget system as they are more accurate and easier to make. The aim of this paper is to build a user friendly and accurate sign language recognition system trained by neural network thereby generating text and speech of the input gesture. This paper also presents text to sign language generation model that enables a way to establish a two-way communication without the need of a translator.

9. Sign language recognition using image based hand gesture recognition techniques [A. S. Nikam and A. G. Ambekar; IEEE; 2016 **]** [13]

This paper introduced software which presents a system prototype that is able to automatically recognize sign language to help deaf and dumb people to communicate more effectively with each other or normal people. Pattern recognition and Gesture recognition are the developing fields of research. Being a significant part in nonverbal communication hand gestures are playing key role in our daily life. Hand Gesture recognition system provides us an innovative, natural, user friendly way of communication with the computer which is more familiar to the human beings. By considering in mind the similarities of human hand shape with four fingers and one thumb, the software aims to present a real time system for recognition of hand gesture on basis of detection of some shape based features like orientation, Centre of mass centroid, fingers status, thumb in positions of raised or folded fingers of hand.

CHAPTER 3 WORK DONE

3.1 Artificial Neural Network (ANN):

The term "Artificial neural network" refers to a biologically inspired sub-field of artificial intelligence modelled after the brain. An Artificial neural network is usually a computational network based on biological neural networks that construct the structure of the human brain. Similar to a human brain has neurons interconnected to each other, artificial neural networks also have neurons that are linked to each other in various layers of the networks. These neurons are known as nodes.

The term "**Artificial Neural Network**" is derived from Biological neural networks that develop the structure of a human brain. Similar to the human brain that has neurons interconnected to one another, artificial neural networks also have neurons that are interconnected to one another in various layers of the networks.

Artificial Neural Network is a connection of neurons, replicating the structure of human brain. Each connection of neuron transfers information to another neuron. Inputs are fed into first layer of neurons which processes it and transfers to another layer of neurons called as hidden layers. After processing of information through multiple layers of hidden layers, information is passed to final output layer.

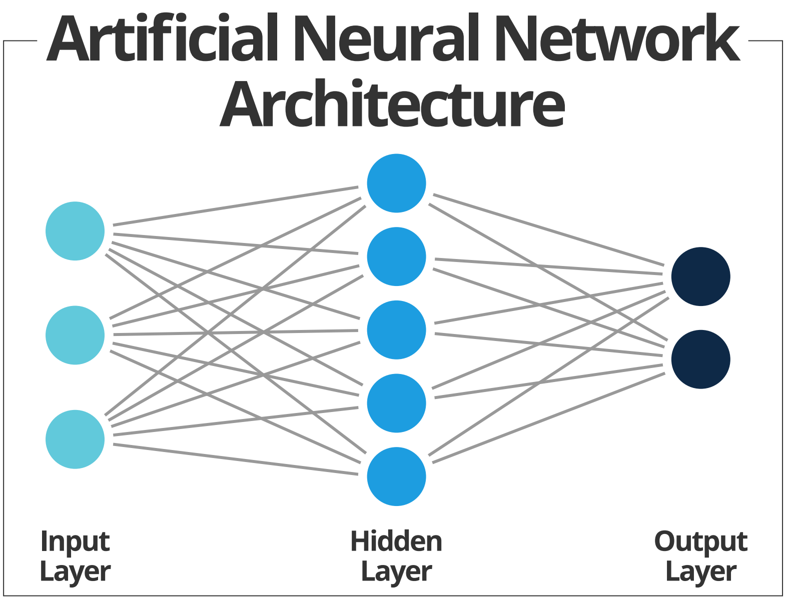

Fig 3.1 Artificial Neural Network

1. **Input Layer:**

As the name suggests, it accepts inputs in several different formats provided by the programmer.

2. **Hidden Layer:**

The hidden layer presents in-between input and output layers. It performs all the calculations to find hidden features and patterns.

3. **Output Layer:**

The input goes through a series of transformations using the hidden layer, which finally results in output that is conveyed using this layer.

The artificial neural network takes input and computes the weighted sum of the inputs and includes a bias. This computation is represented in the form of a transfer function.

$$\sum_{i=1}^{n} Wi * Xi + b$$

It determines weighted total is passed as an input to an activation function to produce the output. Activation functions choose whether a node should fire or not. Only those who are fired make it to the output layer. There are distinctive activation functions available that can be applied upon the sort of task we are performing.

These are capable of learning and have to be trained. There are different learning strategies:

1. Unsupervised Learning: As the name suggests, unsupervised learning is a machine learning technique in which models are not supervised using training dataset. Instead, models itself find the hidden patterns and insights from the given data. It can be compared to learning which takes place in the human brain while learning new things. Unsupervised learning cannot be directly applied to a regression or classification problem because unlike supervised learning, we have the input data but no corresponding output data. The goal of unsupervised learning is to find the underlying structure of dataset, group that data according to similarities, and represent that dataset in a compressed format.

2. Supervised Learning: Supervised learning is the types of machines learning in which machines are trained using well "labelled" training data, and on basis of that data, machines predict the output. The labelled data means some input data is already tagged with the correct output.

In supervised learning, the training data provided to the machines work as the supervisor that teaches the machines to predict the output correctly. It applies the same concept as a student learns in the supervision of the teacher.

3. Reinforcement Learning: Reinforcement Learning is a feedback-based Machine learning technique in which an agent learns to behave in an environment by performing the actions and seeing the results of actions. For each good action, the agent gets positive feedback, and for each bad action, the agent gets negative feedback or penalty.

The agent learns with the process of hit and trial, and based on the experience, it learns to perform the task in a better way. Hence, we can say that "Reinforcement learning is a type of machine learning method where an intelligent agent (computer program) interacts with the environment and learns to act within that.

3.2 Convolutional Neural Network (CNN):

A Convolutional Neural Network (CNN) is a type of deep learning algorithm that is particularly well-suited for image recognition and processing tasks. It is made up of multiple layers, including convolutional layers, pooling layers, and fully connected layers. The convolutional layers are the key component of a CNN, where filters are applied to the input image to extract features such as edges, textures, and shapes. The output of the convolutional layers is then passed through pooling layers, which are used to down-sample the feature maps, reducing the spatial dimensions while retaining the most important information. The output of the pooling layers is then passed through one or more fully connected layers, which are used to make a prediction or classify the image.

CNNs are trained using a large dataset of labelled images, where the network learns to recognize patterns and features that are associated with specific objects or classes. Once trained, a CNN can be used to classify new images, or extract features for use in other applications such as object detection or image segmentation.

They are widely used in computer vision, image processing, and other related fields, and have been applied to a wide range of applications, including self-driving cars, medical imaging, and security systems.

