Real-Time Communication System Powered by AI for Specially Abled

Submitted by :

TEAM ID:- PNT2022TMID26077

SANJAY P (2019PITEC254 PIT)

SANJAIRAJ M (2019PITEC252 PIT)

RUPESH V (2019PITEC247 PIT)

RITHWIN PRASEED (2019PITEC246 PIT)

Real-Time Communication System Powered by AI for Specially Abled

Abstract

In this paper, we discuss a stand-alone technology that would make it simple and fluid for hearing-impaired and normal individuals to converse with one another. We provide an application for automatically translating visual data into text in real time while using image processing to recognise American Sign Language. Video footage from a digital camera or camera application will be used to create a real-time hand gesture detection system after which the hand position and location will be tagged and isolated via cropping. The hand motions will then be identified by image processing and compared to a gesture database that has already been created, which will be utilised for text conversion on the screen. Additionally, the programme allows regular users to write the text down and exhibit the corresponding animation of hand motions. This system does textual representation and real-time recognition of American Sign Language, producing more accurate results in the shortest amount of time. It won't just help the specially abled; it may also be applied in a number of different technological contexts.

Additionally, this method gives users the freedom to study American Sign Language at their own speed, whenever they want, anywhere—at home or at work.

1. Introduction

• Project Overview

By discussing their ideas, opinions, and experiences with others around them, people come to know one another. There are several methods to do this, but the gift of "Speech" is the finest. Speech allows everyone to communicate their ideas and comprehend one another quite well. It would be unfair to ignore those who are denied this wonderful gift: individuals with disabilities. In these circumstances, sign language has traditionally been used using the human hand.

The most common issue for those with hearing/speech impairments is being unable to communicate with others. They utilise Sign Language to communicate with others in order to express their thoughts or feelings (SL). Sign language (SL) is a prominent way of communication mechanism utilised regularly by persons who are deaf or hard of hearing. This nonverbal language employs hand motions as well as occasional face gestures.

With the advancement of technology, some form of device or instrument that can mediate between hard-of-hearing people and normal people is necessary, so that they may easily interact with each other without the need for a third person as an interpreter.

• Purpose

In order to communicate with regular people, the project intends to create a system that translates sign language into text that is legible by humans. A convolution neural network is being used to build a model that is trained on various hand motions. A web application utilising this concept is created. With the use of this software, persons who are deaf or dumb may communicate using signs that are translated into language that is intelligible to others

2. Literature Survey

• Existing Problem

There are three main ways that hearing loss might impact a person:

1. Less access to school and employment owing to communication problems

2. social withdrawal brought on by limited access to services and communication challenges

3. emotional issues brought on by a decline in confidence and self-worth.

Points to keep in mind

- ✓ At least 700 million individuals will need hearing rehabilitation by 2050, when it is predicted that approximately 2.5 billion people would have some degree of hearing loss.
- ✓ Due to dangerous listening habits, almost 1 billion young individuals are at risk of developing permanent, preventable hearing loss.

- **Existing Solutions**

Few existing solutions for these kind of problems are as follows

- *The use of technology*

Technology, such as laptop or smart phone, is one of the simplest methods to connect. Both a hearing person and a deaf person can type out what they wish to communicate. Additionally, a blind person can use voice recognition software to translate their speech into writing, which a Deaf person can subsequently read.

- *The use of interpreter*

If the deaf person is fluent in sign language and a sign language interpreter is present, communication is made easier. Through the interpretation, the deaf person and the blind person may converse. The interpreter can say what has been stated to the blind person after the deaf person has used sign language, and they can also translate any spoken words from the blind person into sign language for the deaf person.

- **References**

1. TITLE:
Innovative study of an AI voice based smart device to assist deaf people

AUTHOR: Dhaya Sindhu Battina

YEAR:
2021 Assistive technology consists of a wide range of hardware and software tools that enable a person to receive information in the format that suits their needs best. These Various technology may be available to the deaf.. . Many items, including cochlear implants, loop systems,accessibility, FM technology, and assistive listening devices,visual warning systems, videophones, and much more . Recognizing the worth and boundaries of different assistive devices can be advantageous for both. Artificial intelligence (AI) enables computers to learn from existing experiences, adapt to new information, and perform tasks that are similar to those carried out by humans . The vast majority of artificial intelligence applications that users know of today – ranging from chess playing robots to self-driving vehicles – are primarily reliant on deep learning and computational linguistics.Computers may be taught to do particular jobs by

processing huge quantities of data and detecting trends in the data. This is accomplished via the use of various technologies Machine learning technologies have the potential to substantially reduce the communication obstacles that deaf or hearing-impaired persons have when interacting with other groups, thus promoting social inclusion for these individuals. Recent advancements in both sensing technology and artificial intelligence algorithms have opened the way for the creation of a broad array of applications aimed at meeting the requirements of the deaf and hearing-impaired populations.

2. TITLE:
Communication system for deaf and dumb people

AUTHOR: Shraddha R. Ghorpade, Prof. Surendra K. Waghmare2

YEAR:
2019 People with disabilities are having a difficult time keeping up with the rapidly evolving technology, which is one of the major issues that our society is dealing with. For those with disabilities, having access to communication tools has become crucial. typically deaf and stupid people use sign language to communicate, but they struggle to do so with non-sign language users language. Information is the main topic of communication between normal and deaf individuals using sign language, which is expressive and natural. So that we can converse with them and comprehend what they're saying, we need a translation. A language translation technology converts common sign language into voice, enabling regular people to communicate with one another. When it comes to communicating with other people,sign language (SL) is the primary method of communication for hearing-impaired individuals and other groups. It is conveyed via both manual (body and hand movements) and non-manual (face expressions) characteristics. All of these characteristics are combined to create utterances that communicate the meaning of words or statements . Understanding and being able to record and comprehend the relationship between utterances and words is critical for the Deaf community's ability to lead us toward a time when automated translating between utterances and words is possible .In recent years, researchers have recognized the need for the development of sign language technology to assist hearing-impaired individuals in communicating and socially integrating into their communities. Even though the advancement of such innovations can be extremely difficult owing to the inclusion of multiple sign languages and a scarcity of large annotated datasets, recent advances in artificial intelligence and machine learning have made significant strides towards automating and improving such systems. Keywords: Image Processing, Human Computer Interface (HCI).

3. TITLE:
Educational Status of Differently Abled Persons and Developed Policies in India

AUTHOR: Chiranjit Majumder

YEAR:
2019 April One of the socially created phenomenon is basically Disability. The fact is that many children and adults suffered from disabilities excluded from mainstream education benefits.

