**SIGN LANGUAGE RECOGNITION USING**

**CONVOLUTIONAL NEURAL**

**NETWORK**

**A Report Submitted to**

CHHATTISGARH SWAMI VIVEKANAND

TECHNICAL UNIVERSITY BHILAI (C.G.), INDIA

*In fulfillment*

***For the Award of the Degree of***

Bachelor of Engineering

**In**

Computer Science and Engineering

By

**Name- Sakshi Likhar**

**Enrolment no: - BA9388**

**Name- Sanika Awasthi**

**Enrolment no: - BA9549**

**Name- Poonam Sahu**

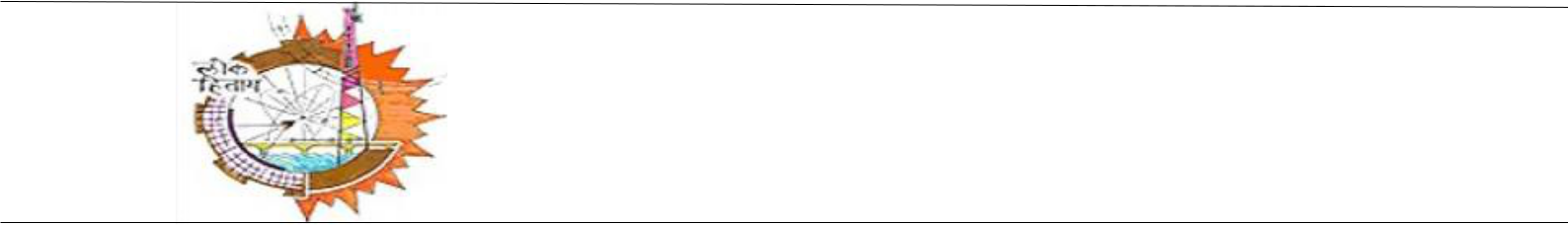
**Enrolment no: - BA7419**

**Bachelor of Engineering**

**Under the Guidance of**

**Ms. Monika Sahu**

**Assistant Professor**



**Department of Computer Science & Engineering Government Engineering College Sejbahar, Raipur (c.g.)**

Session 2020-2021

**DECLARATION BY THE CANDIDATE**

I, the undersigned, solemnly declare that the report of the Report work entitled “SIGN LAGUAGE RECOGNITION Using Convolution Neural Network” is based on my own work carried out during the course of study under the supervision of **Assistant Professor Miss Monika Sahu.**

I assert that the statements made and conclusion drawn are an outcome of my project work. I further declare that to the best of my knowledge and belief, the report does not contain any part of any work which has been submitted for the award of Bachelor of Engineering degree or any other degree/diploma/certificate in this university and any other university of India or abroad.

(Signature of the candidate)

**Name: - Sakshi Likhar**

**Roll No: - 301602216048**

**Enrollment No: BA9388**

(Signature of the candidate)

**Name: - Poonam Sahu**

**Roll No: - 301602216033**

**Enrollment No: BA7419**

(Signature of the candidate)

**Name: - Sanika Awasthi**

**Roll No: - 301602216050**

**Enrollment No: BA9549**

**CERTIFICATE OF THE SUPERVISOR**

This is to certify that the work incorporated in the Report **“SIGN LAGUAGE RECOGNITION** **USING CONVOLUTIONAL NEURAL NETWORK”**, is a record of project work carriedout by **Sakshi Likhar** bearing **Enrollment No**. **BA9388**, **Sanika Awasthi** bearing **Enrollment No. BA9549** and **Poonam Sahu** bearing **Enrollment No. BA7419** under my guidance and supervision for the award of Degree of Bachelor of Engineering in Computer Science and Engineering of Chhattisgarh Swami Vivekanand Technical University, Bhilai (C.G.), India.

To the best of my knowledge and belief the Report

1. Embodies the work of the candidate herself/himself.
2. Has duly been completed.
3. Fullfills the requirement of the Ordinance relating to the Bachelor of Engineering Degree of the University, and
4. Is up to the desired standard growth in respect of contents and language for being referred to the examiners.

|  |  |
| --- | --- |
| (Signature of Supervisor) | (Signature of H.O.D) |
| **Asst. Prof. Ms .Monika Sahu** | **Prof. Dr.R.H.Talwekar** |
| Deptt. Of Computer Science & Engineering, | Deptt. Of Computer Science & Engineering, |
| Government Engineering College, | Government Engineering College, |
| Sejbahar, Raipur (C.G.) | Sejbahar, Raipur (C.G.) |

**Forwarded to Chhattisgarh Swami Vivekanand Technical University, Bhilai**

(Signature of Principal)

