

## IMPORT LIBRARIES

```
In [ ]: from tensorflow.keras.models import Sequential
        from tensorflow.keras.layers import Dense, Conv2D
        from tensorflow.keras.preprocessing.image import ImageDataGenerator
        import numpy as np
        import matplotlib.pyplot as plt
        import cv2
```

unzip the file

```
In [ ]: !unzip -q '/content/drive/MyDrive/zipfile/conversations.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
inflating: Dataset/training_set/G/1241.png
inflating: Dataset/training_set/G/1242.png
inflating: Dataset/training_set/G/1243.png
inflating: Dataset/training_set/G/1244.png
inflating: Dataset/training_set/G/1245.png
extracting: Dataset/training_set/G/1246.png
inflating: Dataset/training_set/G/1247.png
inflating: Dataset/training_set/G/1248.png
inflating: Dataset/training_set/G/1249.png
inflating: Dataset/training_set/G/125.png
inflating: Dataset/training_set/G/1250.png
inflating: Dataset/training_set/G/1251.png
inflating: Dataset/training_set/G/1252.png
inflating: Dataset/training_set/G/1253.png
inflating: Dataset/training_set/G/1254.png
inflating: Dataset/training_set/G/1255.png
inflating: Dataset/training_set/G/1256.png
inflating: Dataset/training_set/G/1257.png
inflating: Dataset/training_set/G/1258.png
inflating: Dataset/training_set/G/1259.png
inflating: Dataset/training_set/G/126.png
inflating: Dataset/training_set/G/1260.png
inflating: Dataset/training_set/G/1261.png
extracting: Dataset/training_set/G/1262.png
inflating: Dataset/training_set/G/1263.png
inflating: Dataset/training_set/G/1264.png
inflating: Dataset/training_set/G/1265.png
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

inflating: Dataset/training\_set/H/316.png  
inflating: Dataset/training\_set/H/317.png  
inflating: Dataset/training\_set/H/318.png



[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]



[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]



[illegible]



[illegible]

[illegible]

[illegible]

```
In [ ]: from keras.preprocessing.image import ImageDat
train_datagen=ImageDataGenerator(rescale = 1./
test_datagen = ImageDataGenerator(rescale=1./2
```

```
In [ ]: x_train = train_datagen.flow_from_directory("/
cl
```

Found 5750 images belonging to 9 classes.

```
In [ ]: x_test = test_datagen.flow_from_directory("/co
class
```

Found 2250 images belonging to 9 classes.

```
In [ ]: len(x_train)
```

Out[ ]: 58

```
In [ ]: len(x_test)
```

Out[ ]: 23

```
In [ ]: x_train.class_indices
```

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

MODEL BUILDING

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

```
In [ ]: #Creating the model
model=Sequential()
#Adding the Layers
model.add(Convolution2D(32,(3,3), input_shape=
model.add(MaxPooling2D(pool_size=(2,2)))
model.add(Flatten())

#adding hidden Layers
model.add(Dense(400, activation='relu'))
model.add(Dense(200, activation='relu'))
model.add(Dense(100, activation='relu'))

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

```
In [ ]: model.compile(loss='categorical_crossentropy',
```

```
In [15]: model.fit_generator(x_train, steps_per_epoch=3
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: UserWarning: `Model.fit_generator` is deprecated and will be removed in a future version. Please use `Model.fit`, which supports generators.
```

```
"""Entry point for launching an IPython kernel.
```

```
Epoch 1/10
```

```
30/30 [=====] - ETA: 0s - loss: 0.8235 - accuracy: 0.7190
```

```
WARNING:tensorflow:Your input ran out of data; interrupting training. Make sure that your dataset or generator can generate at least `steps_per_epoch * epochs` batches (in this case, 50 batches). You may need to use the repeat() function when building your dataset.
```

```
30/30 [=====] - 1325s 45s/step - loss: 0.8235 - accuracy: 0.7190 - val_loss: 0.8001 - val_accuracy: 0.6787
```

```
Epoch 2/10
```

```
30/30 [=====] - 338s 11s/step - loss: 0.1233 - accuracy: 0.9663
```

```
Epoch 3/10
```

```
30/30 [=====] - 172s 6s/step - loss: 0.0842 - accuracy: 0.9770
```

```
Epoch 4/10
```

```
30/30 [=====] - 72s 2s/step - loss: 0.0482 - accuracy: 0.9857
```

```
Epoch 5/10
```

```
30/30 [=====] - 48s 2s/step - loss: 0.0165 - accuracy: 0.9967
```

```
Epoch 6/10
```

```
30/30 [=====] - 29s 942ms/step - loss: 0.0290 - accuracy: 0.9922
```

```
Epoch 7/10
```

```
30/30 [=====] - 19s 616ms/step - loss: 0.0169 - accuracy: 0.9943
```

```
Epoch 8/10
```

```
30/30 [=====] - 19s 638ms/step - loss: 0.0236 - accuracy: 0.9943
```

```
Epoch 9/10
```

```
30/30 [=====] - 21s 692ms/step - loss: 0.0165 - accuracy: 0.9949
```

```
Epoch 10/10
```

```
30/30 [=====] - 18s 584ms/step - loss: 0.0061 - accuracy: 0.9986
```

```
Out[15]:
```

```
In [16]: model.save('Realtime.h5')
```

TEST THE MODEL

```
In [17]: from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np
import cv2
```