Disabled persons are segregated from education system because of social negligence and absence of support system in the home and inadequacy of sufficient facilities in schools particularly. However, education is the most important medium for social, economic and political transformation. Socialization of children with disabilities (CWD) through education receives an unremarkably important roles in societies such as India where social exclusion of Physically Challenged Persons (PCPs) is significant. Indisputably, the literacy level of Physically Challenged Persons (PCPs) is very low in India. Very poor educational outcomes for children with disabilities remain in developing countries specially. Most of disabled persons do not get the full benefits of education. However, some policies in India has started to display some concern for Physically Challenged students. Education is utmost significant to lift up the socio-economic status of PCPs. But education of disabled persons has not received adequate intentness and resources that it requires. Physically Challenged Persons (PCPs), few who are enrolled in schools are not given equal opportunity for middle secondary and higher education levels. Many Disabled persons are educated but they do not get any work for earning in our society. However, in India the existing situation began to change. Indian policies has started to understand as for all people that education is essential for children and adults with IJTSRD21762 International Journal of Trend in Scientific Research and Development (IJTSRD) @ www.ijtsrd.com eISSN: 2456-6470 @ IJTSRD | Unique Paper ID - IJTSRD21762 | Volume – 3 | Issue – 3 | Mar-Apr 2019 Page: 342 disabilities in itself and helpful for participating in employment and other sites of social activity. The Ministry of Human Resource Development (MHRD) has initiated various programmes to give educational opportunities to PCPs in an environment that is inclusive (Ghoshal S.K., 2018). Government of India also understands the needs of appropriate vocational training skills to make them self faithful and productive members of society. But, the scheme coverage has stayed limited. Non Governmental Organizations (NGOs) has a biggest role to improve the life of disabled persons in our society. Educational Status of Disabled Persons in India India is the world's largest democracy. India has a countless challenges for ensuring access to education for over all 200 million (20 Cr) children aged 6 to 13 years. As per 2011 National Census, 1.05% of school going children have a problem like disability (2.13 million = 21 Lakhs 30 Thousand); of these 28% (5 Lakhs 88 Thousand) are not accessing school. Particularly 44% of disabled children are not accessing school have complex and multiple forms of activity limitations and functioning difficulties (Bakhshi et al, 2017). Education is an fruitful tool for socio-economic empowerment which can develop the career of specially disabled ones. disabilities in itself and helpful for participating in employment and other sites of social activity. The Ministry of Human Resource Development (MHRD) has initiated various programmes to give educational opportunities to PCPs in an environment that is inclusive (Ghoshal S.K., 2018). Government of India also understands the needs of appropriate vocational training skills to make them self faithful and productive members of society. But, the scheme coverage has stayed limited. Non Governmental Organizations (NGOs) has a biggest role to improve the life of disabled persons in our society. Educational Status of Disabled Persons in India

- **Problem Statement**

The solution to the issue that the speech and hearing challenged encounter is outlined in this research. The goal of the research is to create a system that narrows the communication gap between speech- and hearing-impaired people and the rest of society.

3. Requirements

Functional Requirements:
Following are the functional requirements of the proposed solution.

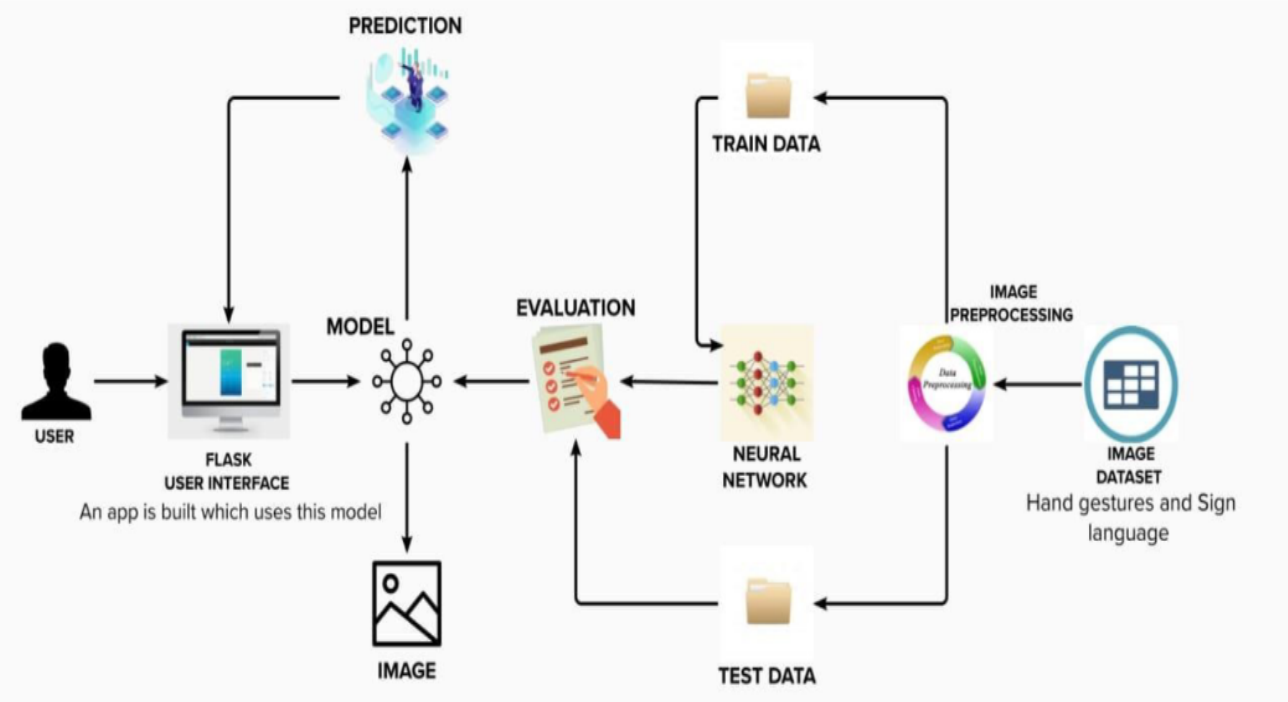
FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Requirement	Converting sign language into speech that can be understand by normal people using an application.
FR-2	User Registration	Manual Sign up using the application or Gmail.
FR-3	User Confirmation	OTP authentication through phone messages, email, notices, paper and confirmation.
FR-4	Product Implementation	Install the dataset to recognise and translate hand gestures and voice for the real-time communication by using the application.
FR-5	Payment Option	Bank transfer, Debit cards, UPI method, if pro version required.
FR-6	Product Feedback	Through the application, phone conversation and Gmail.

Non-functional Requirements:
Following are the non-functional requirements of the proposed solution.

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It is used to describe the application and easy to access the application with the guidelines.
NFR-2	**Security**	It ensures the security of the application by building a firewall and two step verification support. Accessed only by authorised person by given user ID and password or OTP verification.
NFR-3	**Reliability**	To maintain the application conditions and update the version of the application. System update and software update are possible to increase various features and durability based on technology.
NFR-4	**Performance**	This application collects the datasets of hand gestures to provide accurate prediction. Using this method, we can communicate easily at anytime. This application is user friendly and can be accessed by both specially abled and normal people.
NFR-5	**Availability**	Depending on the requirements of the user, all required functions will be offered. When the user requests any features, the features are made available in places where users like to know about it.
NFR-6	**Scalability**	As based on application, real-time communication is accessed on a compatible devices. The application is based on voice conversion system, hand gesture recognition and translation.

4. Project Design

- **Data Flow Diagrams**



- **Block Diagram**

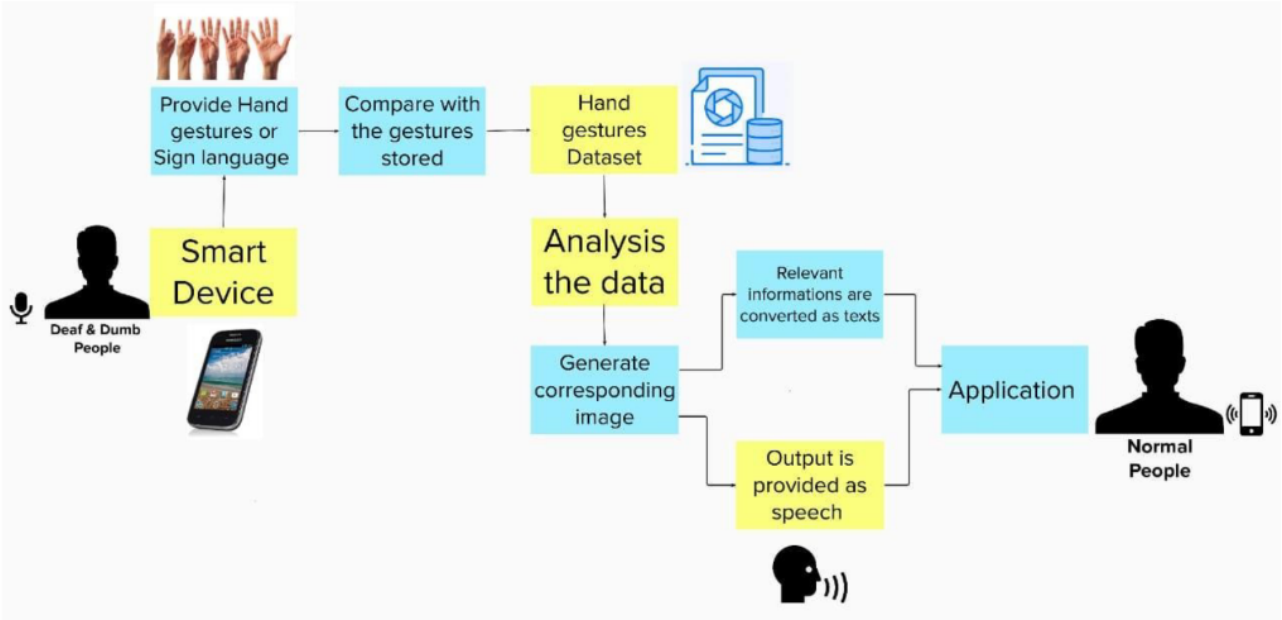


Table-1 : Components & Technologies:

S.No	Component	Description	Technology
1.	User Interface	How user interacts with application e.g. Web UI, Mobile App, Chatbot etc.	HTML, CSS, JavaScript / Angular Js / React Js etc.
2.	Application Logic-1	Logic for a process in the application	Java / Python
3.	Application Logic-2	Logic for a process in the application	IBM Watson STT service
4.	Application Logic-3	Logic for a process in the application	IBM Watson Assistant
5.	Database	Data Type, Configurations etc.	MySQL, NoSQL, etc.
6.	Cloud Database	Database Service on Cloud	IBM DB2, IBM Cloudant etc.
7.	File Storage	File storage requirements	IBM Block Storage or Other Storage Service or Local Filesystem
8.	External API-1	Purpose of External API used in the application	IBM Weather API, etc.
9.	External API-2	Purpose of External API used in the application	Aadhar API, etc.
10.	Machine Learning Model	Purpose of Machine Learning Model	Object Recognition Model, etc.
11.	Infrastructure (Server / Cloud)	Application Deployment on Local System / Cloud Local Server Configuration: Cloud Server Configuration :	Local, Cloud Foundry, Kubernetes, etc.

Table-2: Application Characteristics:

S.No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Robots and other tools provide home-based care and other assistance, allowing people with disabilities to live independently.	Artificial Intelligence like robots and software systems.
2.	Security Implementations	Set the inclusion and exclusion criteria , Report the results in the survey.	Artificial Intelligence
3.	Scalable Architecture	The improvement in the specially abled persons interaction with the environments.	Artificial Intelligence
4.	Availability	Justify the availability of application.	Conferencing technology
5.	Performance	Enables people with disabilities to step into a world where their difficulties are understood and taken into account.	Natural Language Processing (NLP)

- **User Stories**

Use the below template to list all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Register with the users information.	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard in the application.	High	Sprint-1
Customer (Deaf people)	To communicate with people using signs.	USN-2	As a user, I can see my application and made changes in any browser and register to it.	I can login and see my account in the application anywhere at anytime.	High	Sprint-1
Customer (Dumb people)	To communicate with people easily and efficiently.	USN-3	As a user, I can see my application and made changes in any browsers and register to it.	I can login and see my account in the application anywhere.	High	Sprint-1
Customer (Normal people)	User needs to communicate with specially abled people.	USN-4	As a user, I can register for the application by entering my email, password, and confirmation is made.	I can login and see my account.	Medium	Sprint-2
Customer (Learner of Sign language)	User needs to be aware and learn about sign language.	USN-5	As a user, I can create my account in the application with my email and password, to get knowledge about sign languages.	I can create my account and access the dashboard in the application.	High	Sprint-1
Customer (Web user)	They want the update on the application condition.	USN-6	As a user, I can register for the application by entering my email, password, and confirming my password. To get details about real-time communication.	I can able to use any browser to access the application from anywhere, to know anything about real-time communication.	High	Sprint-1
Customer Care Executive	They want to help people by sending application conditions.	USN-7	As a user, I can receive a message from the administration about conditions of application of real-time communication.	I will analyse and send SMS to the people.	High	Sprint-1

5. Coding, Solutioning and Testing

Model Training for Real Time Communication through AI for Specially Abled

Importing the packages.

In [2]:
```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
```

In [3]:
```
# Training Datagen
train_datagen = ImageDataGenerator(rescale=1/255,zoom_range=0.2,horizontal_flip=True,vertical_flip=False)
# Testing Datagen
test_datagen = ImageDataGenerator(rescale=1/255)
```

linking the streaming_body

In [4]:
```
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
cos_client = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='Lzw27RyYAXpOXEjMhc04K638iQoczlOKd5DbxFuLpmhc',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'realtimecommunication-donotdelete-pr-e7yebdi9hvsfug'
object_key = 'Dataset.zip'

streaming_body_4 = cos_client.get_object(Bucket=bucket, Key=object_key)['Body']

# Your data file was loaded into a botocore.response.StreamingBody object.
# Please read the documentation of ibm_boto3 and pandas to learn more about the possibilities to load the data.
# ibm_boto3 documentation: https://ibm.github.io/ibm-cos-sdk-python/
# pandas documentation: http://pandas.pydata.org/
```

unzipping the dataset

In [6]:
```
# Unzip the Dataset Zip File
from io import BytesIO
import zipfile
unzip = zipfile.ZipFile(BytesIO(streaming_body_4.read()), 'r')
file_paths = unzip.namelist()
for path in file_paths:
    unzip.extract(path)
```

In [7]:
```
%%bash
ls Dataset
```

```
test_set
training_set
```

In [8]:
```
# Training Dataset
x_train=train_datagen.flow_from_directory(r'/home/wsuser/work/Dataset/training_set',target_size=(64,64), class_mode='categorical',batch_size=900)
# Testing Dataset
x_test=test_datagen.flow_from_directory(r'/home/wsuser/work/Dataset/test_set',target_size=(64,64), class_mode='categorical',batch_size=900)
```

```
Found 15750 images belonging to 9 classes.
Found 2250 images belonging to 9 classes.
```

In [9]:
```
print("Length of x-train : ", len(x_train))
print("Length of x-test : ", len(x_test))
```

```
Length of x-train :  18
Length of x-test :  3
```

In [10]:
```
# The Class Indices in Training Dataset
x_train.class_indices
```

Out[10]:
```
{'A': 0, 'B': 1, 'C': 2, 'D': 3, 'E': 4, 'F': 5, 'G': 6, 'H': 7, 'I': 8}
```

Model Creation

In [11]:
```
# Importing Libraries
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Convolution2D,MaxPooling2D,Flatten,Dense
```

In [12]:
```
# Creating Model
model=Sequential()
```

In [13]:
```
# Adding Layers
model.add(Convolution2D(32,(3,3),activation='relu',input_shape=(64,64,3)))
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())

# Adding Hidden Layers
model.add(Dense(300,activation='relu'))
model.add(Dense(150,activation='relu'))

# Adding Output Layer
model.add(Dense(9,activation='softmax'))
```

In [14]:
```
# Compiling the Model
model.compile(loss='categorical_crossentropy',optimizer='adam',metrics=['accuracy'])
```

