

FACE & EYE DETECT

# Face detection

เทคโนโลยีที่ถูกสร้างขึ้นเพื่อจดจำ และตรวจจับใบหน้าของมนุษย์ โดยผ่านการเก็บข้อมูล ไบโอเมตริกซ์ (Biometric Data) หรือข้อมูลอัตลักษณ์ของบุคคล ที่ได้จาก กล้องวงจรปิด รูปภาพ หรือวิดีโอ

# Biometric

ลักษณะของมนุษย์ที่สร้างเอกลักษณ์ของแต่ละบุคคล เช่น ลักษณะบนใบหน้า ดวงตาลายนิ้วมือ หรือ แม้กระทั่งการเต้นของหัวใจ ซึ่งลักษณะเหล่านี้สามารถนำไปใช้ในการกำหนดว่าคุณเป็นใครได้

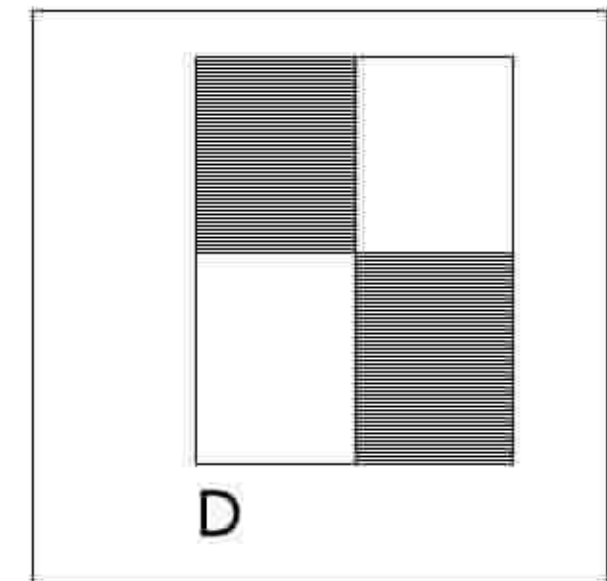
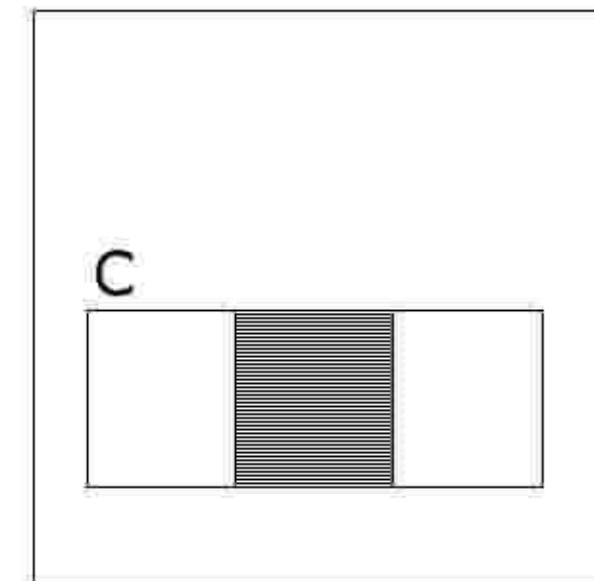
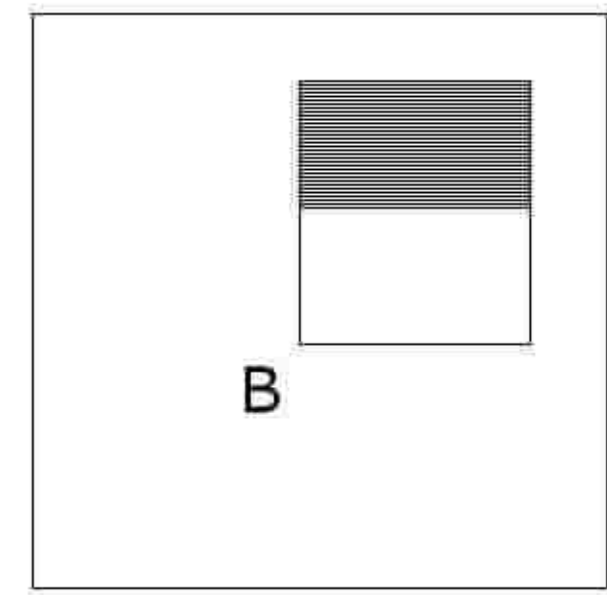
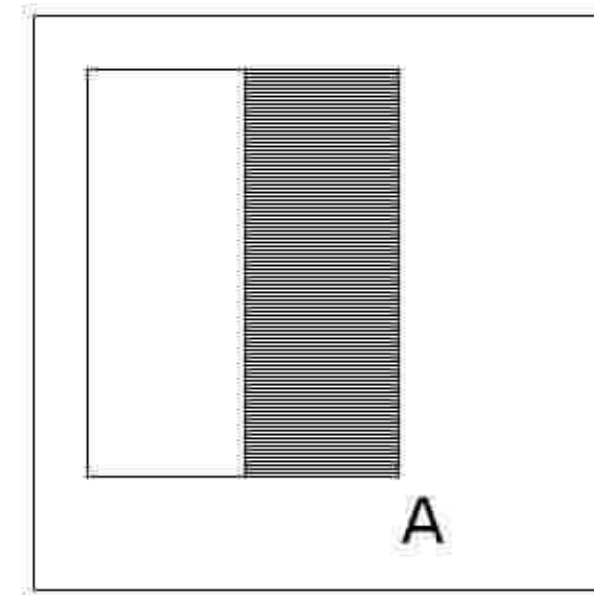
# HAARCASCADES