- A convolutional neural network, or CNN, is a deep learning neural network sketched for processing structured arrays of data such as portrayals.
- CNN are very satisfactory at picking up on design in the input image, such as lines, gradients, circles, or even eyes and faces.
- This characteristic that makes convolutional neural network so robust for computer vision.
- CNN can run directly on a underdone image and do not need any pre-processing.
- The strength of a convolutional neural network comes from a particular kind of layer called the convolutional layer.
- CNN contains many convolutional layers assembled on top of each other, each one competent of recognizing more sophisticated shapes.
- With three or four convolutional layers it is viable to recognize handwritten digits and with 25 layers it is possible to differentiate human faces.

- The agenda for this sphere is to activate machines to view the world as humans do, perceive it in a alike fashion and even use the knowledge for a multitude of duty such as image and video recognition, image inspection and classification, media recreation, recommendation systems, natural language processing, etc.

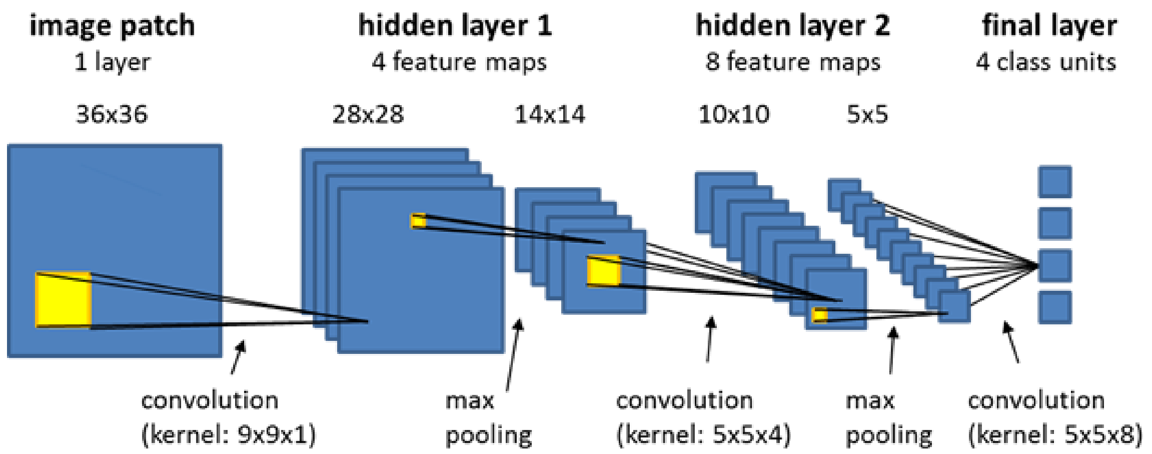

Fig 3.2 Convolutional Neural Network

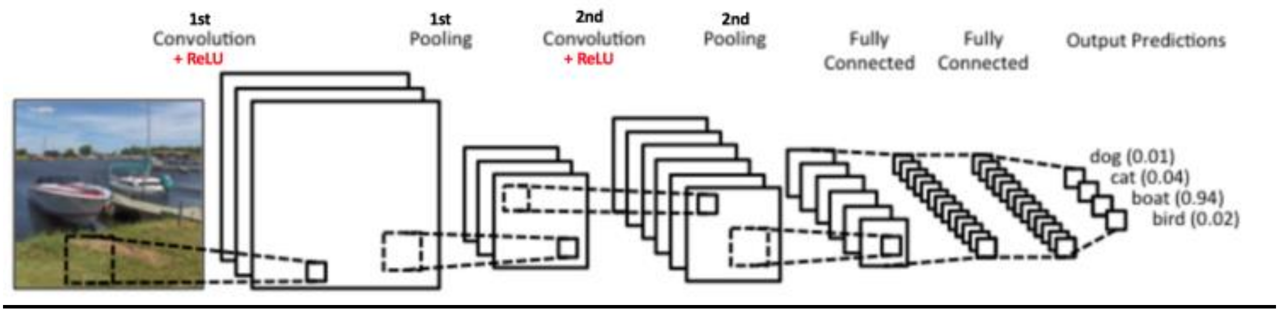

Fig 3.3 Convolutional Neural Network Layers

3.2.1 Convolution Layer :

In convolution layer we take a small window size [typically of length 5*5] that extends to the depth of the input matrix. The layer consists of learnable filters of window size. During every iteration we slid the window by stride size [typically 1], and compute the dot product of filter entries and input values at a given position.

As we continue this process, we will create a 2-Dimensional activation matrix that gives the response of that matrix at every spatial position. That is, the network will learn filters that activate when they see some type of visual feature such as an edge of some orientation or a blotch of some colour.

3.2.2 Pooling Layer:

We use pooling layer to decrease the size of activation matrix and ultimately reduce the learnable parameters. There are two types of pooling:

a. Max Pooling: In max pooling we take a window size [for example window of size 2*2], and only take the maximum of 4 values. Well lid this window and continue this process, so well finally get an activation matrix half of its original Size.

b. Average Pooling: In average pooling, we take advantage of all Values in a window.

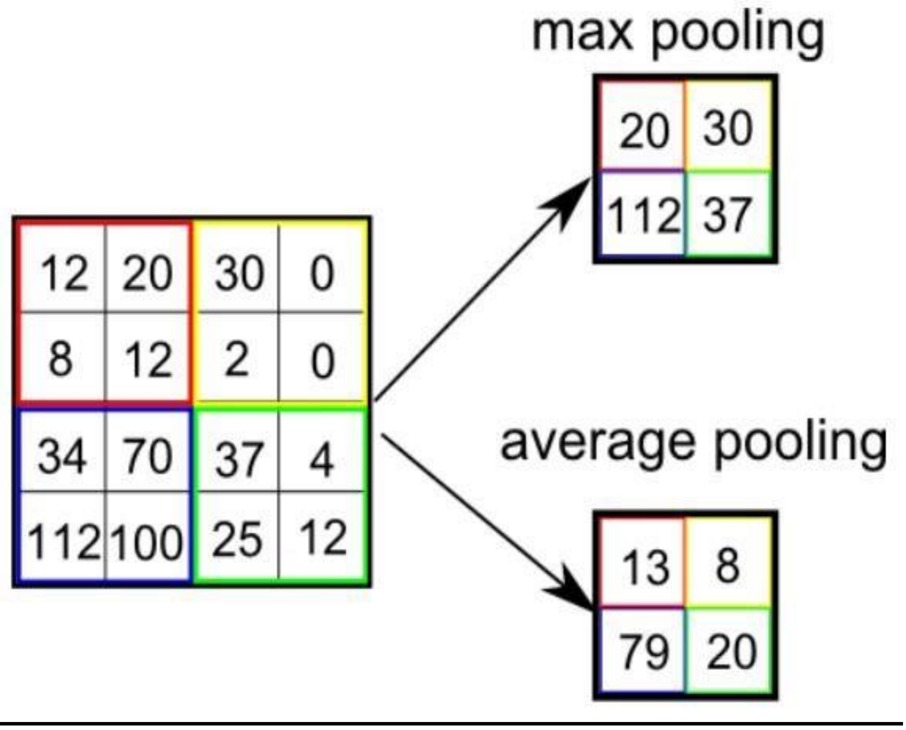

Fig 3.4 Pooling Layer

3.2.3 Fully Connected Layer:

In convolution layer, neurons are connected only to a local region, while in a fully connected region, we will connect all the inputs to neurons.