In [15]:
```
# Fitting the Model Generator
model.fit_generator(x_train,steps_per_epoch=len(x_train),epochs=10,validation_data=x_test,validation_steps=len(x_test))
```

```
/tmp/wsuser/ipykernel_164/1042518445.py:2: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which supports generators.
  model.fit_generator(x_train,steps_per_epoch=len(x_train),epochs=10,validation_data=x_test,validation_steps=len(x_test))
Epoch 1/10
18/18 [==============================] - 71s 4s/step - loss: 1.1343 - accuracy: 0.6152 - val_loss: 0.4147 - val_accuracy: 0.9058
Epoch 2/10
18/18 [==============================] - 69s 4s/step - loss: 0.2602 - accuracy: 0.9239 - val_loss: 0.2582 - val_accuracy: 0.9320
Epoch 3/10
18/18 [==============================] - 71s 4s/step - loss: 0.1165 - accuracy: 0.9669 - val_loss: 0.2227 - val_accuracy: 0.9587
Epoch 4/10
18/18 [==============================] - 72s 4s/step - loss: 0.0610 - accuracy: 0.9846 - val_loss: 0.2374 - val_accuracy: 0.9698
Epoch 5/10
18/18 [==============================] - 73s 4s/step - loss: 0.0357 - accuracy: 0.9902 - val_loss: 0.2313 - val_accuracy: 0.9707
Epoch 6/10
18/18 [==============================] - 72s 4s/step - loss: 0.0265 - accuracy: 0.9939 - val_loss: 0.2498 - val_accuracy: 0.9756
Epoch 7/10
18/18 [==============================] - 71s 4s/step - loss: 0.0170 - accuracy: 0.9965 - val_loss: 0.2795 - val_accuracy: 0.9756
Epoch 8/10
18/18 [==============================] - 72s 4s/step - loss: 0.0119 - accuracy: 0.9977 - val_loss: 0.2573 - val_accuracy: 0.9769
Epoch 9/10
18/18 [==============================] - 72s 4s/step - loss: 0.0095 - accuracy: 0.9981 - val_loss: 0.2782 - val_accuracy: 0.9782
Epoch 10/10
18/18 [==============================] - 72s 4s/step - loss: 0.0075 - accuracy: 0.9987 - val_loss: 0.3134 - val_accuracy: 0.9764
```

Out[15]:

Saving the Model

In [16]:
```
model.save('SANJAI.h5')
# Current accuracy is 0.825
```

In [17]:
```
# Convert the Saved Model to a Tar Compressed Format
!tar -zcvf trainedModel.tgz SANJAI.h5
```

```
SANJAI.h5
```

In [18]:
```
%%bash
ls -ll
```

```
total 210000
drwxrwx--- 4 wsuser wscommon      4096 Nov 16 19:02 Dataset
-rw-rw---- 1 wsuser wscommon 111324760 Nov 16 19:15 SANJAI.h5
-rw-rw---- 1 wsuser wscommon 103709912 Nov 16 19:15 trainedModel.tgz
```

Watson Machine Learning

In [19]:
```
!pip install watson-machine-learning-client --upgrade
```

```
Collecting watson-machine-learning-client
  Downloading watson_machine_learning_client-1.0.391-py3-none-any.whl (538 kB)
     |████████████████████████████████| 538 kB 8.4 MB/s eta 0:00:01
Requirement already satisfied: boto3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.18.21)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2.26.0)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.26.7)
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (0.3.3)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (0.8.9)
Requirement already satisfied: tqdm in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (4.62.3)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2022.9.24)
Requirement already satisfied: ibm-cos-sdk in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (2.11.0)
Requirement already satisfied: pandas in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from watson-machine-learning-client) (1.3.4)
Requirement already satisfied: s3transfer<0.6.0,>=0.5.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learni
ng-client) (0.5.0)
Requirement already satisfied: botocore<1.22.0,>=1.21.21 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learn
ing-client) (1.21.41)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from boto3->watson-machine-learning
-client) (0.10.0)
Requirement already satisfied: python-dateutil<3.0.0,>=2.1 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from botocore<1.22.0,>=1.21.21
->boto3->watson-machine-learning-client) (2.8.2)
Requirement already satisfied: six>=1.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from python-dateutil<3.0.0,>=2.1->botocore<1.22.
0,>=1.21.21->boto3->watson-machine-learning-client) (1.15.0)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson-ma
chine-learning-client) (2.11.0)
Requirement already satisfied: ibm-cos-sdk-core==2.11.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from ibm-cos-sdk->watson-machine-
learning-client) (2.11.0)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->watson-machine-learning-clien
t) (3.3)
Requirement already satisfied: charset-normalizer~=2.0.0 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from requests->watson-machine-le
arning-client) (2.0.4)
Requirement already satisfied: pytz>=2017.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas->watson-machine-learning-client)
(2021.3)
Requirement already satisfied: numpy>=1.17.3 in /opt/conda/envs/Python-3.9/lib/python3.9/site-packages (from pandas->watson-machine-learning-client)
(1.20.3)
Installing collected packages: watson-machine-learning-client
Successfully installed watson-machine-learning-client-1.0.391
```

In [20]:
```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
    "url": "https://us-south.ml.cloud.ibm.com",
    "apikey": "_c84HIUddEc74mO6dInb_l1g8FNC4l3OAAIocQOhaWCI"
}

client = APIClient(wml_credentials)
```

In [21]:
```
client
```

Out[21]:

Save to Deployment Space

In [22]:
```
def guid_from_space_name(client, space_name):
    space = client.spaces.get_details()
    return (next(item for item in space['resources'] if item['entity']["name"] == space_name)['metadata']['id'])
```

In [23]:
```
space_uid = guid_from_space_name(client, 'Real_Time')
print("Space UID : ", space_uid)
```

```
Space UID :  40cfd62c-38d1-4f55-b4a6-c1b77eab8cf5
```

In [24]:
```
client.set.default_space(space_uid)
```

Out[24]: 'SUCCESS'