เป็น feature หนึ่งในไลบรารี OpenCV ที่ใช้ detect วัตถุหรือส่วน ต่างๆ ของภาพ โดยสามารถ download ได้ใน

```
wget https://github.com/opencv/opencv/blob/master/data/haarcascades/haarcascade_frontalface_default.xml
```

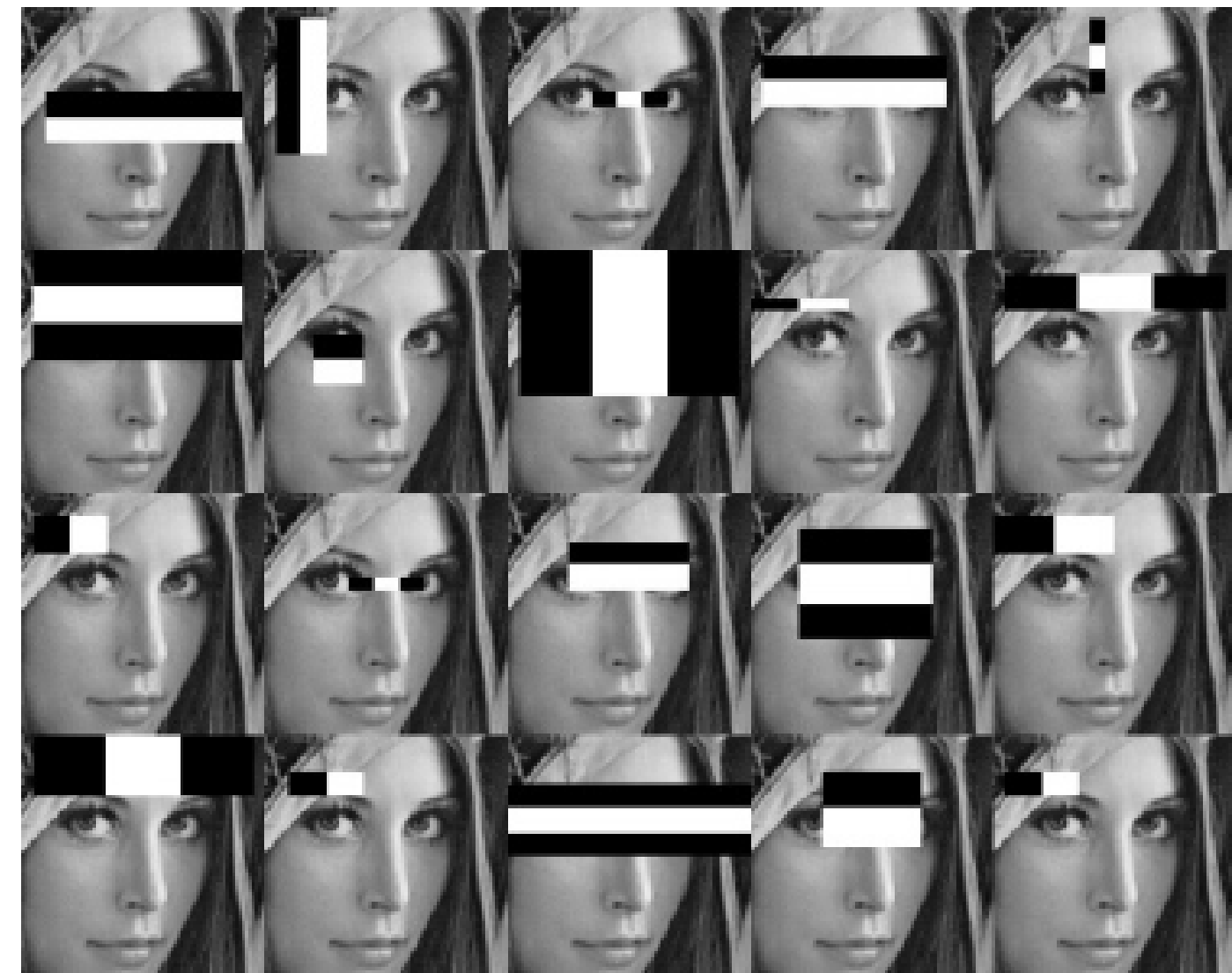
# HAARCASCADES

“Fast computation of Haar-like features” วิธีคือ การแบ่งพื้นที่ของภาพด้วยรูปแบบสำเร็จรูป 4 แบบที่เรียกว่า Haar-like features ไว้ดังรูป ( A - B - C - D )



# HAARCASCADES

จากนั้นจึงคำนวณตารางสีของภาพ  
โดยการ อินทิเกรตตารางสีที่อยู่ใน  
พื้นที่อยู่ในช่องสีขาว ลบด้วยตารางสี  
ที่อยู่ในพื้นที่อยู่ในช่องสีดำ ใน  
กระบวนการสร้าง รูปอินทิกรัล แล้ว  
นำผลที่ได้ไปใช้ใน Adaboost



# หลักการเขียนโปรแกรม

อ่านไฟล์ในการจำแนก (classification) ใบหน้า ซึ่งได้รับการ train มาแล้วว่าตำแหน่งใดเป็น ใบหน้า ในที่นี้คือ haarcascade\_frontalface\_default.xml โดยใช้คำสั่ง **cv2.CascadeClassifier("haarcascade\_frontalface\_default.xml")** และกำหนด ชื่อ ตัวแปรให้กับการอ่าน

# หลักการเขียนโปรแกรม

ทำภาพที่มีอยู่จากภาพสีให้เป็นภาพขาวเทา เนื่องจากจะทำให้การประมวลผลภาพนั้น ทำได้รวดเร็ว  
โดยใช้คำสั่ง `cv.cvtColor(img,cv2.COLOR_BGR2GRAY)`

จำแนกใบหน้า โดยการใช้คำสั่ง

`face_cascade.detectMultiScale(gray_img,scaleFactor,minNeighbors)`



# SCALEFACTOR และ MINNEIGHBORS

- **SCALEFACTOR** คือ ค่าเปอร์เซ็นต์ในการลดขนาดภาพ ตัวอย่างเช่น
  - 1.05 หมายถึงภาพถูก ลดขนาดลง5%โอกาสที่จะจำแนกหน้ามีมากขึ้น
  - 1.8 หมายถึงภาพจะถูก ลดขนาดลง80%โอกาสที่จะจำแนกหน้ามีน้อยลง
- **MINNEIGHBORS** คือ ค่าที่ระบุว่าแต่ละสี่เหลี่ยมควรมีโอกาสที่จะเป็น ใบหน้า จำนวนมากน้อยเท่าใด

# หลักการเขียนโปรแกรม

**เขียนกรอบสี่เหลี่ยมให้กับบริเวณที่พบใบหน้า**รับค่า tuple (x,y) (w,h) จากการ  
ตัวแปรที่ใช้การจำแนกใบหน้าซึ่งก็คือ face\_detect และนำมาวาดรูปสี่เหลี่ยม



```
1 import cv2 as cv
2
3 img = cv.imread("image/125442187_2075825349219537_8954625077520203482_n.jpg")
4
5 face_cascade = cv.CascadeClassifier("opencv-master/data/haarcascades/haarcascade_frontalface_default.xml")
6 gray_img = cv.cvtColor(img, cv.COLOR_RGB2GRAY)
7
8 scale = 1.1
9 minNeighbor = 3
10 face_detect = face_cascade.detectMultiScale(gray_img, scale, minNeighbor)
11 x, y, w, h = face_detect[0]
12 print(face_detect)
13 cv.rectangle(img, (x, y), (x+w, y+h), (0, 255, 0), thickness=10)
14 cv.imshow("face detect", img)
15 cv.waitKey(0)
16 cv.destroyAllWindows()
```

# จับหน้าหลายคน

จะใช้คำสั่ง for ในการวาดรูปสี่เหลี่ยม โดยจะวาดทั้งหมด ใน tuple



```
1 for (x, y, w, h) in face_detect:  
2     cv.rectangle(img, (x,y), (x+w,y+h), (0,255,0), thickness=2)
```

# การตรวจจับหน้าใน vdo

การตรวจจับหน้าจะเป็นการตรวจจับทีละภาพ ดังนั้นการตรวจจับหน้าจากวิดีโอจะเป็นการ ตรวจจับทีละ frame และวาดลงไปในภาพ

```
1 while (cap.isOpened()):
2     chack,frame = cap.read()
3     if chack :
4         gray_cap = cv.cvtColor(frame,cv.COLOR_RGB2GRAY)
5         face_detect = face_cascade.detectMultiScale(gray_cap)
6         for (x, y, w, h) in face_detect:
7             cv.rectangle(frame,(x,y),(x+w,y+h),(0,255,0),thickness=2)
8         cv.imshow("face detect",frame)
9         if cv.waitKey(1) & 0xFF == ord("q"):
10             break
11     else :
12         break
```

EYE DETECT

# EYE DETECT

จะเหมือนการตรวจจับใบหน้าแต่ใช้ **classification**  
เป็น `haarcascade_eye_tree_eyeglasses.xml`

# EYE DETECT

```
● ● ●
1 import cv2 as cv
2
3 img = cv.imread("image/125442187_2075825349219537_8954625077520203482_n.jpg")
4
5 eye_cascade = cv.CascadeClassifier("opencv-master/data/haarcascades/haarcascade_eye_tree_eyeglasses.xml")
6 gray_img = cv.cvtColor(img,cv.COLOR_RGB2GRAY)
7
8 scale = 1.1
9 minNeighbor = 10
10 eye_detect = eye_cascade.detectMultiScale(gray_img,scale,minNeighbor)
11
12 print(eye_detect)
13 for (x, y, w, h) in eye_detect:
14     cv.rectangle(img,(x,y),(x+w,y+h),(0,0,255),thickness=2)
15 cv.imshow("eye detect",img)
16 cv.waitKey(0)
17 cv.destroyAllWindows()
```



# งาน

ให้นักเรียนตรวจจับใบหน้าและดวงตาพร้อม โดยใช้กล้อง webcam