Describe the preprocessing steps required on the image (if any).

--For test data, we can use the [Real-Life Drowsiness Dataset] created by a research team from the University of Texas at Arlington specifically for detecting multi-stage drowsiness. The dataset consists of around 30 hours of videos of 60 unique participants. From the dataset, I was able to extract facial landmarks from 44 videos of 22 participants. This allowes us to obtain a sufficient amount of data for both the alert and drowsy state.For each video, we used OpenCV (python module) to extract 1 frame per second

--Code:

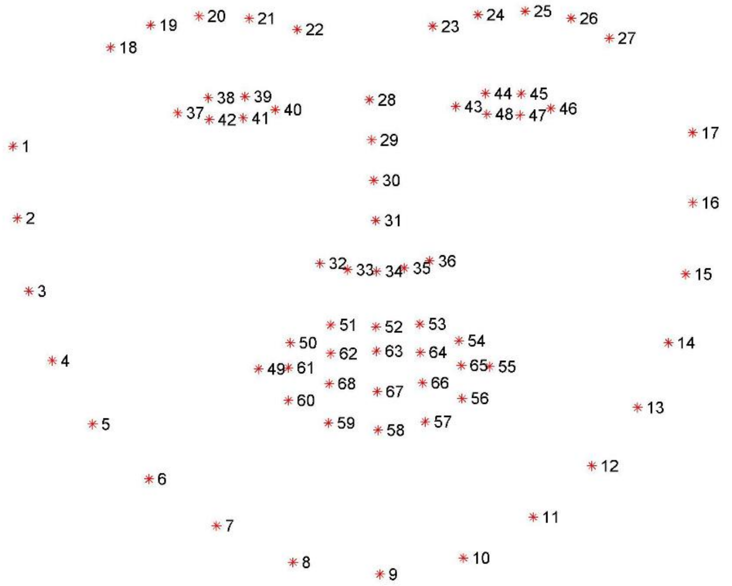
```
import cv2
data = []
labels = []
for j in [60]:
for i in [10]:
vidcap = cv2.VideoCapture('drive/My Drive/Fold5_part2/' + str(j) +'/' + str(i) +
'.mp4')
sec = 0
frameRate = 1
success, image = getFrame(sec)
count = 0
while success and count < 240:
landmarks = extract_face_landmarks(image)
if sum(sum(landmarks)) != 0:
count += 1
data.append(landmarks)
labels.append([i])
sec = sec + frameRate
sec = round(sec, 2)
success, image = getFrame(sec)
print(count)
else:
```

```
sec = sec + frameRate

sec = round(sec, 2)

success, image = getFrame(sec)

print("not detected")
```



There were 68 total landmarks per frame but we decided to keep the landmarks for the eyes and mouth only (Points 37–68). These were the important data points we used to extract the features for our model.

Describe techniques you would use to detect the eyes in the entire image.

There is a fixed connection among facial features. For example, the eyes are set in the upper part of the face and the mouth is located in the lower part of the face. In order to improve the accuracy and speed of detection, our algorithm determines the region of interest (ROI) of the eyes and mouth, and then detects the target on the ROI region. After obtaining the facial image, the upper half of the image is extracted and recorded as image I_1, the upper one-eighth of image I_1 is removed, and the lower seven-eighths of image I_1 is reserved and set as the eye ROI, as shown in Fig. 2(a). In this ROI, we use the EyeMap algorithm[23] to locate the eye region. This method builds two EyeMaps in the YCbCr space,[24] EyeMapC and EyeMapL; then, these two maps are combined into a single map. Experiments find high Cb components and low Cr components around the eyes, and EyeMapC is calculated as follows:

$$EyeMapC=[C^2_b+(255-C_r)^2+C_b/C_r]/3.$$

The values of C^2_b, $(255-C_r)^2$, and CbCr are normalized to the range [0, 255]. In addition, eyes contain bright and dark values in the luminance component; therefore, grayscale dilation and erosion with ball structuring elements are used to construct EyeMapL. EyeMapL is calculated as follows:

$$EyeMapL=Y(x,y)\oplus g(x,y)/Y(x,y)\Theta g(x,y),$$

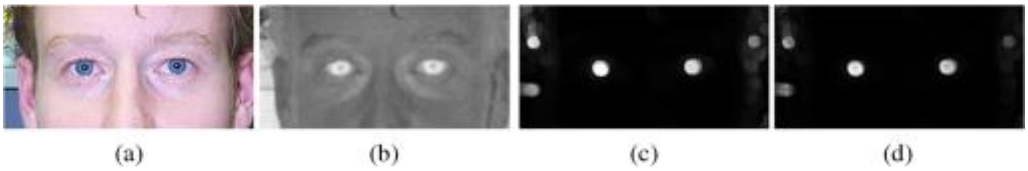
where g(x,y)g(x,y) represents the ball structuring element and $\oplus$ and Θ denote the grayscale dilation and erosion operations.

Then, EyeMapC is multiplied by EyeMapL to obtain EyeMap

EyeMap=EyeMapC×EyeMapL.

