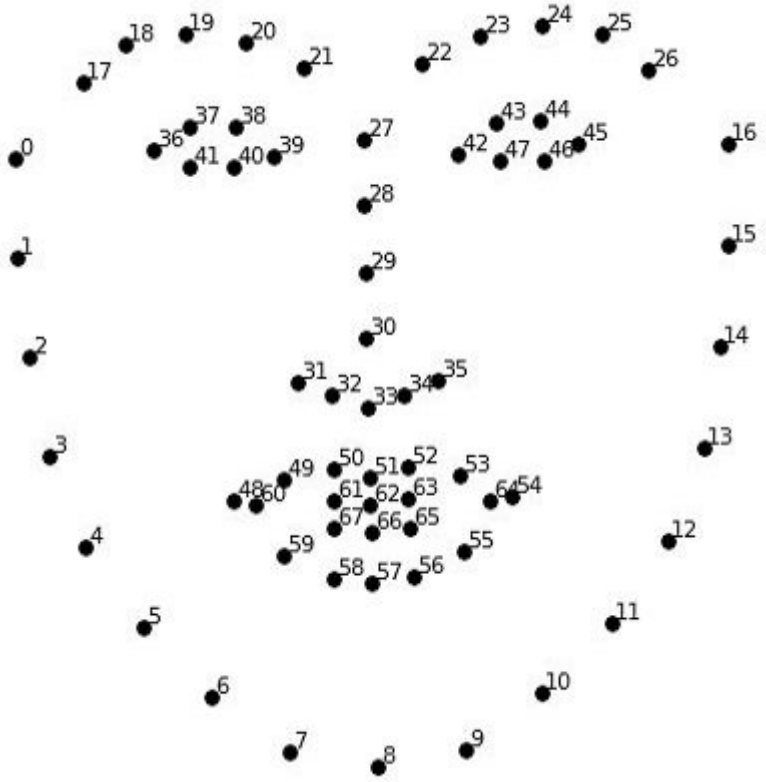
1.产生摄像头检测

```
import cv2

cap = cv2.VideoCapture(0)

while True:
    _, frame = cap.read()
    cv2.imshow("Frame", frame)

    key = cv2.waitKey(1)
    if key == 27:#if the key is esc, then quit
        break

cap.release()
cv2.destroyAllWindows()
```

2.检测人脸

```
import cv2
import dlib
cam = cv2.VideoCapture(0)
detect = dlib.get_frontal_face_detector()

while True:
    _, frame = cam.read()

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = detect(gray)

    for face in faces:
        print(face)
        cv2.rectangle(frame,(face.left(),face.top()),
(face.right(),face.bottom()),(0,255,255),3)


    cv2.imshow("Frame", frame)

    key = cv2.waitKey(1)
    if key == 27:#if the key is esc, then quit
        break

cam.release()
cv2.destroyAllWindows()
```

3.找到眼睛




```
import cv2
import dlib
cam = cv2.VideoCapture(0)
detect = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")
while True:
    _, frame = cam.read()

    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = detect(gray)

    for face in faces:
        # print(face)
        # cv2.rectangle(frame,(face.left(),face.top()),
(face.right(),face.bottom()),(0,255,255),3)

        landmarks = predictor(gray,face)
        #use landmarks.part(x) to get the coordinate
        for p in range(36,42):#draw points from 36 - 41
            x = landmarks.part(p).x
            y = landmarks.part(p).y
            cv2.circle(frame,(x,y),3,(0,0,255),2)
        for p in range(42,48):
            x = landmarks.part(p).x
            y = landmarks.part(p).y
            cv2.circle(frame,(x,y),3,(0,0,255),2)


    cv2.imshow("Frame", frame)

    key = cv2.waitKey(1)
    if key == 27:#if the key is esc, then quit
```

```
        break

cam.release()
cv2.destroyAllWindows()
```

4.判断眼睛是否眨眼

具体思路：判断比率

```
import cv2#opencv python
import dlib
from math import hypot
import math

def distance_between_landmarks(n1, n2, n3, n4, landmarks):
    x1 = landmarks.part(n1).x
    y1 = landmarks.part(n1).y
    x2 = landmarks.part(n2).x
    y2 = landmarks.part(n2).y
    x3 = landmarks.part(n3).x
    y3 = landmarks.part(n3).y
    x4 = landmarks.part(n4).x
    y4 = landmarks.part(n4).y

    x_mid, y_mid = (x1 + x2) / 2, (y1 + y2) / 2
    x_mid2, y_mid2 = (x3 + x4) / 2, (y3 + y4) / 2

    return math.hypot(x_mid2 - x_mid, y_mid2 - y_mid)


detect = dlib.get_frontal_face_detector()
predictor = dlib.shape_predictor("shape_predictor_68_face_landmarks.dat")

cap = cv2.VideoCapture(0)
while True:
    _, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    faces = detect(gray)

    #(0,0,0) 0-255 red green blue
    for face in faces:

        landmarks = predictor(gray,face)

        #36-39, 42-45
        left_row_length = hypot((landmarks.part(36).x-landmarks.part(39).x),
(landmarks.part(36).y-landmarks.part(39).y))
        right_row_length = hypot((landmarks.part(42).x-landmarks.part(45).x),
(landmarks.part(42).y-landmarks.part(45).y))
        left_row_length = distance_between_landmarks(37,38,40,41,landmarks)
        right_col_length = distance_between_landmarks(43,44,46,47,landmarks)
        #TODO: compute the ratio of left_row_length,left_row_length,judge when is
blink by experiment

    #face vector<vector{float,float}> face[0]
```

```
    cv2.imshow("Frame", frame)

    key = cv2.waitKey(1)
    if key == 27:#if the key is esc, then quit
        break

cap.release()
cv2.destroyAllWindows()
```