In [25]:
```
client.software_specifications.list()
```

```
------------------------------  ------------------------------------  ----
NAME                            ASSET_ID                              TYPE
default_py3.6                   0062b8c9-8b7d-44a0-a9b9-46c416adcbd9  base
kernel-spark3.2-scala2.12       020d69ce-7ac1-5e68-ac1a-31189867356a  base
pytorch-onnx_1.3-py3.7-edt      069ea134-3346-5748-b513-49120e15d288  base
scikit-learn_0.20-py3.6         09c5a1d0-9c1e-4473-a344-eb7b665ff687  base
spark-mllib_3.0-scala_2.12      09f4cff0-90a7-5899-b9ed-1ef348aebdee  base
pytorch-onnx_rt22.1-py3.9       0b848dd4-e681-5599-be41-b5f6fccc6471  base
ai-function_0.1-py3.6           0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda  base
shiny-r3.6                      0e6e79df-875e-4f24-8ae9-62dcc2148306  base
tensorflow_2.4-py3.7-horovod    1092590a-307d-563d-9b62-4eb7d64b3f22  base
pytorch_1.1-py3.6               10ac12d6-6b30-4ccd-8392-3e922c096a92  base
tensorflow_1.15-py3.6-ddl       111e41b3-de2d-5422-a4d6-bf776828c4b7  base
autoai-kb_rt22.2-py3.10         125b6d9a-5b1f-5e8d-972a-b251688ccf40  base
runtime-22.1-py3.9              12b83a17-24d8-5082-900f-0ab31fbfd3cb  base
scikit-learn_0.22-py3.6         154010fa-5b3b-4ac1-82af-4d5ee5abbc85  base
default_r3.6                    1b70aec3-ab34-4b87-8aa0-a4a3c8296a36  base
pytorch-onnx_1.3-py3.6          1bc6029a-cc97-56da-b8e0-39c3880dbbe7  base
kernel-spark3.3-r3.6            1c9e5454-f216-59dd-a20e-474a5cdf5988  base
pytorch-onnx_rt22.1-py3.9-edt   1d362186-7ad5-5b59-8b6c-9d0880bde37f  base
tensorflow_2.1-py3.6            1eb25b84-d6ed-5dde-b6a5-3fbdf1665666  base
spark-mllib_3.2                 20047f72-0a98-58c7-9ff5-a77b012eb8f5  base
tensorflow_2.4-py3.8-horovod    217c16f6-178f-56bf-824a-b19f20564c49  base
runtime-22.1-py3.9-cuda         26215f05-08c3-5a41-a1b0-da66306ce658  base
do_py3.8                        295addb5-9ef9-547e-9bf4-92ae3563e720  base
autoai-ts_3.8-py3.8             2aa0c932-798f-5ae9-abd6-15e0c2402fb5  base
tensorflow_1.15-py3.6           2b73a275-7cbf-420b-a912-eae7f436e0bc  base
kernel-spark3.3-py3.9           2b7961e2-e3b1-5a8c-a491-482c8368839a  base
pytorch_1.2-py3.6               2c8ef57d-2687-4b7d-acce-01f94976dac1  base
spark-mllib_2.3                 2e51f700-bca0-4b0d-88dc-5c6791338875  base
pytorch-onnx_1.1-py3.6-edt      32983cea-3f32-4400-8965-dde874a8d67e  base
spark-mllib_3.0-py37            36507ebe-8770-55ba-ab2a-eafe787600e9  base
spark-mllib_2.4                 390d21f8-e58b-4fac-9c55-d7ceda621326  base
autoai-ts_rt22.2-py3.10         396b2e83-0953-5b86-9a55-7ce1628a406f  base
xgboost_0.82-py3.6              39e31acd-5f30-41dc-ae44-60233c80306e  base
pytorch-onnx_1.2-py3.6-edt      40589d0e-7019-4e28-8daa-fb03b6f4fe12  base
pytorch-onnx_rt22.2-py3.10      40e73f55-783a-5535-b3fa-0c8b94291431  base
default_r36py38                 41c247d3-45f8-5a71-b065-8580229facf0  base
autoai-ts_rt22.1-py3.9          4269d26e-07ba-5d40-8f66-2d495b0c71f7  base
autoai-obm_3.0                  42b92e18-d9ab-567f-988a-4240ba1ed5f7  base
pmml-3.0_4.3                    493bcb95-16f1-5bc5-bee8-81b8af80e9c7  base
spark-mllib_2.4-r_3.6           49403dff-92e9-4c87-a3d7-a42d0021c095  base
xgboost_0.90-py3.6              4ff8d6c2-1343-4c18-85e1-689c965304d3  base
pytorch-onnx_1.1-py3.6          50f95b2a-bc16-43bb-bc94-b0bed208c60b  base
autoai-ts_3.9-py3.8             52c57136-80fa-572e-8728-a5e7cbb42cde  base
spark-mllib_2.4-scala_2.11      55a70f99-7320-4be5-9fb9-9edb5a443af5  base
spark-mllib_3.0                 5c1b0ca2-4977-5c2e-9439-ffd44ea8ffe9  base
autoai-obm_2.0                  5c2e37fa-80b8-5e77-840f-d912469614ee  base
pytorch-onnx_1.7-py3.8          6d4d3cdc-b562-5bf9-a2d4-ea90a478456b  base
------------------------------  ------------------------------------  ----
Note: Only first 50 records were displayed. To display more use 'limit' parameter.
```

In [26]:
```
software_spec_uid = client.software_specifications.get_uid_by_name("tensorflow_rt22.1-py3.9")
software_spec_uid
```

Out[26]: 'acd9c798-6974-5d2f-a657-ce06e986df4d'

In [27]:
```
model_details = client.repository.store_model(model='trainedModel.tgz', meta_props={
    client.repository.ModelMetaNames.NAME: "CNN",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID: software_spec_uid,
    client.repository.ModelMetaNames.TYPE: "tensorflow_2.7"})
model_id = client.repository.get_model_id(model_details)
```

In [28]:
```
model_id
```

Out[28]: '4154aedd-4fff-46f6-b056-ea4e566d3643'

In [29]:
```
client.repository.download(model_id,'SANJAI1.tar.gz')
```

```
Successfully saved model content to file: 'SANJAI1.tar.gz'
```

Out[29]: '/home/wsuser/work/SANJAI1.tar.gz'

TESTING PART OF MODEL

In [30]:
```
#Testing the model.
import numpy as np
from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
```

In [31]:
```
model=load_model('SANJAI.h5')
img=image.load_img(r'/home/wsuser/work/Dataset/test_set/A/1.png',
                   target_size=(64,64))
```

In [32]:
```
img
```

Out[32]:

In [33]:
```
img=image.load_img(r"/home/wsuser/work/Dataset/test_set/A/90.png",target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]
```

Out[33]: 'A'

In [34]:
```
img=image.load_img(r"/home/wsuser/work/Dataset/test_set/C/90.png",target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]
```

Out[34]: 'C'

In [35]:
```
img=image.load_img(r"/home/wsuser/work/Dataset/test_set/I/90.png",target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]
```

Out[35]: 'I'

In [36]:
```
img=image.load_img(r"/home/wsuser/work/Dataset/test_set/E/90.png",target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]
```

Out[36]: 'E'

In [37]:
```
img=image.load_img(r"/home/wsuser/work/Dataset/test_set/F/90.png",target_size=(64,64))
x=image.img_to_array(img)
x=np.expand_dims(x,axis=0)
y=np.argmax(model.predict(x),axis=1)
index=['A','B','C','D','E','F','G','H','I']
index[y[0]]
```

Out[37]: 'F'

Accuracy is over 90+ percentage because of the overfitting phenomenon.when we test our model with live data then the accuracy will decrease.

Spyder Deployment Code

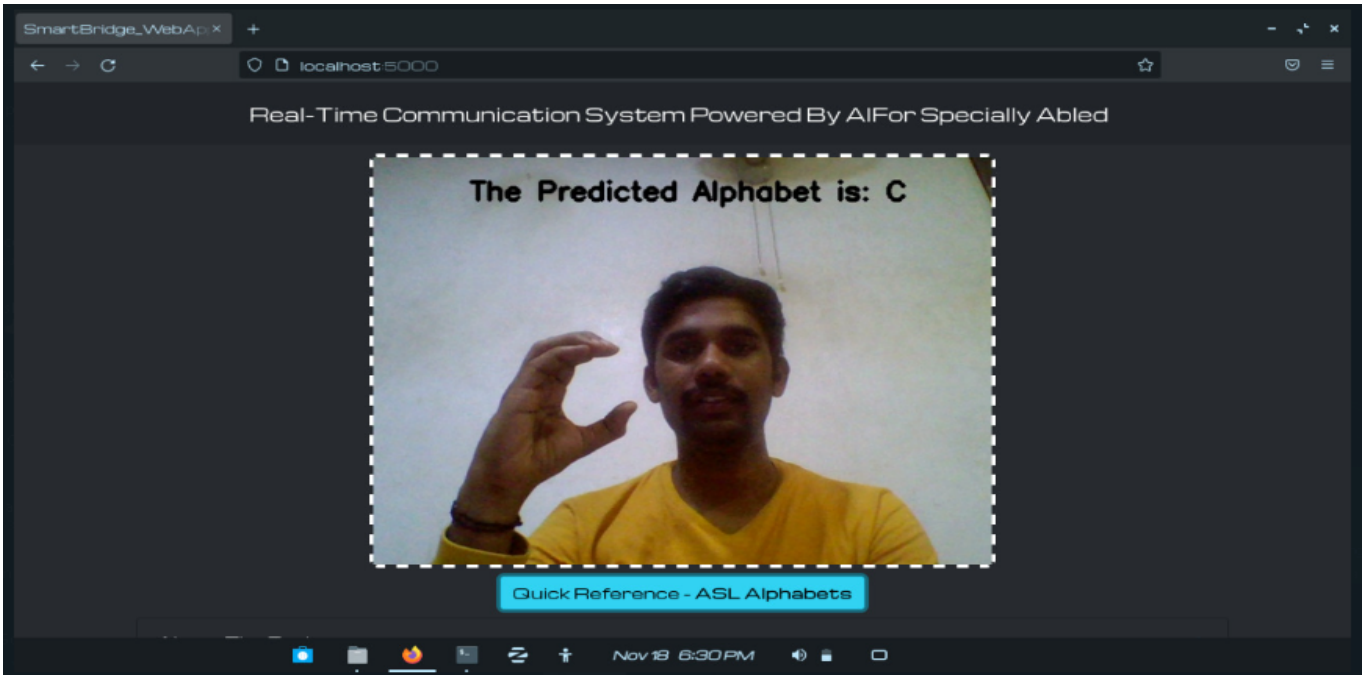
In [ ]:
```
import cv2 #mporting opencv Library this i to open camera and take the video
import numpy as np # to convert image to array and expand dimensions
from tensorflow.keras.models import load_model # to load the saved model
from tensorflow.keras.preprocessing import image # to preproccess the image
model = load_model("dataset.h5") # we are loading the saved moodek
video = cv2.VideoCapture(0) # two parameters 1, bool 0 or 1, frame
index = ["A","B","C","D","E","F","G","H","I"]
index=['A','B','C','D','E','F','G','H','I']
#from playsound import playsound
while(1):
    success,frame = video.read()
    cv2.imwrite("image.jpg",frame)
    img = image.load_img("image.jpg",target_size = (64,64))
    x = image.img_to_array(img)
    x = np.expand_dims (x,axis = 0)
    pred = np.argmax(model.predict(x),axis=1)
    p = index [pred[0]]
    print("predicted letter is: "+ str(p))
    #playSound("letter"+str(str(index [p])+"is detected"))
    cv2.putText (frame, "predicted letter is "+str(p), (100, 100), cv2. FONT_HERSHEY_SIMPLEX, 1,(0,0,0), 4)
    cv2.imshow("showcasewindow", frame)

    if cv2.waitkey(1) & 0xFF == ord('a'):
        break
video.release()
cv2.destroyAllwindows()
```

