

## SPRINT- 1

TEAM ID	PNT2022TMID27588
PROJECT NAME	Real-Time Communication System Powered by AI for Specially Abled
DATE OF THE MEETING	26 -10-2022

**MINUTES OF THE MEETING:**

- The process of data collection system was discussed and procedure was implemented in the local system.
- The training images were tested and verified.
- The image processing procedure was detailly examined and discussed .
- The required python modules were installed and issues in the model were rectified .

### Data collection:

[illegible]

## Image preprocessing:

The screenshot shows a Google Meet window with a presentation titled "JAYA SURYA K 19IT009 is presenting". The presentation content is a Jupyter Notebook titled "Image Preprocessing" on the IBM Cloud Platform. The notebook code is as follows:

```
In [1]: # Importing Libraries
from tensorflow.keras.preprocessing.image import ImageDataGenerator

In [2]: # Image Augmentation
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]:
import os, types
import pandas as pd
from h2oai.client import Config
import urllib3

def __test__(self): return 0

# Returns: bool
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove these credentials before you share the notebook.
os_client = os_client.client(username='api',
                             ibm_api_key_id='mc11c6e-07b4a8301212807b199180_880077191',
                             ibm_auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
                             config=Config(signature_version='oauth')),
endpoint_url='https://api.private.us.cloud-object-storage.appdomain.cloud')

bucket = 'imageclassification-donordata-pg-127a19a7adap'
object_key = 'dataset.zip'

streaming_body_2 = os_client.get_object(bucket=bucket, key=object_key)

# Your data file was loaded into a h2oai.response.StreamingBody object.
# Please read the documentation of ibm_botocore and pandas to learn more about the possibilities to load the data.
```

The notebook interface shows the "Image Preprocessing" title and the code cells. The bottom of the screen shows the Google Meet interface with a timer at 10:44 and a participant list on the right.

## SPRINT- 2

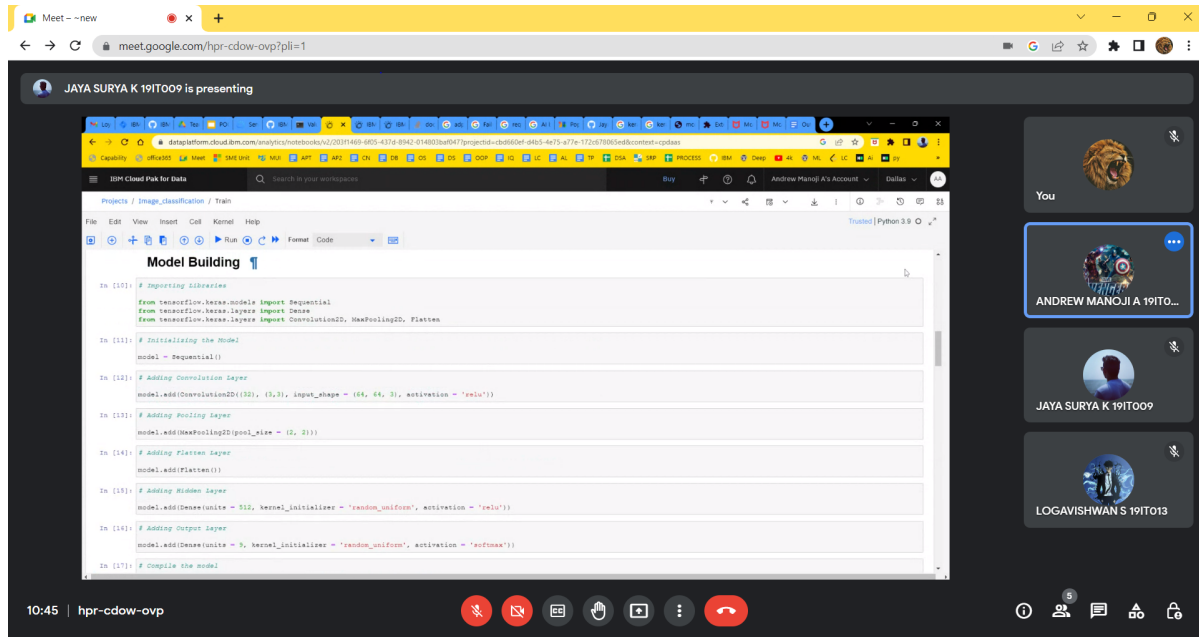
TEAM ID	PNT2022TMID27588
PROJECT NAME	Real-Time Communication System Powered by AI for Specially Abled
DATE OF THE MEETING	02-11-2022

## MINUTES OF THE MEETING:

- The model building of the module was discussed and implemented in the IBM cloud.
- The required packages were installed and imported .
- The issues in installation of packages were cleared.

