

```
1 import cv2
2 import dlib
3 from scipy.spatial import distance
4 from imutils import face_utils
5
6 cap = cv2.VideoCapture(0)
7
8 detector = dlib.get_frontal_face_detector()
9 predictor = dlib.shape_predictor(r"C:\Users\rkssp\Downloads\facial-landmarks-recognition-master\
shape_predictor_68_face_landmarks.dat")
10
11 def eye_aspect_ratio(eye):
12     A = distance.euclidean(eye[1], eye[5])
13     B = distance.euclidean(eye[2], eye[4])
14
15     C = distance.euclidean(eye[0], eye[3])
16     eye = (A + B) / (2.0 * C)
17
18     return eye
19
20 count = 0
21 total = 0
22
23 while True:
24     success, img = cap.read()
25     imgGray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
26     faces = detector(imgGray)
27
28     for face in faces:
29         landmarks = predictor(imgGray, face)
30
31         landmarks = face_utils.shape_to_np(landmarks)
32         leftEye = landmarks[42:48]
33         rightEye = landmarks[36:42]
34
35         leftEye = eye_aspect_ratio(leftEye)
36         rightEye = eye_aspect_ratio(rightEye)
37
38         eye = (leftEye + rightEye) / 2.0
39
40         if eye<0.3:
41             count+=1
42         else:
43             if count>=3:
44                 total+=1
45
46                 count=0
```

47

```
48 cv2.putText(img, "Blink Count: {}".format(total), (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (0, 0, 255), 2)
```

49

```
cv2.imshow('Video',img)
```

50

```
if cv2.waitKey(1) & 0xff==ord('q'):
```

51

```
    break
```