6. Result

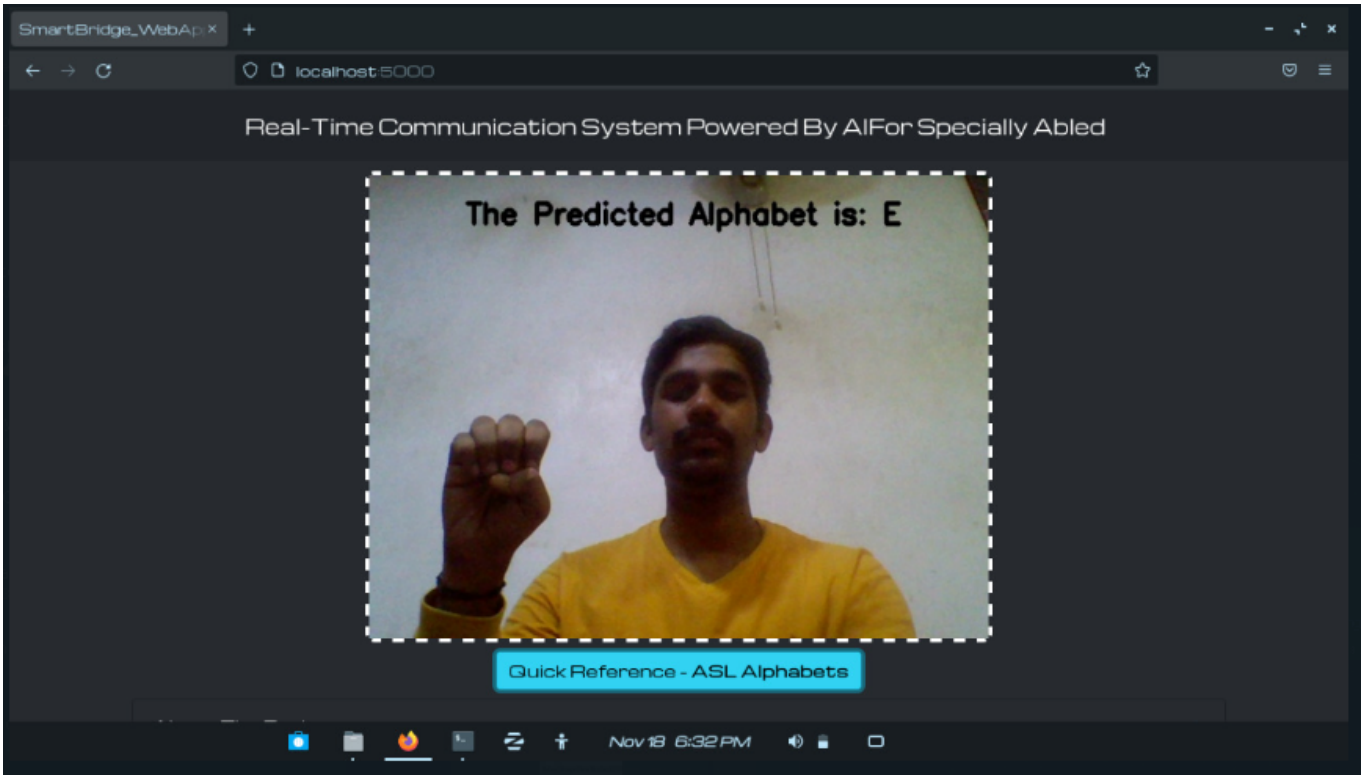
A series of photos were used to implement and test the suggested approach. A collection of 2250 photos of the alphabet from "A" to "I" are utilised for the testing database, while a set of 15750 images are used for the training database.

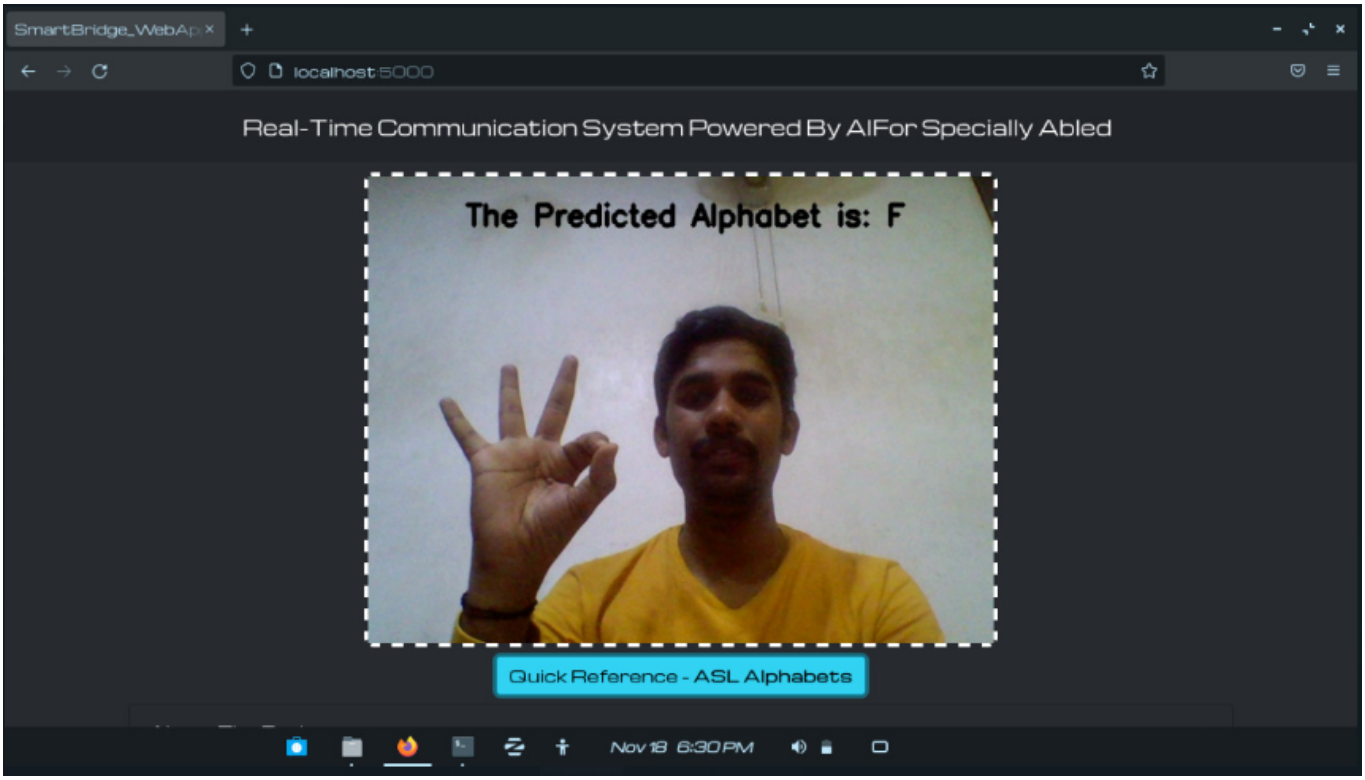
The corresponding Alphabets are displayed on the screen as soon as the motion is recognised.