- The dataset was imported in Ibm cloud and it was unzipped using python code.
- The model was trained using the given training images in the dataset.

### Model Building and training:



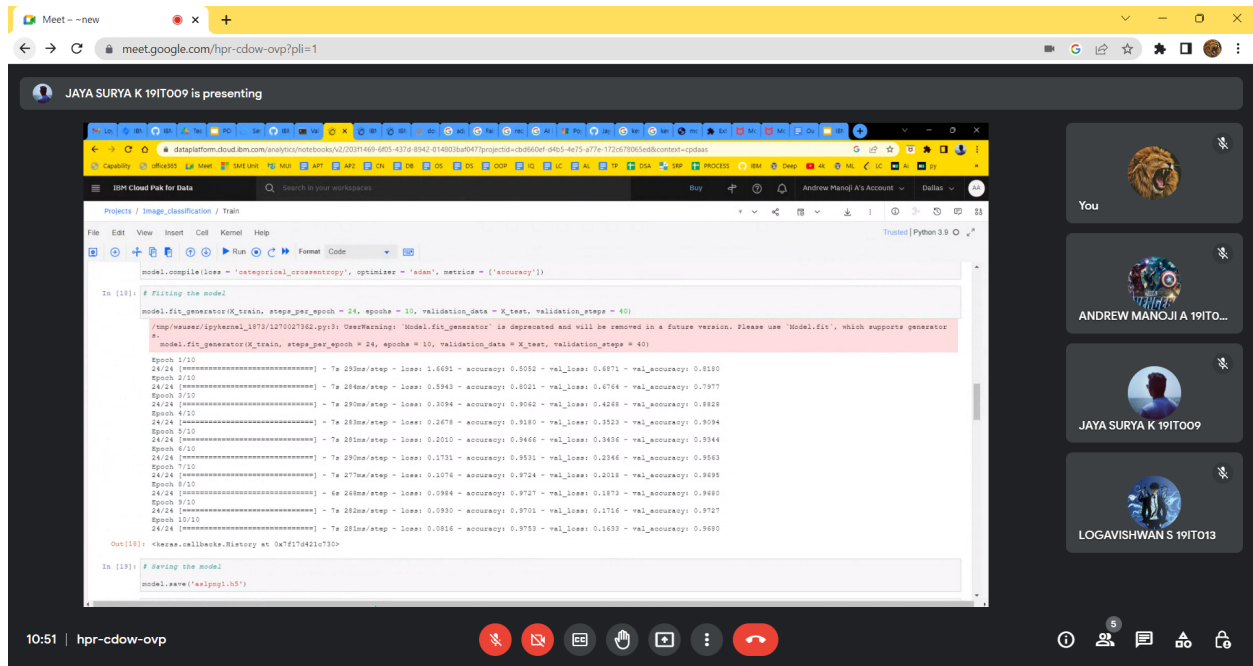
## SPRINT- 3

TEAM ID	PNT2022TMID27588
PROJECT NAME	Real-Time Communication System Powered by AI for Specially Abled
DATE OF THE MEETING	7-11-2022

### MINUTES OF THE MEETING:

- The model trained in the earlier sprint was saved and loaded using the `model.load()` function.
- The loaded model was tested using the sample images provided in the testset of the dataset packages.
- The errors in the testing the model was rectified and the correctness of the model was improved.

## Training and testing:



## SPRINT- 4

TEAM ID	PNT2022TMID27588
PROJECT NAME	Real-Time Communication System Powered by AI for Specially Abled
DATE OF THE MEETING	14-11-2022

## MINUTES OF THE MEETING:

- The train and test model was integrated with an HTML page with Flask application.
- The flask application development was discussed and done with video reference provided in the Ibm cloud.
- The API routes were built in the flask application and the python code for the opencv module was built under the prediction route.
- The model was loaded and it was made to read the real time human signals with the help of opencv2 and keras.
- The output was then tested and verified.

### Implementation of the application:

