CODE FOR FACIAL EXPRESSION DETECTION (GASPING)

```
import cv2
import numpy as np

# Open webcam
cap = cv2.VideoCapture(0)

# Parameters for gasping detection
MOUTH_OPEN_THRESHOLD = 50  # Threshold for detecting mouth openness



def calculate_mouth_openness(mouth_region):
    """
    Calculate the mouth openness based on the height of the mouth contour.
    :param mouth_region: Contour of the mouth region.
    :return: Height of the mouth contour.
    """
    _, _, mouth_width, mouth_height = cv2.boundingRect(mouth_region)
    return mouth_height



while True:
    # Capture frame-by-frame
    ret, frame = cap.read()
    if not ret:
        break
```

```
# Convert frame to grayscale
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

# Detect faces in the frame
face_cascade = cv2.CascadeClassifier(cv2.data.haarcascades +
                                     "haarcascade_frontalface_default.xml")
faces = face_cascade.detectMultiScale(gray,
                                      scaleFactor=1.1,
                                      minNeighbors=5,
                                      minSize=(30, 30))

# Iterate through detected faces
for (x, y, w, h) in faces:
    # Extract region of interest (ROI) for the face
    roi_gray = gray[y:y + h, x:x + w]

    # Approximate mouth region as a polygon (not exact)
    mouth_region = np.array([
        (x + w // 4, y + 3 * h // 4),
        (x + 3 * w // 4, y + 3 * h // 4),
        (x + w // 2, y + h)
    ], np.int32)

    # Calculate mouth openness
    mouth_height = calculate_mouth_openness(mouth_region)

    # Check if gasping based on mouth threshold
    if mouth_height > MOUTH_OPEN_THRESHOLD:
```

```
            cv2.putText(frame, "Gasping", (x, y - 10),
                        cv2.FONT_HERSHEY_SIMPLEX, 0.9, (0, 0, 255), 2)
            cv2.rectangle(frame, (x, y), (x + w, y + h),
                          (0, 0, 255), 2)
        else:
            cv2.putText(frame, "Not Gasping", (x, y - 10),
                        cv2.FONT_HERSHEY_SIMPLEX, 0.9, (0, 255, 0), 2)
            cv2.rectangle(frame, (x, y), (x + w, y + h),
                          (0, 255, 0), 2)

    # Display the resulting frame
    cv2.imshow("Gasping Detection", frame)

    # Break the loop if 'q' is pressed
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

# Release video capture object and close windows
cap.release()
cv2.destroyAllWindows()
```