**Dr. R.S. Parihaar**Government Engineering College,Sejbahar, Raipur (C.G.)

**CERTIFICATE BY THE EXAMINERS**

This Report entitled **“SIGN LAGUAGE RECOGNITION USING CONVOLUTIONAL** **NEURAL NETWORK”**, submitted by **Sakshi Likhar** bearing **Enrollment No. BA9388**, **Sanika Awasthi** bearing **Enrollment No. BA9549** and **Poonam Sahu** bearing **Enrollment No. BA7419** has been examined by the undersigned as a part of the examination and is hereby recommended for the award of the Degree of Bachelor of Engineering in Computer Science and Engineering of Chhattisgarh Swami Vivekanand Technical University, Bhilai (C.G.), India.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Internal Examiner

Date:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

External Examiner

Date:

**ACKNOWLEDGEMENT**

I would like to thank my guide **Asst. Prof. Miss. Monika Sahu**, Department of Computer Science and Engineering for her immense support and enlightened guidance for my project which we have developed as B.E. student. I am very grateful for the inspiring discussions with all my faculties their valuable support and path-guiding suggestions have helped me to develop this project. I am thankful to **Prof. Dr. R.H. Talwekar**, HOD (CSE), for giving thoughtful suggestions during my work and for their inspiration and constant encouragement that enabled me to present my work in this form.

I would like to take this opportunity to express my thanks towards everyone in the Computer Laboratory of Government Engineering College, Sejbahar, and Raipur (C.G.).

I specially thank my colleagues, batch mates, and friends for their support and help that they offered me during the course of development of this project.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Signature of the candidate)

**Name: - Sakshi Likhar**

**Roll No: - 301602216048**

**Enrollment No: BA9388**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Signature of the candidate)

**Name: - Poonam Sahu**

**Roll No: - 301602216033**

**Enrollment No: BA7419**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Signature of the candidate)

**Name: - Sanika Awasthi**

**Roll No: - 301602216050**

**Enrollment No: BA9549**

**ABSTRACT**

There is an undeniable communication problem between the Deaf community and the hearing majority. Innovations in automatic sign language recognition try to tear down this communication barrier. Our contribution considers a recognition system using convolutional neural networks (CNNs). Instead of constructing complex handcrafted features, CNNs are able to automate the process of feature construction. A Convolutional network has a benefit over other Artificial Neural networks in extracting and utilizing the features data, enhancing the knowledge of 2D shapes with higher degree of accuracy and unvarying to translation, scaling and other distortions.

The data set is a collection of images of alphabets from the American Sign Language(ASL), separated in 29 folders which represent the various classes. The training data set contains 87,000 images which are 200x200 pixels. There are 29 classes, of which 26 are for the letters A-Z and 3 classes for SPACE, DELETE and NOTHING.

These 3 classes are very helpful in real time applications, and classification.

The test data set contains a mere 29 images, to encourage the use of real world test images.

The images are passed into input layers and then into the hidden layers which contain three sets of convolutional, activation and pooling layers. Then finally it is mapped onto the fully connected layer and given a softmax classifier to translate the images.

Our project is basically been designed to help the deaf and dumb people in communication or interaction with other people. This system can be very effective and can be used in the near future for blind people for detecting and recognizing the object that lies in front of the blind. The system is able to recognize one handed sign representation of the standard alphabets (A-Z) and numeric values (0-9). In our project we used OpenCV for gesture capture and machine learning algorithms to train and recognize the gesture. Our project is fully python based. The only requirement of this project is a webcam that captures the gestures and a good computer.