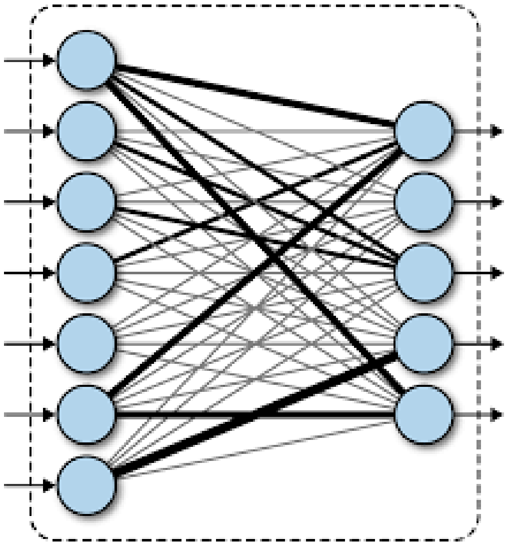

Fig 3.5 Fully Connected Layer

3.2.4 Final Output Layer:

After getting values from fully connected layer, we will connect them to the final layer of neurons [having count equal to total number of classes], that will predict the probability of each image to be in different classes.

3.2.5 TensorFlow:

TensorFlow is an end-to-end open-source platform for Machine Learning. It has a comprehensive, flexible ecosystem of tools, libraries and community resources that lets researchers push the state-of-the-art in Machine Learning and developers easily build and deploy Machine Learning powered applications.

TensorFlow offers multiple levels of abstraction so you can choose the right one for your needs. Build and train models by using the high-level Keras API, which makes getting started with TensorFlow and machine learning easy.

If you need more flexibility, eager execution allows for immediate iteration and intuitive debugging. For large ML training tasks, use the Distribution Strategy API for distributed training on different hardware configurations without changing the model definition.

3.2.6 Keras:

Keras is a high-level neural networks library written in python that works as a wrapper to TensorFlow. It is used in cases where we want to quickly build and test the neural network with minimal lines of code. It contains implementations of commonly used neural network elements like layers, objective, activation functions, optimizers, and tools to make working with images and text data easier.

3.2.7 OpenCV:

OpenCV (Open-Source Computer Vision) is an open-source library of programming functions used for real-time computer-vision.

It is mainly used for image processing, video capture and analysis for features like face and object recognition. It is written in C++ which is its primary interface, however bindings are available for Python, Java, MATLAB/OCTAVE.

3.3 Methodology:

The system is a vision-based approach. All signs are represented with bare hands and so it eliminates the problem of using any artificial devices for interaction.

3.3.1 Data Set Generation:

For the project we tried to find already made datasets but we couldn't find dataset in the form of raw images that matched our requirements. All we could find were the datasets in the form of RGB values. Hence, we decided to create our own data set. Steps we followed to create our data set are as follows.

We used Open computer vision (OpenCV) library in order to produce our dataset.

Firstly, we captured around 800 images of each of the symbol in ASL (American Sign Language) for training purposes and around 200 images per symbol for testing purpose. First, we capture each frame shown by the webcam of our machine. In each frame we define a Region Of Interest (ROI) which is denoted by a blue bounded square as shown in the image below:

Fig 3.6 Dataset Creation

Then, we apply Gaussian Blur Filter to our image which helps us extract various features of our image. The image, after applying Gaussian Blur, looks as follows:

Fig 3.7 Pre-processes image

3.3.2 Gesture Classification:

Our approach uses two layers of algorithm to predict the final symbol of the user.

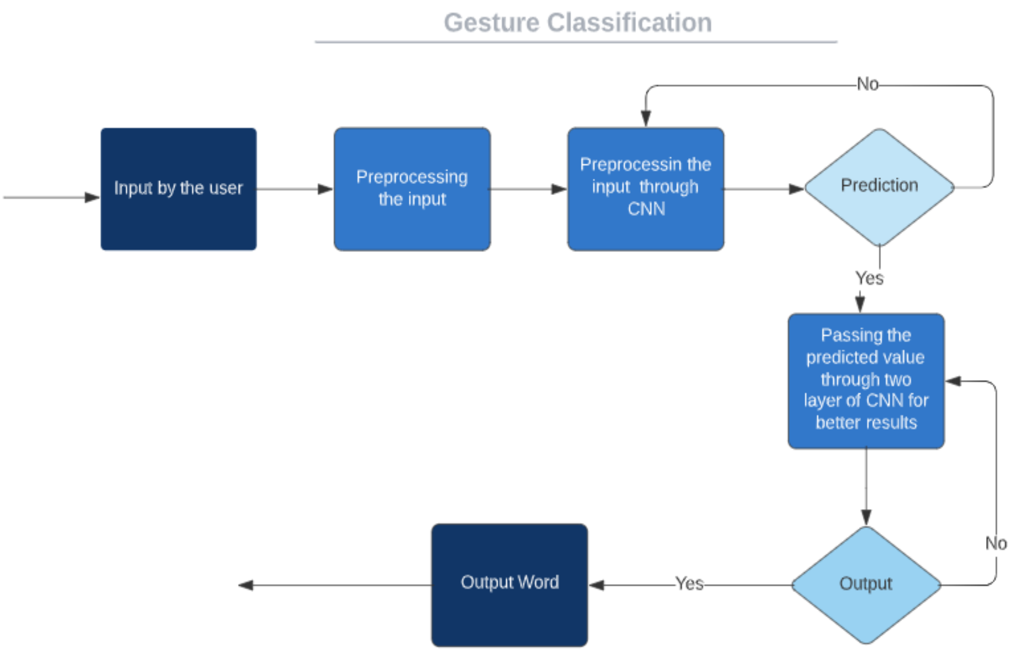

Fig 3.8 Flow Diagram

Algorithm Layer 1:

1. Apply Gaussian Blur filter and threshold to the frame taken with openCV to get the processed image after feature extraction.
2. This processed image is passed to the CNN model for prediction and if a letter is detected for more than 50 frames then the letter is printed and taken into consideration for forming the word.
3. Space between the words is considered using the blank symbol.