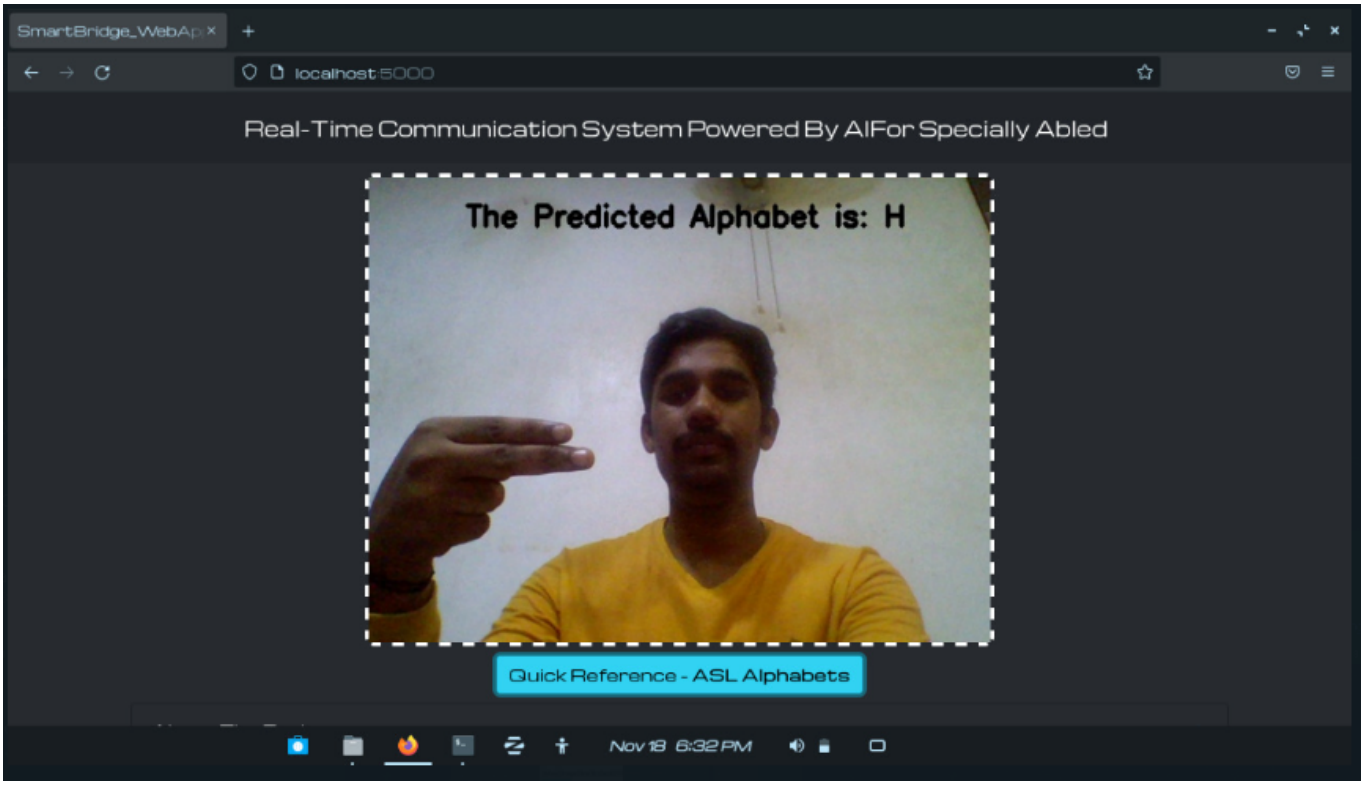
Snapshots of our model predicting the hand gestures are shown below :




SmartBridge_WebAp
localhost:5000
Real-Time Communication System Powered By AIFor Specially Abled
The Predicted Alphabet is: D
Quick Reference - ASL Alphabets
Nov 18 6:30 PM


SmartBridge_WebAp
localhost:5000
Real-Time Communication System Powered By AIFor Specially Abled
The Predicted Alphabet is: E
Quick Reference - ASL Alphabets
Nov 18 6:32 PM


SmartBridge_WebAp
localhost:5000
Real-Time Communication System Powered By AIFor Specially Abled
The Predicted Alphabet is: F
Quick Reference - ASL Alphabets
Nov 18 6:30 PM


SmartBridge_WebAp
localhost:5000
Real-Time Communication System Powered By AIFor Specially Abled
The Predicted Alphabet is: H
Quick Reference - ASL Alphabets
Nov 18 6:32 PM

7. Advantages and Disadvantages :

Advantages:

1. It is feasible to develop a mobile application to close the communication gap between the hearing-impaired and the rest of society.
2. The user may select which sign language to read by adding the dataset when new sign language standards are created.
3. The disabled people who have hearing impairment will not have to be socially anxious anymore. They can communicate with great confidence

Disadvantages:

1. The present model is limited to the letters A through I.
2. Alphabets from J cannot be recognised in the absence of gesture recognition because they need user input in the form of a gesture.
3. The accuracy isn't excellent because there aren't many or high-quality photographs in the dataset, but that can be fixed by changing the datas.

8. Conclusion :

The use of sign languages can help normal and deaf individuals communicate more effectively.

Our approach strives to reduce the communication gap between the deaf community and the rest of society since it supports two-way conversation.

Our suggested technology converts sign languages into human-understandable English language.

With the help of this technology, the model receives hand gestures, recognises them, and then shows the corresponding Alphabet on the screen.

This initiative allows deaf-mute persons to perform sign language with their hands, which will later be translated into alphabets.

9. Future Scope :

For persons with particular needs, such as the deaf and dumb, having technology that can convert hand sign language to its appropriate alphabet is a game changer. The web programme may easily be developed to detect letters other than "I," numbers, and other symbols with the addition of gesture recognition. Gesture recognition can also be used to control software and hardware interfaces.

10. References

[1] Keras Image Processing Doc :- https://keras.io/api/preprocessing/image/

[2] Keras ImageDataset From Directory Doc :- https://keras.io/api/preprocessing/image/#imagedatasetfromdirectory-function

[3] CNN using Tensorflow :- https://www.youtube.com/watch?v=umGJ30-15_A

[4] OpenCV Basics of Processing Image :- https://www.youtube.com/watch?v=mjKd1Tzl70I

[5] Flask Basics :- https://www.youtube.com/watch?v=lj4I_CvBnt0

[6] IBM Academic Partner Account Creation :- https://www.youtube.com/watch?v=x6i43M7BAqE

[7] CNN Deployment and Download through IBM Cloud :- https://www.youtube.com/watch?v=BzouqMGJ41k

[8] Matusiak, K., Skulimowski, P., & Strumiłło, P. (2013, June). Object recognition in a mobile phone application for visually impaired users. In 2013 6th International Conference on Human System Interactions (HSI) (pp. 479-484). IEEE
.

[9] Hermus, K., & Wambacq, P. (2006). A review of signal subspace speech enhancement and its application to noise robust speech recognition. EURASIP Journal on Advances in Signal Processing, 2007(1), 045821.