**LIST OF FIGURES**

Figure 1- Gesture Recognition… 17

Figure 2-Hand Gloves Recognition 19

Figure 3- System Architecture … 20

Figure 4- Preprocessing 22

Figure 5- Feature extraction 26

Figure 6- Activity Diagram 28

Figure 7- Set hand histogram 31

Figure 8-process of cnn 32

Figure 9- Set hand histogram result………………………………………………..32

Figure 10- Hand gesture input 34

Figure 11- Final accuracy of the trained model 35

Figure 11- Gesture to text 35

**TABLE OF CONTENT**

**Abstract** **6**

**List of Figures** **7**

**Chapter** **Title** **Page No.**

**1.** **INTRODUCTION** **9**

**1.1** **What is gesture recognition** **11**

**1.2** **Objective** **11**

**2.** **LITERATURE SURVEY** **13**

**2.1** **Problem Statement** **14**

**2.2** **Existing System** **14**

**2.3** **Proposed System** **15**

**3.** **Requirement Specification** **24**

**3.1** **Hardware Requirements** **25**

**3.2** **Software Requirements** **25**

**4.** **METHODOLOGY** **27**

**4.1** **System Architecture** **28**

**4.2** **Preprocessing** **28**

**4.3** **Feature Extraction** **28**

**4.4** **Activity Diagram** **28**

**4.5** **Implementation Module** **28**

**5.** **RESULTS AND CONCLUSION** **31**

**5.1** **Intermediate Result** **34**

**5.2** **Conclusion** **35**

**5.3** **Future Enhancement** **28**

***CHAPTER – I***

***INTRODUCTION***

**What is Gesture Recognition?**

Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Gestures can originate from any bodily motion or state but commonly originate from the face or hand.

Gesture recognition technology has been considered to be the highly successful technology as it saves time to unlock any device. Gesture recognition can be conducted with techniques from computer vision and image processing.

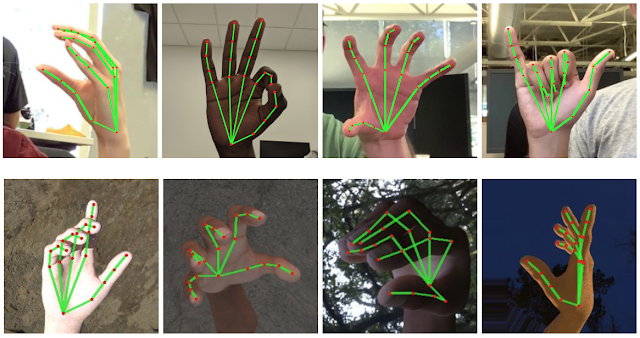


Fig 1.1: Gesture Recognition

* 1. **OBJECTIVE**

Very few people understand sign language. Moreover, contrary to popular belief, it is not an international language. Obviously, this further complicates communication between the Deaf community and the hearing majority. The alternative of written communication is cumbersome, because the Deaf community is generally less skilled in writing a spoken language. Furthermore, this type of communication is impersonal and slow in face-to-face conversations. For example, when an accident occurs, it is often necessary to communicate quickly with the emergency physician where written communication is not always possible. The purpose of this work is to contribute to the ﬁeld of automatic sign language recognition. We focus on the recognition of the signs or gestures. There are two main steps in building an automated recognition system for human actions in spatio-temporal data. The ﬁrst step is to extract features from the frame sequences. This will result in a representation consisting of one or more feature vectors, also called descriptors. This representation will aid the computer to distinguish between the possible classes of actions. The second step is the classiﬁcation of the action. A classiﬁer will use these representations to discriminate between the diﬀerent actions (or signs). In our work, the feature extraction is automated by using convolutional neural networks (CNNs).

The main objective is to translate sign language to text. The framework provides a helping-hand for speech-impaired to communicate with the rest of the world using sign language. This leads to the elimination of the middle person who generally acts as a medium of translation.

In brief, the aim of the project is to develop a system capable of recognizing the gestures given by the user and convert the gesture into text

***CHAPTER – II***

***LITERATURE SURVEY***

**2.1 Problem Statement**

Communication is a fundamental necessity in every social scale.There is a need to solve language barriers. Most research and developments focus on verbal language translation This puts the deaf community at a disadvantage.

A working model that takes hand gesture as an input, recognize the gesture and convert it to a text and further convert the generated text to speech.

**2.2 Existing Systems**

2.2.1Monochrome Glove: A Robust Real-Time Hand Gesture Recognition Method by using a Fabric Glove with Design of Structured Markers[2]

AUTHORS: Hidetoshi Ishiyama, Shuichi Kurabayashi

This project presents a method for recognizing human-hand postures in real time, even if environment light cannot be configured appropriately. This project gave us the idea required in the making of our project. Here, the key technology is a monochrome glove that is patterned with augmented reality (AR) markers on its palm and is also designed with a structured marker on each finger as shown in the figure 2.2.1 This project acted as one of the catalyst in the making of our project as it gave us the idea of using hand signs. As the glove only uses white color for the design of the patterns, it can achieve robust gesture recognition in a natural lighting environment by using a single camera to track a hand wearing the glove.



Fig 2.2.1: Hand Gloves Recognition

2.2.2 Machine learning for hand gesture recognition using bag-of-words AUTHOR: Marouane Benmoussa, Abdelhak Mahmoudi[5]

Human Computer Interaction received a great deal of attention this last decade. Last researches have turned to more natural interaction systems like gestural human machine interfaces. However, few of them are taking into account mandatory requirements to apply the workflow of a learning model, mainly data unbalance, model selection and generalization performance metric choice. In this work, we proposed a machine learning method for real time recognition of 16 gestures of user hands using the Kinect sensor that respects such requirements. The recognition is triggered only when there is a moving hand gesture. The method is based on the training of a Support Vector Machine model on hand depth data from which bag of words of SIFT and SURF descriptors are extracted. The data was kept balanced and the model kernel and parameters were selected using cross validation procedure. The method achieved 98% overall performance using the area under the ROC curve measure.

**2.3 Proposed System**

We have come up with the concept that consist of both the hand gesture recognition and American Sign Language to make a computer application which recognizes the Sign Language according to the gesture given in the web camera in the computer system and provide output as a text.

In our implementation we have trained the network in such a way that it learns many filters of size 5 by 5 and then pass it through a ReLU activation function followed by 5 by 5 max pooling in both dimensions. We then take the output of Maxpooling layers to apply it to fully connected layers. Our fully connected layers contain around 500 units which we will pass through another ReLU activation that enables us to combine them into classes , which are useful for identifying our image.

***CHAPTER – III***

***REQUIREMENT SPECIFICATION***

**3.1 Hardware requirements**

**3.1.1 Web camera:**

A webcam is a video camera that feeds or streams its image in real time to or through a computer to a computer network. When "captured" by the computer, the video stream may be saved, viewed or sent on to other networks via systems such as the internet, and emailed as an attachment.

**3.2 Software requirements**

**3.2.1 Python:**

Python is an interpreted high-level programming language for general-purpose programming. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library. Python interpreters are available for many operating systems.

**3.2.2 Keras:**

Keras is a high-level neural networks API, written in Python and capable of running on top of Tensor Flow, CNTK, or Theano. It was developed with a focus on enabling fast experimentation. Being able to go from idea to result with the least possible delay is a key to doing good research. Using Keras for deep learning library that:

•Allows for easy and fast prototyping (through user friendliness, modularity, and extensibility).

•Supports convolution networks and recurrent networks, as well as combinations of the two.

•Runs seamlessly on CPU and GPU.

**3.2.3 Tensorflow:**

it is an [open-source](https://en.wikipedia.org/wiki/Open-source_software) [software library](https://en.wikipedia.org/wiki/Library_(computing)) for [dataflow programming](https://en.wikipedia.org/wiki/Dataflow_programming) across a range of tasks. It is a symbolic math library, and is also used for [machine learning](https://en.wikipedia.org/wiki/Machine_learning) applications such as [neural networks](https://en.wikipedia.org/wiki/Neural_networks)

**3.2.4 OpenCV:**

OpenCV (Open Source Computer Vision Library) is released under a BSD license and hence it’s free for both academic and commercial use. It has C++, Python and Java interfaces and supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications. Written in optimized C/C++, the library can take advantage of multi-core processing. Enabled with OpenCL, it can take advantage of the hardware acceleration of the underlying heterogeneous compute platform.

Adopted all around the world, OpenCV has more than 47 thousand people of user community and estimated number of downloads exceeding 14 million. Usage ranges from interactive art, to mines inspection, stitching maps on the web or through advanced robotics.

**3.2.5 SQLite:**

SQLite is an in-process library that implements a self-contained, serverless, zeroconfiguration, transactional SQL database engine. The code for SQLite is in the public domain and is thus free for use for any purpose, commercial or private. SQLite is the most widely deployed database in the world with more applications than we can count, including several high-profile projects.

***CHAPTER-IV***

***METHODOLOGY***

**4.1 System Architecture:**

System architecture is a conceptual model that defines the structure, behavior, and more views of the system. An architectural description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system. The fig.1 shows the basic architecture.

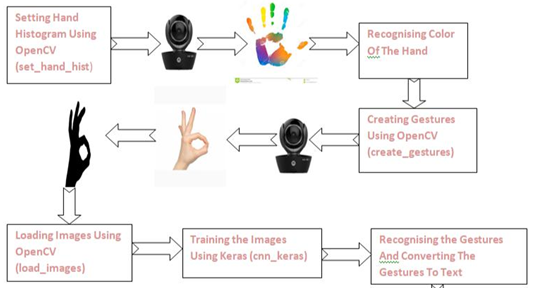


Fig 4.1: System Architecture

**4.2 Pre-processing:**

As these images are not taken in a controlled lightening environment also images are taken with a digital camera, they have different sizes and different resolutions. So in image pre-processing is required. Local changes due to noise and digitization errors should not radically alter the image scene and information. In order to satisfy the memory requirements and the environmental scene conditions, pre-processing of the raw video content is highly important. Various factors like illumination, background, camera parameters, and viewpoint or camera location are used to address complexity of signs. The first most step of preprocessing block is filtering. A moving average or median filter is used to remove the unwanted noise from the acquired image. Background subtraction forms the next major step in the pre-processing block. Running Gaussian average method is used in order to obtain the background subtraction as it is very fast and consumes low memory when compared to other methods. This takes into consideration of the illumination changes like lightning, camera motion changes etc.

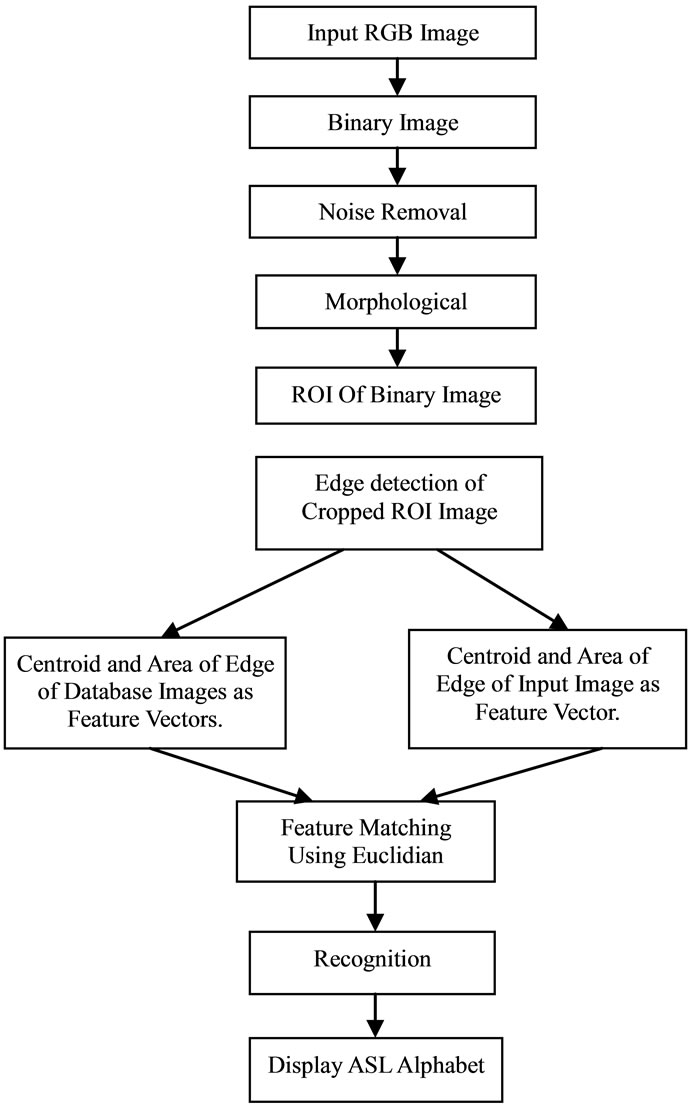


Fig 4.2: Preprocessing

**4.3 Feature Extraction:**

Features are the crucial elements for sign language recognition. Feature extraction reduces the computational time without sacrificing the accuracy. The appearance of the gesture being recognized depends on the position of the camera, distance of the signer from the camera etc. These methods have to maintain a balance between accuracy and computational complexity during the real time performance.

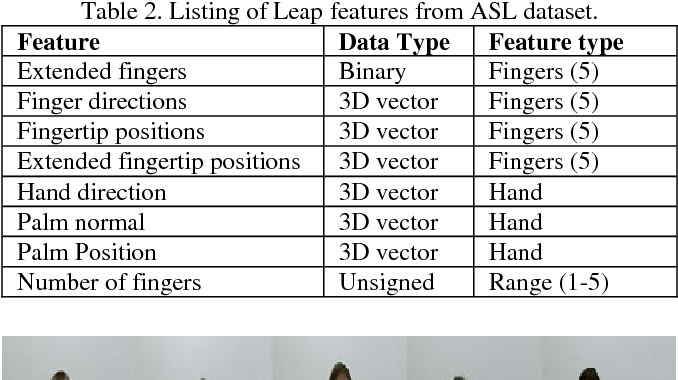


Fig 4.3: Feature Extraction

**4.4 Activity Diagram**

Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. In Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. The control flow is drawn from one operation to another. Fig4.4 shows the operation and the control of the system.

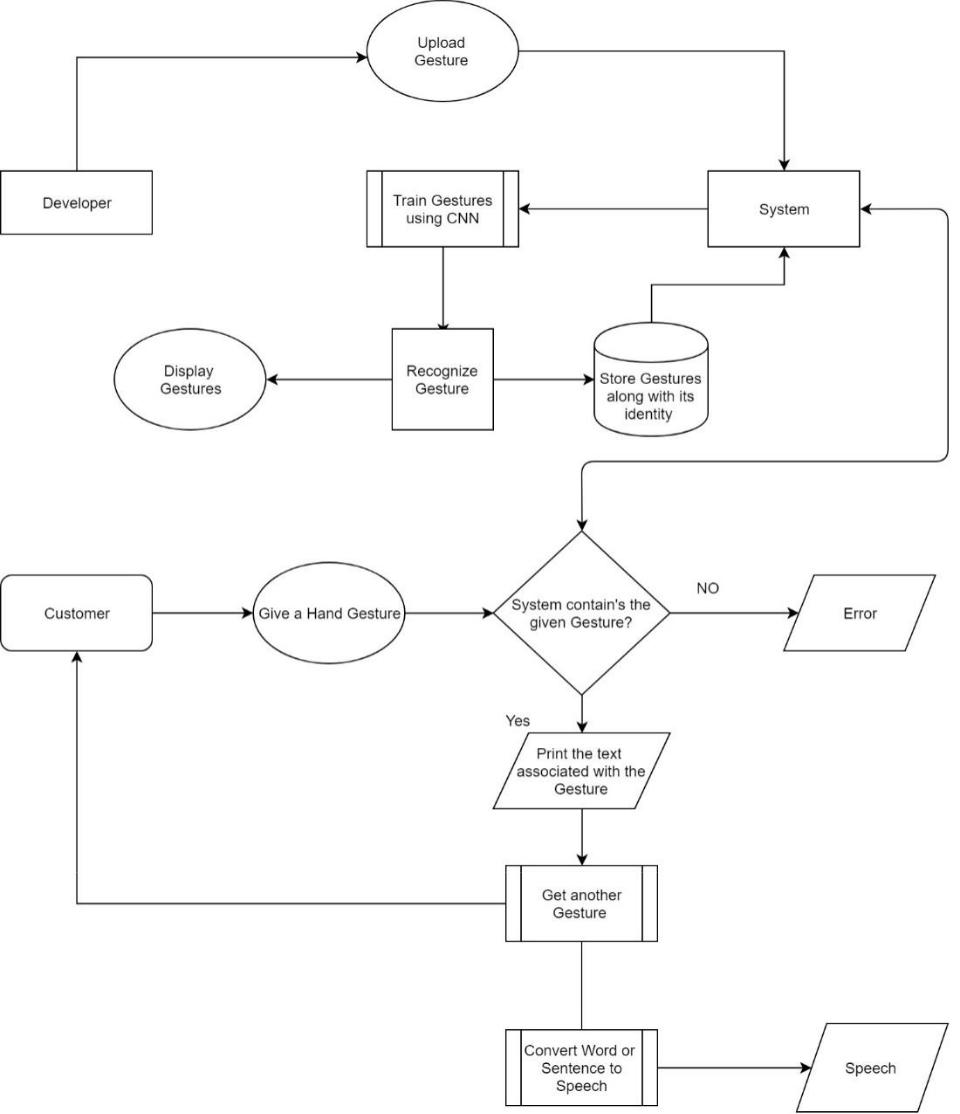


Fig 4.4: Activity Diagram

**4.5 Implementation Module**

**4.5.1 Setting hand histogram:**

An image histogram is a type of histogram that acts as a graphical representation of the tonal distribution in a digital image. It plots the number of pixels for each tonal value. By looking at the histogram for a specific image a viewer will be able to judge the entire tonal distribution at a glance. Image histograms are present on many modern digital cameras. Photographers can use them as an aid to show the distribution of tones captured, and whether image detail has been lost to blown-out highlights or blacked-out shadows. This is less useful when using a raw image format, as the dynamic range of the displayed image may only be an approximation to that in the raw file. Image editors typically have provisions to create a histogram of the image being edited. The histogram plots the number of pixels in the image (vertical axis) with a particular brightness value (horizontal axis). Algorithms in the digital editor allow the user to visually adjust the brightness value of each pixel and to dynamically display the results as adjustments are made. Improvements in picture brightness and contrast can thus be obtained.

