

Date: 20221008 รหัสนักศึกษา B6321451 ชื่อ-สกุล ขวัญจิรา พันธุเกตุ

Wk0911.20221005,08 -- Object Detection by Keypoint Matching

Wk0912.20221005,08 -- Gender and Age Detection by CNN

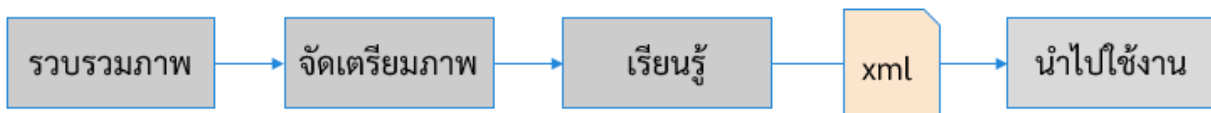
Wk0913.20221005,08 -- Picture to Video

- ทำเอกสารให้สมบูรณ์
- กำหนดชื่อไฟล์ตามรูปแบบนี้ “B3601234-Week09-นายวิชัย ศรีสุรักษ์.pdf”
- MC ส่งงาน Class Check ก่อน 21:00น วันพุธที่ 5 ตค 65 ที่ <https://forms.gle/NtKVrmxHXiNh8nkH9>
- MC ส่งงาน Homework ก่อน 06:00น วันพุธที่ 12 ตค 65 ที่ <https://forms.gle/S8tTMEipmogFtG9F6>
- PC ส่งงาน Class Check ก่อน 17:00น วันเสาร์ที่ 8 ตค 65 ที่ <https://forms.gle/yuAfkgbWEePyz3RK6>
- PC ส่งงาน Homework ก่อน 06:00น วันเสาร์ที่ 15 ตค 65 ที่ <https://forms.gle/twTys2TVy4ngFxZcA>

การตรวจจับวัตถุด้วยการใช้แบบจำลองของฮาร์(Haar)

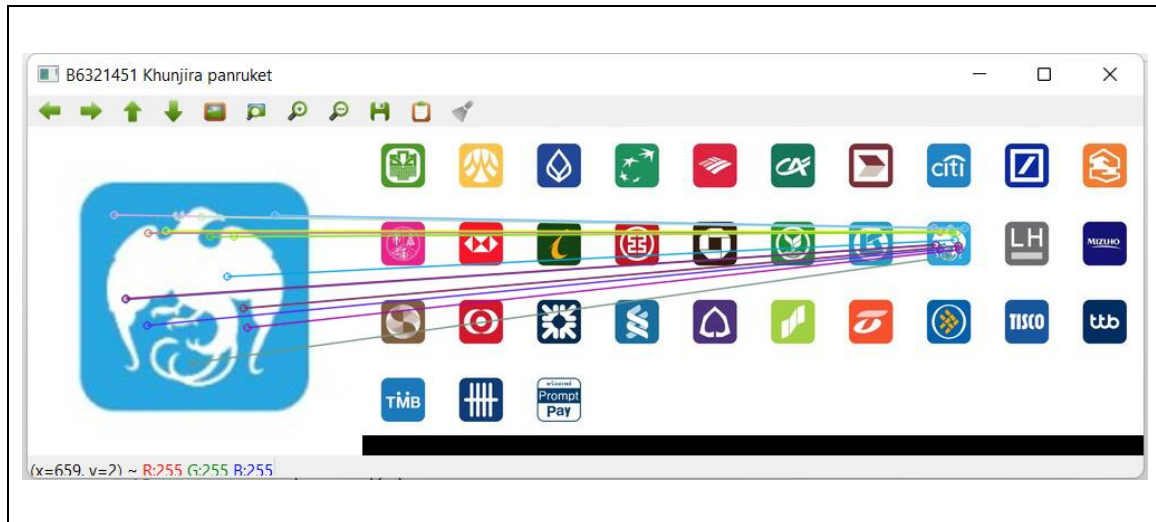
"การตรวจจับวัตถุโดยใช้ตัวแบ่งประเภทแบบเรียงซ้อน" เป็นการตรวจหาวัตถุอย่างรวดเร็วโดยใช้วิธีจัดเรียงคุณลักษณะเด่นอย่างง่าย ซึ่งเป็นหนึ่งในวิธีการเรียนรู้ของเครื่องจักร โดยการสอนภาพลักษณะที่ดีและไม่ดี จากนั้นจะใช้เพื่อตรวจนำไปตรวจจับวัตถุในภาพอื่นๆ