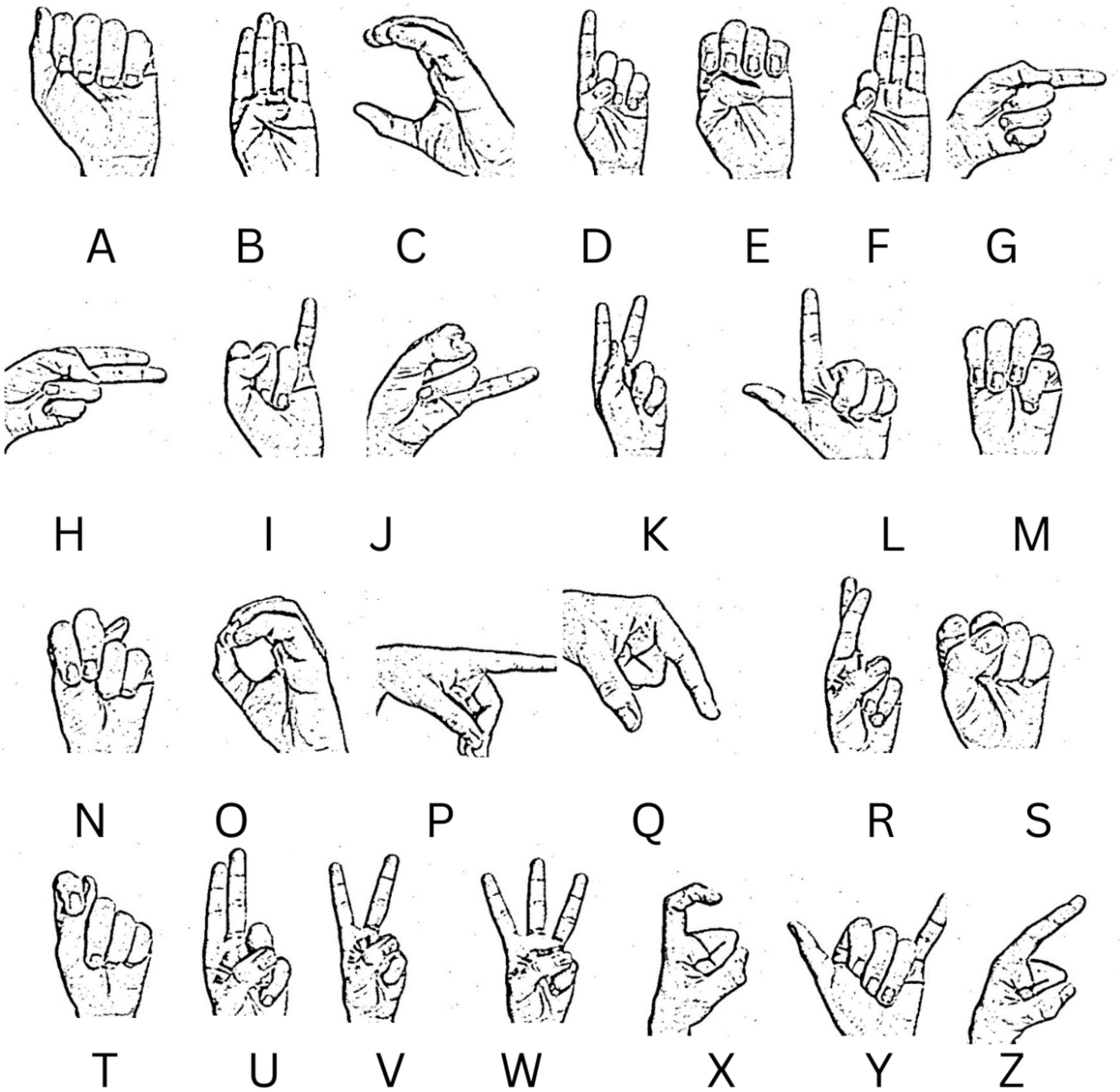

Fig 3.9 Pre – process images

Algorithm Layer 2:

1. We detect various sets of symbols which show similar results on getting detected.
2. We then classify between those sets using classifiers made for those sets only.

Layer 1:

- **CNN Model:**

1. **1st Convolution Layer:** The input picture has resolution of 128x128 pixels. It is first processed in the first convolutional layer using 32 filter weights (3x3 pixels each). This will result in a 126X126 pixel image, one for each Filter-weights.
2. **1st Pooling Layer:** The pictures are down sampled using max pooling of 2x2 i.e we keep the highest value in the 2x2 square of array. Therefore, our picture is down sampled to 63x63 pixels.
3. **2nd Convolution Layer:** Now, these 63 x 63 from the output of the first pooling layer is served as an input to the second convolutional layer. It is processed in the second convolutional layer using 32 filter weights (3x3 pixels each). This will result in a 60 x 60 pixel image.
4. **2nd Pooling Layer:** The resulting images are down sampled again using max pool of 2x2 and is reduced to 30 x 30 resolution of images.
5. **1st Densely Connected Layer:** Now these images are used as an input to a fully connected layer with 128 neurons and the output from the second convolutional layer is reshaped to an array of 30x30x32 =28800 values. The input to this layer is an array of 28800 values. The output of these layer is fed to the 2nd Densely Connected Layer. We are using a dropout layer of value 0.5 to avoid overfitting.
6. **2nd Densely Connected Layer:** Now the output from the 1st Densely Connected Layer is used as an input to a fully connected layer with 96 neurons.
7. **Final layer:** The output of the 2nd Densely Connected Layer serves as an input for the final layer which will have the number of neurons as the number of classes we are classifying (alphabets + blank symbol).

- **Activation Function:**

  We have used ReLU (Rectified Linear Unit) in each of the layers (convolutional as well as fully connected neurons).

  ReLU calculates max(x,0) for each input pixel. This adds nonlinearity to the formula and helps to learn more complicated features. It helps in removing the vanishing gradient problemand speeding up the training by reducing the computation time.

- **Pooling Layer:**

  We apply **Max** pooling to the input image with a pool size of (2, 2) with ReLU activation function. This reduces the amount of parameters thus lessening the computation cost and reduces overfitting.

- **Dropout Layers:**

  The problem of overfitting, where after training, the weights of the network are so tuned to the training examples they are given that the network doesn't perform well when given new examples. This layer "drops out" a random set of activations in that layer by setting them to zero. The network should be able to provide the right classification or output for a specific example even if some of the activations are dropped out [5].

- **Optimizer:**

  We have used Adam optimizer for updating the model in response to the output of the loss function.

  Adam optimizer combines the advantages of two extensions of two stochastic gradient descent algorithms namely adaptive gradient algorithm (ADA GRAD) and root mean square propagation (RMSProp).

Layer 2:

We are using two layers of algorithms to verify and predict symbols which are more similar to each other so that we can get us close as we can get to detect the symbol shown. In our testing we found that following symbols were not showing properly and were giving other symbols also:

1. For D : R and U
2. For U : D and R
3. For I : T, D, K and I
4. For S : M and N

So, to handle above cases we made three different classifiers for classifying these sets:

1. {D, R, U}
2. {T, K, D, I}
3. {S, M, N}

3.3.3 Finger Spelling Sentence Formation Implementation:

1. Whenever the count of a letter detected exceeds a specific value and no other letter is close to it by a threshold, we print the letter and add it to the current string (In our code we kept the value as 50 and difference threshold as 20).
2. Otherwise, we clear the current dictionary which has the count of detections of present symbol to avoid the probability of a wrong letter getting predicted.
3. Whenever the count of a blank (plain background) detected exceeds a specific value and if the current buffer is empty no spaces are detected.
4. In other case it predicts the end of word by printing a space and the current gets appended to the sentence below.