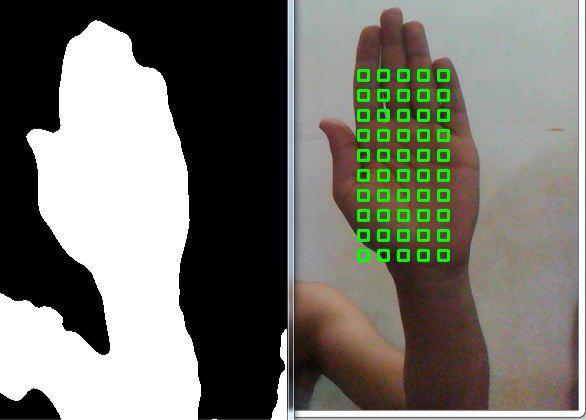


Fig 4.5.1: Set Hand Histogram

**4.5.2 Create and capture gestures:**

On starting executing this program, you will have to enter the gesture number and gesture name/text as shown in figure Then an OpenCV window called "Capturing gestures" which will appear as shown in [figure ] In the webcam feed you will see a green window (inside which you will have to do your gesture) and a counter that counts the number of pictures stored.

**4.5.3 Loading the images for training:**

Prior to training we have stored 1200 images for each and every gesture in our ‘gestures’ folder. These images need to be fed to our training model. In order to train our model, we need to load these images. After loading a pickle dump is performed. This generates four files i.e. test\_images, test\_labels, train\_images, train\_labels.

**4.5.4 Training the dataset:**

The gestures that is stored in the training dataset, after loading is ready to be trained for recognition. This model uses Keras for training the model. We use CNN (Convolutional Neural Network). Convolution neural networks (CNNs) have been widely adopted in the field of computer vision. These models are based on convolution operations applied to the input image at multiple hierarchical layers. CNNs are very powerful because they can be trained end-to-end in a supervised manner and thus obviate the need to manually devise features, and have substantially outperformed the state-of-the-art for classification of natural images on large and well established databases. In medical image analysis, CNNs are also increasingly used. Their capability to learn a complex, hierarchical representation of the data makes CNNs useful to discern the complex disease specific patterns, difficult to be encoded by humans and by simpler traditional classifiers. Recent works on cancer detection and brain segmentation have shown CNN achieved remarkable performance. CNN training process is a sequential process requiring many iterations (or epochs) to optimize the network parameters and learn discriminative features. In every epoch, a subset of samples is randomly selected from the training data and is presented to the network to update its parameters through back-propagation, minimizing a cost function.

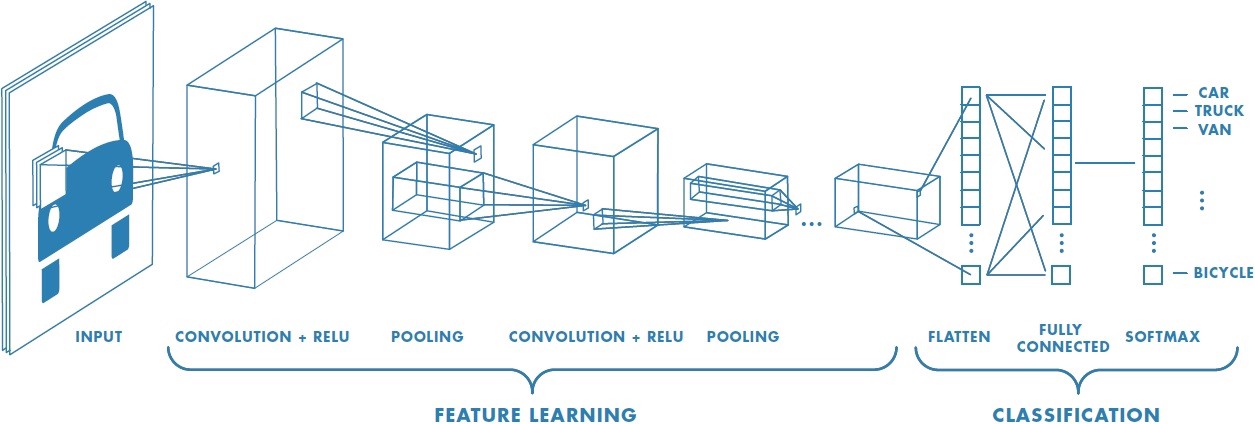


Fig 4.5.4: Process CNN

**4.5.5 The gesture and converting to text:**

After the training is completed, the model is now ready to recognize or predict the given input gesture which is the main motivation for training. This module will predict the gesture provided as n input to the model and the gesture in the training dataset. Using the positions of fingers and palm dimensions, we model our hand Then we compare the model with a dictionary Gestures defined to determine presence of gesture.

***CHAPTER – III***

***RESULT AND CONCLUSION***

**5.1 Result:**

We have successfully built a system that takes in gestures as an input, converts the gesture provided into text and further the text into speech. In this chapter we will present the results that we have got from the system and their snapshot of the same. The project consists of different modules, so we show the results for each model.

**5.1.1 Set hand histogram:**

The first module works perfectly fine. This model is to detect the colour of the object to be recognized. Figure 5.1.1 shows Colour detention pattern of the hand given.

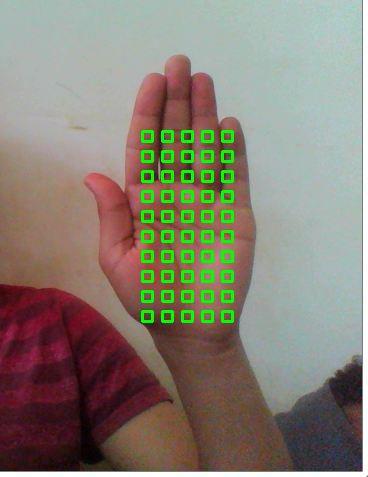
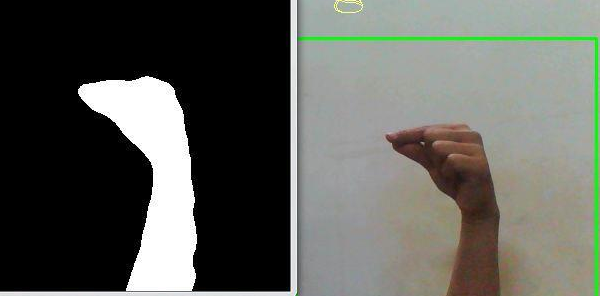


Fig 5.1.1: Set Hand Histogram

**5.1.2 Creating the gestures:**

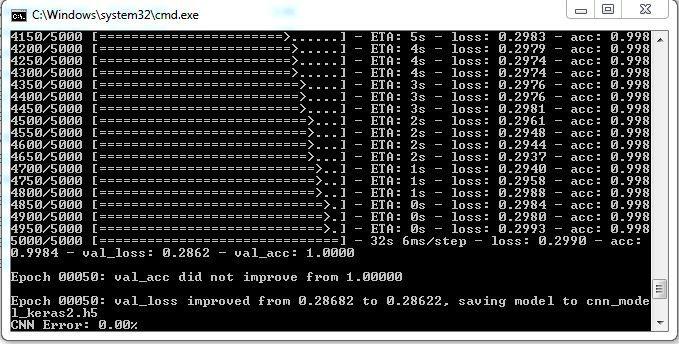
In this module, we capture images to train for the model. The recognition of gesture uses the captured 1200 photos of the gesture to compare. In figure 5.1.2 we have given a hand gesture to store that gesture in the system.



**Fig 5.1.2: Hand gesture input**

**5.1.3 Training the model:**

In the figure 5.1.3 the gestures given above are trained accordingly.



**Fig 5.1.3:Trained model**

**5.1.4 Recognizing gestures:**

This is the gesture recognition module. This module accepts the gestures and converts the given gesture that has been captured and trained into text and text further into speech. We can see in the figure5.1.4 how the sign has been converted to text.

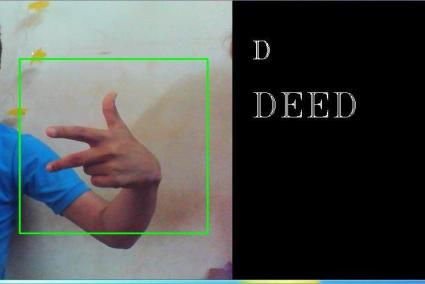


Fig 5.1.4: Gesture to text (DEED)

**5.2 Conclusion**

In this project, we presented the concept of gesture to text conversion concept, due to which the communication between the impaired people of the society and the common people will be carried out without any obstruction. The explanation of the design and implementation is presented along with the prototype which captures the gesture of the American Sign Language. As compared to the other system this concept not only focuses on the gesture to word display but also on the speech synthesis. The use of Machine Learning shows that the more number of training datasets the more efficient the gesture capturing and speech synthesizing is performed.

**5.3 Future Enhancements**

The concept of gesture to speech conversion concept is introduced, due to which the communication between the vocally impaired people of the society and the common people will be carried out without any obstruction. Thus we tried to bring more precise gesture capturing process along with fast reply. Not only that but this system can be adapted on various Smartphone’s in the form of application, so that it is accessible to everyone.

This system can further be enhanced to help blind as well. Similar to recognizing the gestures, the system can also recognize the object that lies in front of them if they are trained properly. So if integrated in a blind stick or the glass, this system could recognize object that lies ahead of them and warn about the object being there.