

```

2 import numpy as np
3 from tensorflow.keras.models import load_model
4 from tensorflow.keras.preprocessing import image
5 import os
6
7 class Video(object):
8     def __init__(self):
9         self.video = cv2.VideoCapture(0)
10        self.roi_start = (50, 150)
11        self.roi_end = (250, 350)
12        #self.model = load_model('asl_model.h5') # Execute Local Trained Model
13        self.model = load_model('realtime.h5') # Execute IBM Trained Model
14        self.index=['A','B','C','D','E','F','G','H','I']
15        self.y = None
16
17    def __del__(self):
18        k = cv2.waitKey(1)
19
20        self.video.release()
21
22    def get_frame(self):
23        ret, frame = self.video.read()
24        frame = cv2.resize(frame, (640, 480))
25        copy = frame.copy()
26        copy = copy[150:150+200, 50:50+200]
27        # prediction starts
28        cv2.imwrite('image.jpg', copy)
29        copy_img = image.load_img('image.jpg', target_size=(64, 64, 3))
30        x = image.img_to_array(copy_img)
31        x = np.expand_dims(x, axis=0)
32        pred = np.argmax(self.model.predict(x), axis=1)

```

```
27
28     x = image.img_to_array(copy_img)
29     x = np.expand_dims(x, axis=0)
30     pred = np.argmax(self.model.predict(x), axis=1)
31     self.y = pred[0]
32     cv2.putText(frame, 'The Predicted Alphabet is: ' + str(self.index[self.y]), (100, 50), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 0), 3)
33     ret, jpg = cv2.imencode('.jpg', frame)
34     return jpg.tobytes()
```