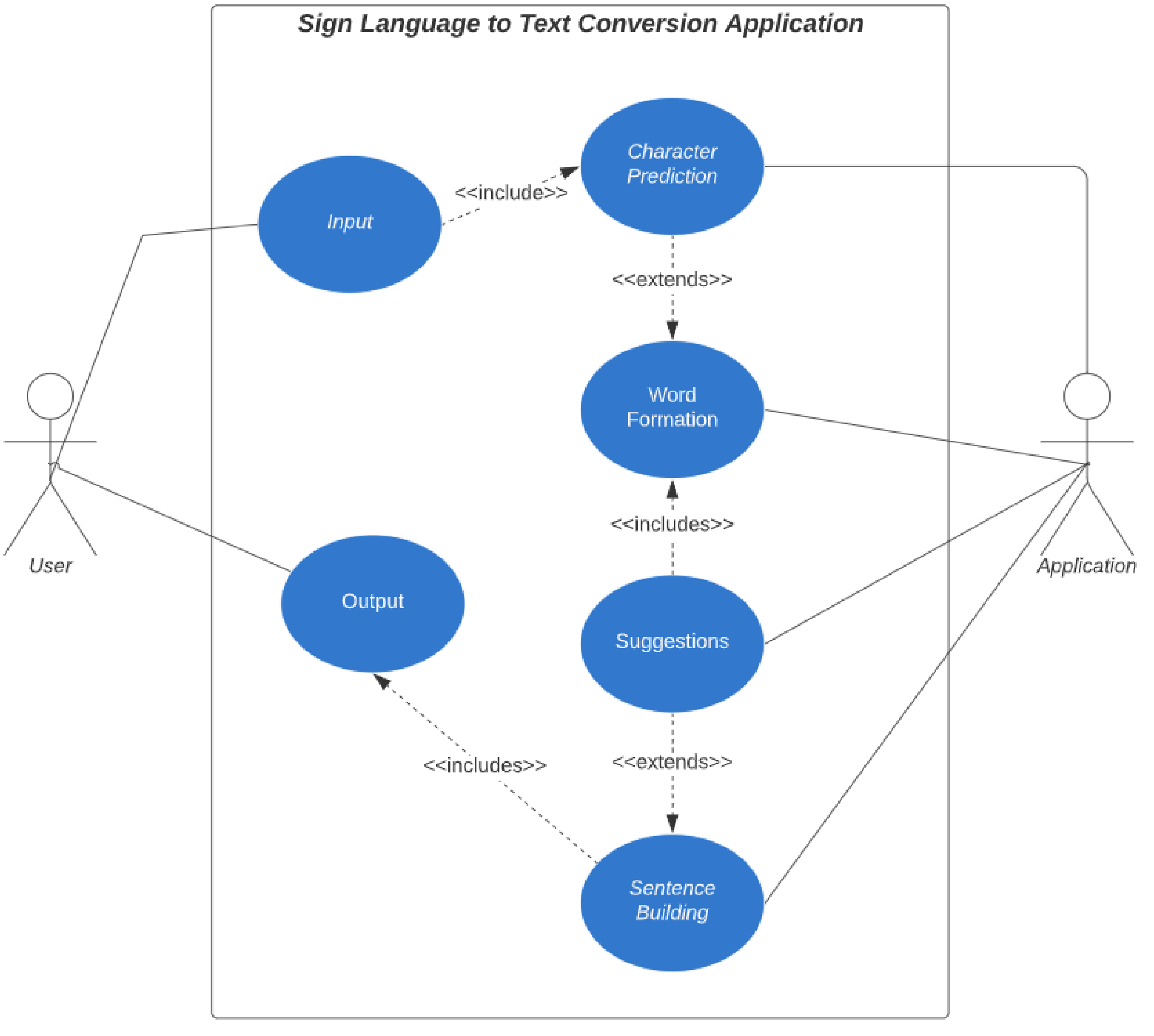

Fig 3.9 UML Diagram

3.3.4 Training and Testing:

We convert our input images (RGB) into grayscale and apply gaussian blur to remove unnecessary noise. We apply adaptive threshold to extract our hand from the background and resize our images to 128 x 128.

We feed the input images after pre-processing to our model for training and testing after applying all the operations mentioned above.

The prediction layer estimates how likely the image will fall under one of the classes. So, the output is normalized between 0 and 1 and such that the sum of each value in each class sums to 1. We have achieved this using SoftMax function.

At first the output of the prediction layer will be somewhat far from the actual value. To make it better we have trained the networks using labelled data. The cross-entropy is a performance measurement used in the classification. It is a continuous function which is positive at values which is not same as labelled value and is zero exactly when it is equal to the labelled value. Therefore, we optimized the cross-entropy by minimizing it as close to zero. To do this in our network layer we adjust the weights of our nural networks. TensorFlow has an inbuilt function to calculate the cross entropy.

As we have found out the cross-entropy function, we have optimized it using Gradient Descent in fact with the best gradient descent optimizer is called Adam Optimizer.

3.3.5 Challenges Faced:

There were many challenges faced during the project. The very first issue we faced was that concerning the data set. We wanted to deal with raw images and that too square images as CNN in Keras since it is much more convenient working with only square images.

We couldn’t find any existing data set as per our requirements and hence we decided to make our own data set. Second issue was to select a filter which we could apply on our images so that proper features of the images could be obtained and hence then we could provide that image as input for CNN model.

We tried various filters including binary threshold, canny edge detection, Gaussian blur etc. but finally settled with Gaussian Blur Filter.

More issues were faced relating to the accuracy of the model we had trained in the earlier phases. This problem was eventually improved by increasing the input image size and also by improving the data set.

CHAPTER 4 RESULTS AND DISCUSSIONS

4.1 Results:

By capturing data from camera in real time as input and converting it into text with the help of CNN model. As we can see in figure the meaning of hand sign is predicted and displayed on the screen. We have used American Sign Language to implement the model in the project. As the accuracy of the model is not that high, we can try to increase it by implementing more layer and number of epochs used to train the CNN models.

```
Models > Model.ipynb > Part 3 - Training the CNN > Saving the Model > model_json = classifier.to_json()
+ Code  + Markdown | Run All  Clear All Outputs  Restart | Variables  Outline ...

Output exceeds the size limit. Open the full output data in a text editor
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 128, 128, 32)      320
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 64, 64, 32)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 64, 64, 32)        9248
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 32, 32, 32)        0
_________________________________________________________________
flatten (Flatten)            (None, 32768)             0
_________________________________________________________________
dense (Dense)                (None, 128)               4194432
_________________________________________________________________
dense_1 (Dense)              (None, 128)               16512
_________________________________________________________________
dropout (Dropout)            (None, 128)               0
_________________________________________________________________
dense_2 (Dense)              (None, 96)                12384
_________________________________________________________________
dropout_1 (Dropout)          (None, 96)                0
_________________________________________________________________
dense_3 (Dense)              (None, 64)                6208
...
Total params: 4,240,859
Trainable params: 4,240,859
Non-trainable params: 0
_________________________________________________________________

classifier.fit(training_set,
```

Fig 4.1 Summarized review of CNN Layers

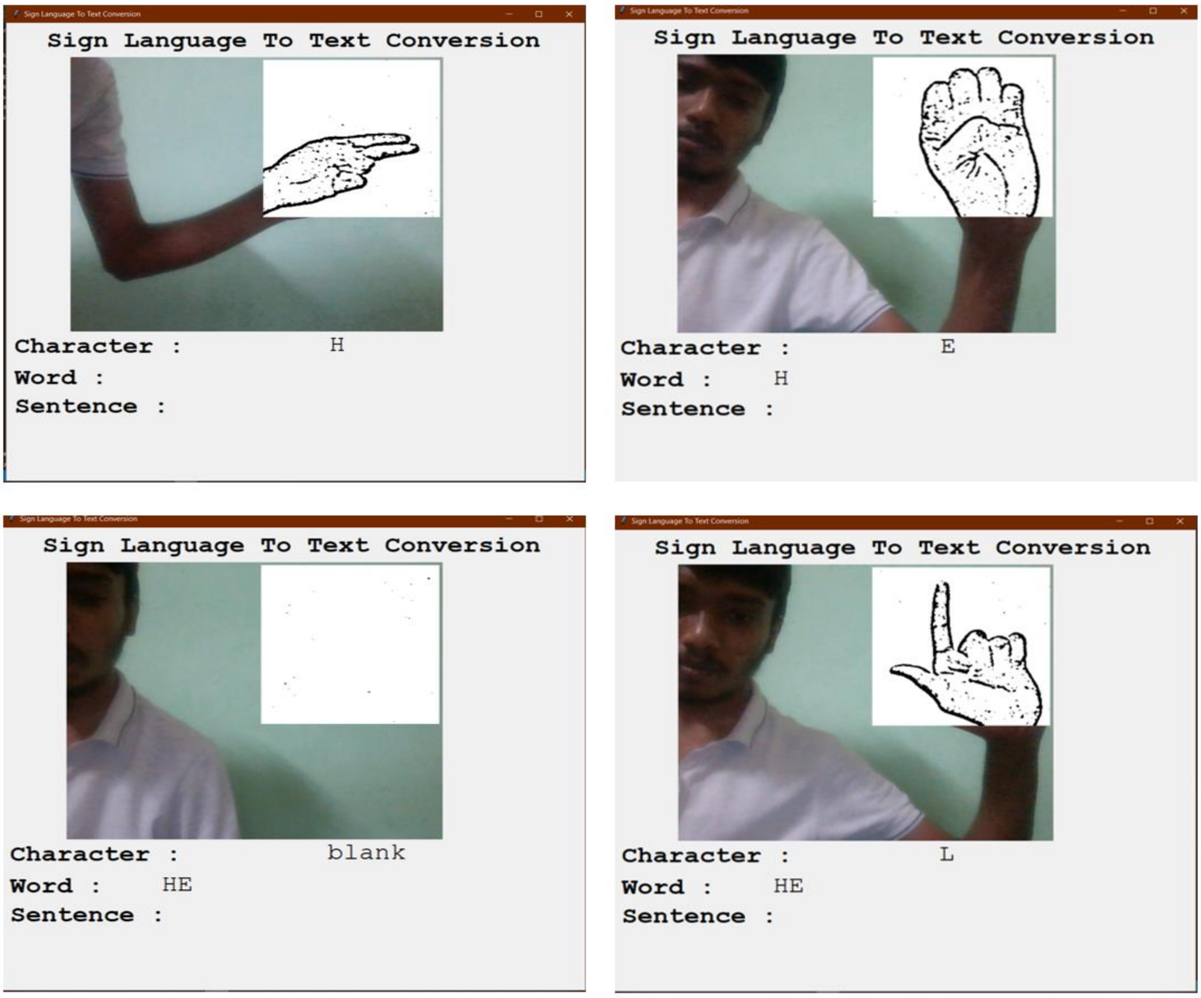

Fig. 4.2 Printing HELLO [1]

In above figure we are trying to print “HELLO” by using American Hand Sign Language. First, the character are recognised and is shown in front of **Character,** after that character is stored in **Word** so that words can created using sign language and using those word sentences are formed. Above figure shows how character H, E, L are recognised and saved so that word can be formed.

Fig. 4.3 Printing HELLO [2]

As show in above figure 4.2 we can see that how word "HELLO" is recognised and displayed on screen, while taking input from camera in real time and recognising and displayed on the screen. These words are used for making sentences as shown in figure 4.3 .

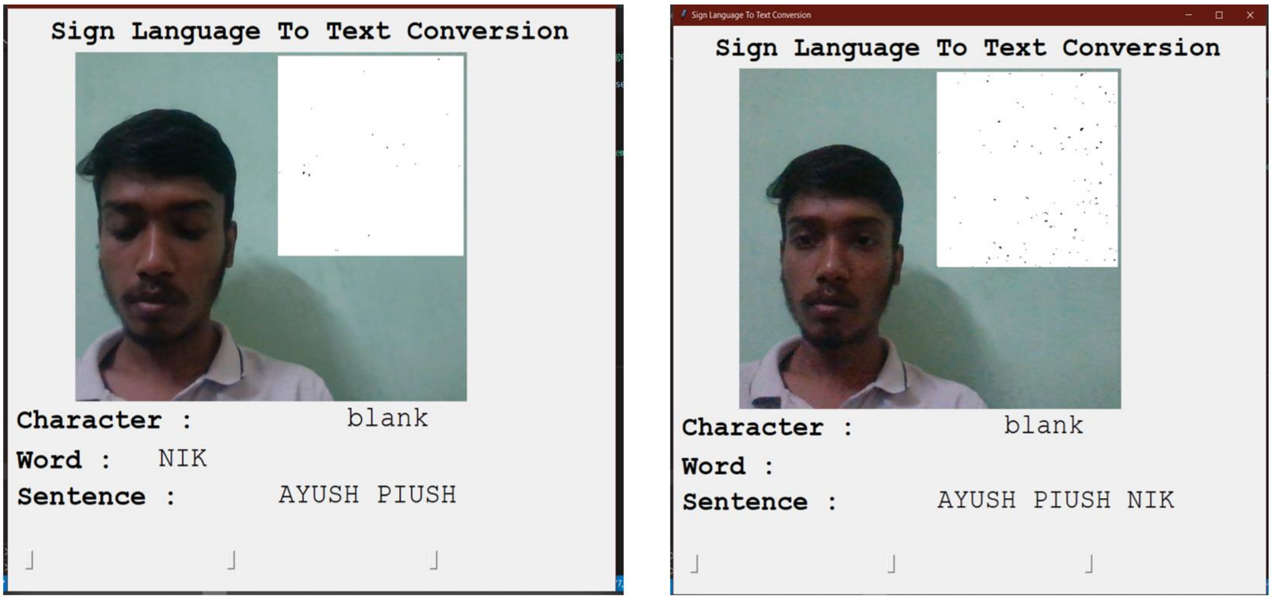

Fig 4.4 Output

In above figure we can see that word **NIK** is added to the sentence **AYUSH PIUSH** after some time and we get **AYUSH PIUSH NIK** sentence. In this way we can communicate with people who cannot speak using computer application.