[10] Dimitrov, V., Jullien, G., & Muscedere, R. (2017). Multiple-base number system: theory and applications. CRC press.
.

[11] Huyan, Z., Xu, L., Fang, S., Liu, Z., Zhang, X., & Li, L. (2014). Field information acquisition system research based on offline speech recognition. Int. J. Database Theory Appl, 7, 45-58.

[12] Bigham, J. P., Jayant, C., Miller, A., White, B., & Yeh, T. (2010, June). VizWiz:: LocateIt-enabling blind people to locate objects in their environment. In 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops (pp. 65-72). IEEE.

[13] Manduchi, R., Kurniawan, S., & Bagherinia, H. (2010, October). Blind guidance using mobile computer vision: A usability study. In Proceedings of the 12th international ACM SIGACCESS conference

on
Computers and accessibility (pp. 241-242).

[14] Ivanchenko, V., Coughlan, J., Gerrey, W., & Shen, H. (2008, October). Computer vision-based clear path
guidance for blind wheelchair users. In Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility (pp. 291-292).

[15] Johnsen, A., Grønli, T. M., & Bygstad, B. (2012). Making touch-based mobile phones accessible for the
visually impaired. Norsk informatikkonferanse,(Bodø, Norway, 2012).

[16] Jiang, R., Lin, Q., & Qu, S. (2016). Let Blind People See: Real-Time Visual Recognition with Results
Converted to 3D Audio. Report No. 218, Standord University, Stanford, USA.

[17] Kamble, K., & Kagalkar, R. (2014). A review: translation of text to speech conversion for Hindi
language. International Journal of Science and Research (IJSR) Volume, 3.

[18] Kumar, A., & Chourasia, A. (2018). Blind Navigation System Using Artificial Intelligence. International
Research Journal of Engineering and Technology, 5(3).

[19] BELGHIT, H., & BELLARBI, A. Object Recognition Based on ORB Descriptor for Markerless
Augmented Reality.

[20] Coughlan, J., & Manduchi, R. (2009). Functional assessment of a camera phone-based wayfinding system operated by blind and visually impaired users. International Journal on Artificial Intelligence
Tools, 18(03), 379-397.

[21] Chen, C., & Raman, T. V. (2009). Announcing eyes-free shell for Android. Retrieved December, 21, 2016.

[22] Gill, J. (2000). Personal electronic mobility devices. Information for Professionals Working with Visually
Disabled People. http://www. tiresias. org.

[23] Coughlan, J., & Manduchi, R. (2007). Color targets: Fiducials to help visually impaired people find their
way by camera phone. EURASIP Journal on Image and Video Processing, 2007, 1-13.

[24]Arora, S. J., & Singh, R. P. (2012). Automatic speech recognition: a review. International Journal of

Computer Applications, 60(9).

[25] Omankhanlen, A. E., & Ogaga-Oghene, J. (2013). The Dynamics of Global Strategy and Strategic
Alliances in International Trade and Investment. INTERNATIONAL JOURNAL OF RESEARCH IN
COMPUTER APPLICATION & MANAGEMENT, 3(12), 41-48.

[26] **(PDF) A Model for Real-Time Recognition and Textual Representation of Malaysian Sign Language through Image Processing (researchgate.net)**

11. Appendix :

- **Source code for model building**

```
IMPORTING THE DATASETS
```

```
In [1]: !unzip '/content/drive/MyDrive/IBMPROJECT/conversation engine for deaf and dumb.zip'
```

```
Streaming output truncated to the last 5000 lines.
 extracting: Dataset/training_set/G/1225.png
 extracting: Dataset/training_set/G/1226.png
 extracting: Dataset/training_set/G/1227.png
 extracting: Dataset/training_set/G/1228.png
 extracting: Dataset/training_set/G/1229.png
  inflating: Dataset/training_set/G/123.png
 extracting: Dataset/training_set/G/1230.png
 extracting: Dataset/training_set/G/1231.png
 extracting: Dataset/training_set/G/1232.png
  inflating: Dataset/training_set/G/1233.png
  inflating: Dataset/training_set/G/1234.png
  inflating: Dataset/training_set/G/1235.png
  inflating: Dataset/training_set/G/1236.png
  inflating: Dataset/training_set/G/1237.png
  inflating: Dataset/training_set/G/1238.png
  inflating: Dataset/training_set/G/1239.png
  inflating: Dataset/training_set/G/124.png
  inflating: Dataset/training_set/G/1240.png
```

```
TRAIN AND TEST THE DATA
```

```
In [2]: from keras.preprocessing.image import ImageDataGenerator
        train_datagen=ImageDataGenerator(rescale = 1./255, shear_range=0.2, zoom_range=0.2,horizontal_flip=True)
        test_datagen = ImageDataGenerator(rescale=1./255)
```

```
In [3]: x_train = train_datagen.flow_from_directory("/content/Dataset/training_set", target_size=(64,64),batch_size=300,
                                                 class_mode='categorical', color_mode ="grayscale")
```

```
Found 15750 images belonging to 9 classes.
```

```
In [4]: x_test = test_datagen.flow_from_directory("/content/Dataset/test_set", target_size=(64,64),batch_size=300,
                                               class_mode='categorical', color_mode ="grayscale")
```

```
Found 2250 images belonging to 9 classes.
```

```
MODEL TRAINING
```

In [5]:
```
from keras.models import Sequential
from keras.layers import Dense
from keras.layers import Convolution2D
from tensorflow.keras.layers import Conv2D, MaxPooling2D
from keras.layers import Dropout
from keras.layers import Flatten
```

In [6]:
```
model=Sequential()
```

In [7]:
```
model.add(Convolution2D(32,(3,3), input_shape=(64,64,1), activation = 'relu'))
```

In [8]:
```
model.add(MaxPooling2D(pool_size=(2,2)))
```

In [9]:
```
model.add(Flatten())
```

In [14]:
```
model.add(Dense( units=512, activation='relu'))
```

In [15]:
```
model.add(Dense(units=9, activation='softmax'))
```

In [16]:
```
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])
```

In [17]:
```
model.fit_generator(x_train, steps_per_epoch=24, epochs=10, validation_data=x_test,validation_steps=40)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: UserWarning: `Model.fit_generator` is deprecated and will be removed i
n a future version. Please use `Model.fit`, which supports generators.
  """Entry point for launching an IPython kernel.

Epoch 1/10
24/24 [==============================] - 17s 666ms/step - loss: 1.9786 - accuracy: 0.5628
Epoch 2/10
24/24 [==============================] - 16s 662ms/step - loss: 1.4525 - accuracy: 0.6621
Epoch 3/10
24/24 [==============================] - 16s 676ms/step - loss: 0.9580 - accuracy: 0.6842
Epoch 4/10
24/24 [==============================] - 16s 675ms/step - loss: 0.7076 - accuracy: 0.7240
Epoch 5/10
24/24 [==============================] - 16s 659ms/step - loss: 0.6103 - accuracy: 0.7488
Epoch 6/10
24/24 [==============================] - 16s 663ms/step - loss: 0.5005 - accuracy: 0.8054
Epoch 7/10
24/24 [==============================] - 17s 679ms/step - loss: 0.4164 - accuracy: 0.8904
Epoch 8/10
24/24 [==============================] - 18s 723ms/step - loss: 0.3408 - accuracy: 0.8994
Epoch 9/10
24/24 [==============================] - 16s 659ms/step - loss: 0.2641 - accuracy: 0.9536
Epoch 10/10
24/24 [==============================] - 16s 664ms/step - loss: 0.1676 - accuracy: 0.9672
```

Out[17]:
```
<keras.callbacks.History at 0x7f8d786377d0>
```

In [18]:
```
model.save('RSL.h5')
```

- **Github link :**
  https://github.com/IBM-EPBL/IBM-Project-1855-1658418245

- **Project demo link :**
  https://youtu.be/6MU_YC6KWE8