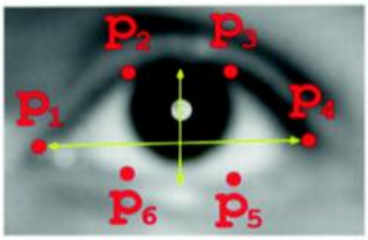
EyeMap of a typical image (from the California Polytechnic University color face database) is constructed as shown . Among them, the original eyes' ROI is as shown and EyeMapC, EyeMapL, and EyeMap are as shown in 2(b)–2(d), respectively.

llustration of EyeMap construction: (a) the original eye ROI region. (b) EyeMapC. (c) EyeMapL. (d) EyeMap.



- Describe techniques you would use to detect drowsiness in the eyes.

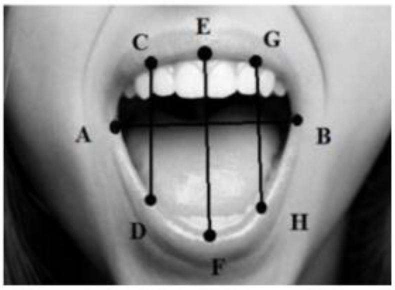
- ***Eye Aspect Ratio (EAR)***
- EAR, as the name suggests, is the ratio of the length of the eyes to the width of the eyes. The length of the eyes is calculated by averaging over two distinct vertical lines across the eyes as illustrated in the figure below.



$$\text{EAR} = \frac{\|p_2 - p_6\| + \|p_3 - p_5\|}{2\|p_1 - p_4\|}$$

Eye Aspect Ratio (EAR)

- Our hypothesis was that when an individual is drowsy, their eyes are likely to get smaller and they are likely to blink more. Based on this hypothesis, we expected our model to predict the class as drowsy if the eye aspect ratio for an individual over successive frames started to decline i.e. their eyes started to be more closed or they were blinking faster.
- ***Mouth Aspect Ratio (MAR)***
- Computationally similar to the EAR, the MAR, as you would expect, measures the ratio of the length of the mouth to the width of the mouth. Our hypothesis was that as an individual becomes drowsy, they are likely to yawn and lose control over their mouth, making their MAR to be higher than usual in this state.



$$\text{MAR} = \frac{|EF|}{|AB|}$$

Mouth Aspect Ratio (MAR)

- ***Pupil Circularity (PUC)***

- PUC is a measure complementary to EAR, but it places a greater emphasis on the pupil instead of the entire eye.

$$Circularity = \frac{4 * \pi * Area}{perimeter^2} \qquad Area = \left(\frac{\text{Distance(p2, p5)}}{2}\right)^2 * \pi$$

$$\begin{aligned} Perimeter = {} & Distance(p1,p2) + Distance(p2,p3) + Distance(p3,p4) + \\ & Distance(p4,p5) + Distance(p5,p6) + Distance(p6,p1) \end{aligned}$$

Pupil Circularity

- For example, someone who has their eyes half-open or almost closed will have a much lower pupil circularity value versus someone who has their eyes fully open due to the squared term in the denominator. Similar to the EAR, the expectation was that when an individual is drowsy, their pupil circularity is likely to decline.

- **Facial Action Coding System (FACS):**

  It is one of the most popular expression coding systems, used to code facial expressions. The facial expressions are decomposed to 46 component movements, this number corresponds to the number of an individual's facial muscles. Head motions can be detected through an automatic eye tracking and an accelerometer. FACS is also capable of discovering new patterns based on emotional states.

- **Support Vector Machines (SVM):**

  For face detection, the Haar feature algorithm is used. Each feature is classified by a Haar feature classifier. It takes the captured face as input, detected face as output. Detects eyes image from this detected face and this detected eye is sent to the ML algorithm for further ML processes. SVM is used to identify whether the eyes are closed or open. An SVM can be trained to detect the face and see whether the eyes are shut to open and then decide to trigger the alarm or not. The training set has a set of images that have eyes shut and some set of images that have eyes open. When the model is built, it will be used to classify any new pre-processed eye-image.

  An ML classifier is built using this algorithm to classify the pre-processed eye image. SVMs can efficiently solve linear or non-linear classification problems. It maximizes the margin around the separating hyperplane and then can find an optimal classifier.

- **Convolution Neural Network (CNN):**

This method uses layers of spatial convolutions that are well suited for images, which exhibit strong spatial convolutions. CNNs produce more accurate results in comparison with SVMs and HMMs. Voila and Jones algorithm can be used to detect faces. Images are cropped to square images and are fed to the first layer of the network that has filters. The output is passed further for classification.

The blinking of only the one eye, either right or left, is detected so that the memory for detecting both the eyes is saved. We only need one eye blinking information because we blink both the eyes together.

The programming in all the methods is done with Python using a library called OpenCV. OpenCV has inbuilt, pre-trained classifiers for features like face, eyes and smiles.

Conclusion :

- ***Mouth aspect ratio over Eye aspect ratio (MOE)***

- Finally, we decided to add MOE as another feature. MOE is simply the ratio of the MAR to the EAR.

$$MOE = \frac{MAR}{EAR}$$

Mouth Over Eye Ratio (MOE)

- The benefit of using this feature is that EAR and MAR are expected to move in opposite directions if the state of the individual changes. As opposed to both EAR and MAR, MOE as a measure will be more responsive to these changes as it will capture the subtle changes in both EAR and MAR and will exaggerate the changes as the denominator and numerator move in opposite directions. Because the MOE takes MAR as the numerator and EAR as the denominator, our theory was that as the individual gets drowsy, the MOE will increase.