CHAPTER 5 SUMMARY AND CONCLUSION

5.1 Social Utilities

Sign language recognition has significant societal relevance for several reasons:

Accessibility: Sign language recognition can provide access to information and communication for the deaf and hard-of-hearing community. It can help to break down communication barriers and enable deaf people to participate more fully in society.
Inclusion: Sign language recognition can promote inclusivity and diversity in society by recognizing and valuing the linguistic and cultural identity of the deaf and hard-of-hearing community.

Education: Sign language recognition can facilitate the learning of sign language and provide support for deaf and hard-of-hearing students in educational settings. It can also provide opportunities for hearing students to learn sign language and interact with the deaf community.

Employment: Sign language recognition can help to increase employment opportunities for deaf and hard-of-hearing individuals. It can enable them to work in fields where communication is critical, such as healthcare, education, and customer service.

Technology: Sign language recognition can drive advancements in technology and artificial intelligence by developing new techniques and algorithms for recognizing and interpreting gestures and movements.

Overall, sign language recognition has the potential to make a significant impact on the lives of deaf and hard-of-hearing individuals, promoting inclusivity and accessibility, and creating a more equitable society.

5.2 Summary:

There is a communication problem between the deaf community and the hearing majority which can be solved by using innovation ideas and techniques which can be used to automatically recognise sign language and convert it into text. Sign Language is used by the deaf and dumb community to communicate with others, but the most common problem here is that people around them cannot understand sign language. Written communication can be also be used but it is slow as compared to face-to-face communication. In case emergency situations its more efficient to use sign language than written communication therefore, many sign languages were created in different countries of world. To solve this problem, we can use a custom Convolution Neural Networks (CNN) model to recognize hand gestures.

5.3 Conclusion:

In this report, a functional real time vision based American Sign Language recognition for D&M people have been developed for asl alphabets.

Sign Language recognition (SLR) is very complex topic because of the complexity and diversity of Sign Languages. Sign language recognition is a collaborative research field that includes natural language processing, computer vision, pattern matching, and linguistics. There are many different types of Sign Language used in region of world. For example, Indian sign language (ISL) uses two hand to symbolize letter "a" while American Sign Language (ASL) required only one hand. We can build UI so that it can become user friendly to use.

This gives us the ability to detect almost all the symbols provided that they are shown properly.

5.4 Future Scope:

1. The aim of this project is to build a real time SLR computer application that can automatically detect hand gestures in natural lighting conditions and convert that into text.

2. The model can be further trained with a dataset such that its accuracy does not get affected by the background of image gesture which is taken from camera.

3. Modules can be optimized and integrated in AI systems which are used by general public so that it can available to use. For example, Google assistant.

CHAPTER 6 APPENDIX

Appendix A: Publication Details

Sr. No.	Authors	Title of Paper	Name of International Conference	Place and date and details of Publication
1	Prof. P. Moon Nikhil Dapkekar Piyush Dabare Ayush Raut Jay Rathod	Hand Sign Recognition Using Machine Learning	International Journal of Advanced Research in Science, Communication and Technology (IJARSCT)	Submitted – 18/05/2023. Acceptance – In Progress

Table 6.1 Table of Publication

Appendix B: Resumes

1. Piyush Dabare

YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

(An Autonomous Institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University)

Piyush Ramrao Dabare

Mobile: +91-7744806955 | Email-id:piyushdabare2001@gmail.com | Date of Birth: 26/04/2001 Address: 6, Ekta Nagar, Borgaon, Nagpur - 440013

OBJECTIVE:

Seeking a position in the field of Computer Science where I can utilize my skills to further work towards personal and professional development and contribute towards the prosperity of the organization.

ACADEMIC QUALIFICATION:

Enrollment Number: 19010898

Branch of Study: Computer Technology

QUALIFICATION	INSTITUTE	YEAR	SCORE/SGPA
B.E. 7^{th} Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2022	7.28/10
B.E. 6^{th} Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2022	7.59/10
B.E. 5^{th} Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2021	7.15/10
B.E. 4^{th} Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2021	7.08/10
B.E. 3^{rd} Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2020	7.63/10
B.E. 2^{nd} Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2020	9.04/10
B.E. 1^{st} Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2019	8.71/10

TECHNICAL ACTIVITIES/PROJECTS:

Hand Sign Recognition Using Machine Learning

Synopsis: Computer application used to capture sign language images from camera, extract hand image from the input image and convert it into text by using CNN for classification of images and display it on screen.

COMPUTER PROFICIENCY

Programming Languages: python

Libraries used: OpenCV, Tensorflow, Keras

2. Nikhil Dapkekar

YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

(An Autonomous Institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University)

Nikhil Balaji Dapkekar

Mobile: +91-9405611564 | Email-id: nikhildapkekar42031@gmail.com | Date of Birth: 30/01/2000 Address: 45, lokmanya nagar, Nagpur - 440016

OBJECTIVE:

Seeking a position in the field of Computer Science where I can utilize my skills to further work towards personal and professional development and contribute towards the prosperity of the organization.

ACADEMIC QUALIFICATION:

Enrollment Number: 19010995

Branch of Study: Computer Technology

QUALIFICATION	INSTITUTE	YEAR	SCORE/SGPA
B.E. 7th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2022	7.03/10
B.E. 6th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2022	6.5/10
B.E. 5th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2021	7.25/10
B.E. 4th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2021	7.05/10
B.E. 3rd Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2020	6.5/10
B.E. 2nd Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2020	8.7/10
B.E. 1st Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2019	6.9/10

TECHNICAL ACTIVITIES/PROJECTS:

Hand Sign Recognition Using Machine Learning

Synopsis: Computer application used to capture sign language images from camera, extract hand image from the input image and convert it into text by useing CNN for classification of images and display it on screen.

COMPUTER PROFICIENCY

Programming Languages: python

Libraries used: OpenCV, Tensorflow, Keras

3. Ayush Raut

YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

(An Autonomous Institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University)

Ayush Gangadhar Raut

Mobile: +91-9172553507 | Email-id: aayushr512@gmail.com | Date of Birth: 05/02/2002

Address: 49, Siddheshwar Nagar Kharbi Middle Ring Road Nagpur

OBJECTIVE:

Seeking a position in the field of Computer Science where I can utilize my skills to further work towards personal and professional development and contribute towards the prosperity of the organization.

