main_video_text.py

```
from keras.models import load_model
from time import sleep
from keras.preprocessing.image import img_to_array
from keras.preprocessing import image
import cv2
import numpy as np

face_classifier = cv2.CascadeClassifier('C:/Users/Pankaj/Desktop/Human_Emotion_Detection_Wapps/haarcascade_frontalface_default.xml')
classifier = load_model('C:/Users/Pankaj/Desktop/Human_Emotion_Detection_Wapps/model.h5')

emotion_labels = ['Angry', 'Disgust', 'Fear', 'Happy', 'Neutral', 'Sad', 'Surprise']

#cap = cv2.VideoCapture(0)
cap = cv2.VideoCapture("sample1.mp4")

desired_width = 640
desired_height = 480

while True:
    _, frame = cap.read()
    labels = []
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = face_classifier.detectMultiScale(gray)

    for (x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 255), 2)
        roi_gray = gray[y:y + h, x:x + w]
        roi_gray = cv2.resize(roi_gray, (48, 48), interpolation=cv2.INTER_AREA)

        if np.sum([roi_gray]) != 0:
            roi = roi_gray.astype('float') / 255.0
            roi = img_to_array(roi)
            roi = np.expand_dims(roi, axis=0)

            prediction = classifier.predict(roi)[0]
            label = emotion_labels[prediction.argmax()]
            label_position = (x, y - 10)
            cv2.putText(frame, label, label_position, cv2.FONT_HERSHEY_SIMPLEX, 1.5, (0, 255, 0), 3)
        else:
            cv2.putText(frame, 'No Faces', (30, 80), cv2.FONT_HERSHEY_SIMPLEX, 1.5, (0, 255, 0), 3)  #


    frame = cv2.resize(frame, (desired_width, desired_height))

```

```
    cv2.imshow('Emotion Detector', frame)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

```