ACADEMIC QUALIFICATION:

Enrollment Number: 19010349

Branch of Study: Computer Technology

QUALIFICATION	INSTITUTE	YEAR	SCORE/SGPA
B.E. 7th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2022	7.60/10
B.E. 6th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2022	6.50/10
B.E. 5th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2021	7.78/10
B.E. 4th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2021	7.00/10
B.E. 3rd Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2020	7.00/10
B.E. 2nd Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2020	7.50/10
B.E. 1st Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2019	6.50/10

TECHNICAL ACTIVITIES/PROJECTS:

Hand Sign Recognition Using Machine Learning:

Synopsis: Computer application used to capture sign language images from camera, extract hand image from the input image and convert it into text by using CNN for classification of images and display it on screen.

COMPUTER PROFICIENCY

Programming Languages: python, Java, Cpp.

Libraries used: OpenCV, Tensorflow, Keras.

4. Jay Rathod

YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING

(An Autonomous Institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University)

Nikhil Balaji Dapkekar

Mobile: +91- 84079 03157 | Email-id: jayr8473@gmail.com | Date of Birth: 30/01/2000

Address: Pundlikbaba nagar, katora naka, Amravati 444601

OBJECTIVE:

Seeking a position in the field of Computer Science where I can utilize my skills to further work towards personal and professional development and contribute towards the prosperity of the organization.

ACADEMIC QUALIFICATION:

Enrollment Number: 19010298

Branch of Study: Computer Technology

QUALIFICATION	INSTITUTE	YEAR	SCORE/SGPA
B.E. 7th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2022	6.9/10
B.E. 6th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2022	6.87/10
B.E. 5th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2021	6.81/10
B.E. 4th Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2021	6.85/10
B.E. 3rd Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2020	5.79/10
B.E. 2nd Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2020	6.53/10
B.E. 1st Semester	Yeshwantrao Chavan College Of Engineering, Nagpur	2019	5.43/10

TECHNICAL ACTIVITIES/PROJECTS:

Hand Sign Recognition Using Machine Learning

Synopsis: Computer application used to capture sign language images from camera, extract hand image from the input image and convert it into text by useing CNN for classification of images and display it on screen.

COMPUTER PROFICIENCY

Programming Languages: python

Libraries used: OpenCV, Tensorflow, Keras

CHAPTER 7 REFERENCES

[1] G. Pala, J. B. Jethwani, S. S. Kumbhar and S. D. Patil, "Machine Learning-based Hand Sign Recognition," 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), 2021, pp. 356-363, doi: 10.1109/ICAIS50930.2021.9396030.

[2] G. Anantha Rao, K. Syamala, P. V. Kishore, "Deep Convolutional Neural Networks for Sign Language Recognition," Biomechanics and Vision Computing Research Centre, Department of ECE, K.L. University, Green Fields, Guntur (DT), Andhra Pradesh, INDIA.

[3] N. Adaloglou, Theocharis Chatzis, Ilias Papastratis, "A Comprehensive Study on Deep Learning-Based Methods for Sign Language Recognition," in IEEE Transactions on Multimedia, vol. 24, pp. 1750-1762, 2022, doi: 10.1109/TMM.2021.3070438.

[4] Suharjito, H. Gunawan, N. Thiracitta and A. Nugroho, "Sign Language Recognition Using Modified Convolutional Neural Network Model," 2018 Indonesian Association for Pattern Recognition International Conference (INAPR), 2018, pp. 1-5, doi: 10.1109/INAPR.2018.8627014.

[5] Khalil, Khalid & El-Darymli, Khalid & Khalifa, Othman & Enemosah, Hassan. (2006). "Speech to Sign Language Interpreter System (SSLIS)," IEEE International Conference of Computer and Communication Engineering (ICCCE'06).

[6] Ankita Wadhawan and Parteek Kumar, "Sign Language Recognition Systems: A Decade Systematic Literature Review," 2019 CIMNE, Barcelona, Spain 2019

[7] SHILPI KATARIA, AMAN SOHI, SHAMINDER SINGH, THAKUR NILESHSINGH V. , AU2021101012 ," A SYSTEM FOR TRANSLATING SIGN LANGUAGE INTO SPEECH AND VICE-VERSA "(2021)

[8] Kikuo Fujimura, Xia Liu , WO2005114556A2 ," SIGN BASED HUMAN-MACHINE INTERACTION" (2005)

[9] I. H. Yemenoglu, A. F. M. S. Shah and H. Ilhan, "Deep Convolutional Neural Networks-Based Sign Language Recognition System," 2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2021, pp. 0573-0576, doi: 10.1109/IEMCON53756.2021.9623068.

[10] Zhengyu Zhou, Tobias Menne, Kehuang Li, Kui Xu . US20160307469A1 ,"SYSTEM AND METHOD FOR AUTOMATED SIGN LANGUAGE RECOGNITION" (2017)

[11] Anthony Vernon Walker Smith, Alistair Ian Sutherland, Arnaud Lemoine, Sean Mcgrath . USOO6128003A , "HAND GESTURE RECOGNITION SYSTEM AND METHOD" (2018)

[12] "Real Time Sign Language Recognition and Speech Generation**",** Amrita Thakur, Pujan Budhathoki, Sarmila Upreti, Shirish Shrestha, Subarna Shakya; JIIP; 2020

[13] "Sign language recognition using image based hand gesture recognition techniques **",** A. S. Nikam and A. G. Ambekar; IEEE; 2016

[14] A. S. Nikam and A. G. Ambekar, "Sign language recognition using image based hand gesture recognition techniques," 2016 Online International Conference on Green Engineering and Technologies (IC-GET), 2016, pp. 1-5, doi: 10.1109/GET.2016.7916786.

[15] T. D. Sajanraj and M. Beena, "Indian Sign Language Numeral Recognition Using Region of Interest Convolutional Neural Network," 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT), 2018, pp. 636-640, doi: 10.1109/ICICCT.2018.8473141.

[16] N. Soodtoetong and E. Gedkhaw, "The Efficiency of Sign Language Recognition using 3D Convolutional Neural Networks," 2018 15th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), Chiang Rai, Thailand, 2018, pp. 70-73, doi: 10.1109/ECTICon.2018.8619984.

[17] "Sign Language Recognition using Deep Learning", AUG 2022, https://towardsdatascience.com/sign-language-to-text-using-deep-learning-7f9c8018c593

[18] Triyono, Liliek & Pratisto, Eko & Bawono, (2018). "Sign Language Translator Application Using OpenCV," IOP Conference Series: Materials Science and Engineering. 333. 012109. 10.1088/1757-899X/333/1/012109.

[19] Mohammed Waleed Kalous, Machine recognition of Auslan signs using PowerGloves: Towards large-lexicon recognition of sign language.

[20] Sahoo Ashok, Mishra Gouri and Kiran Ravulakollu, "Sign language recognition: State of the art" ,ARPN , 2016

[21] Number System Recognition (https://github.com/chasinginfinity/number-sign-recognition)

[22] https://opencv.org/

[23] https://en.wikipedia.org/wiki/TensorFlow

[24] https://en.wikipedia.org/wiki/Convolutional_neural_nework