ประกอบด้วยขั้นตอนหลักสองขั้นตอนคือขั้นตอนการเรียนรู้และการนำไปใช้งานสำหรับการตรวจหาวัตถุ การตรวจหาวัตถุด้วยการใช้แบบจำลองของฮาร์ (Haar) สามารถอธิบายถึงขั้นตอนในการตรวจหาวัตถุ โดยการทำงานของฟังก์ชันที่จำเป็นในการเรียนรู้ตัวจำแนกประเภทแบบเรียงซ้อน ผ่านการดำเนินการขั้นตอนต่าง ๆ ดังนี้

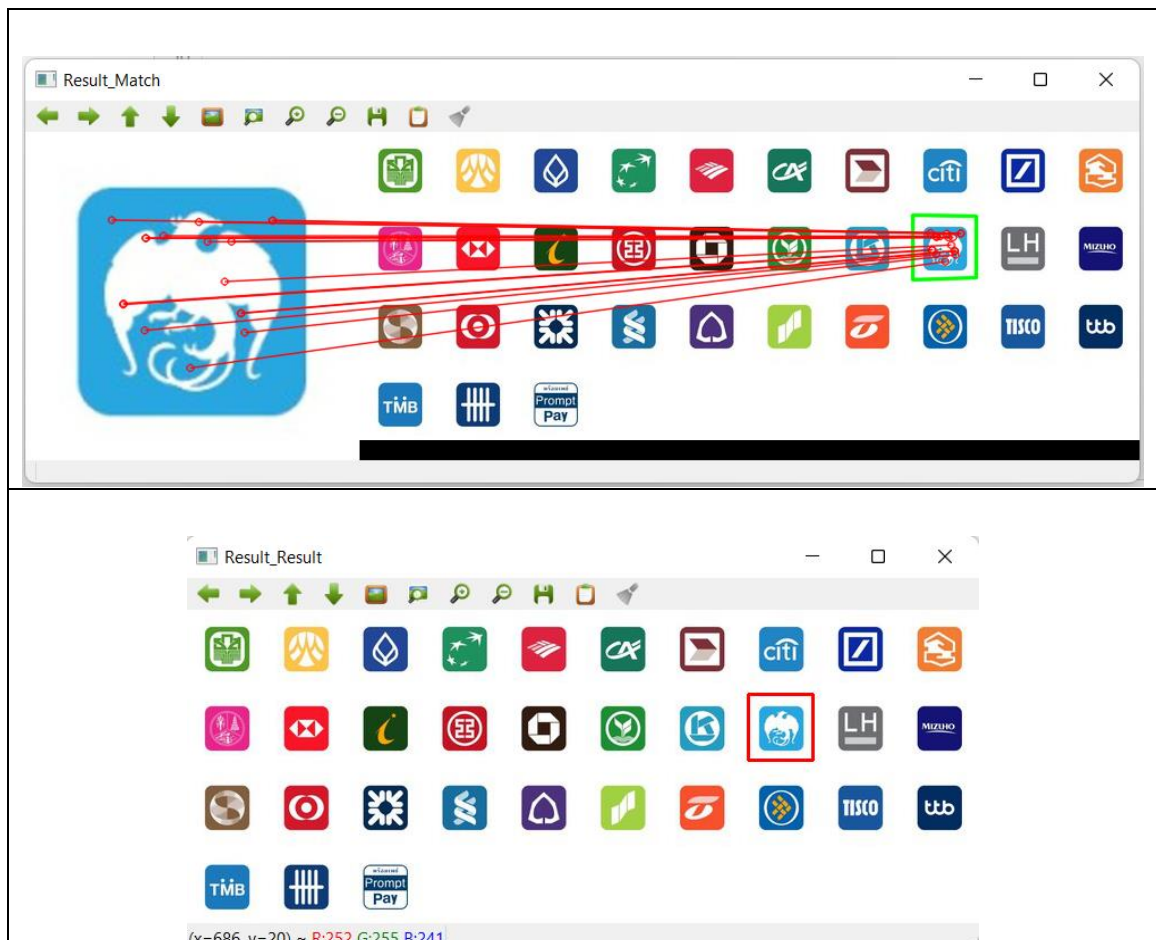


Wk0911.20221005,08 -- Object Detection by Keypoint Matching

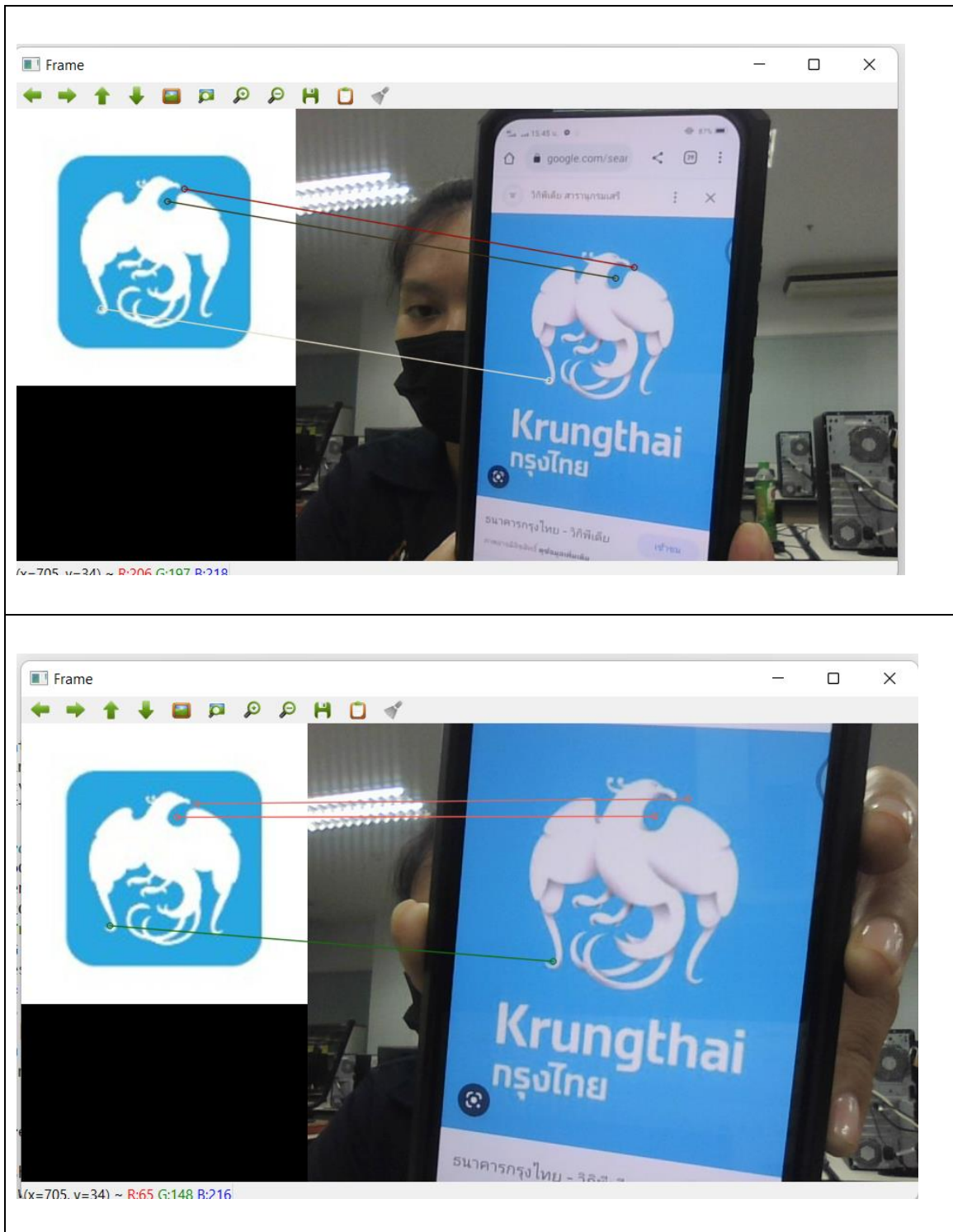
- กิจกรรม-101 -- ให้ทดสอบกับตัวอย่างอื่นๆ ที่เราสนใจ



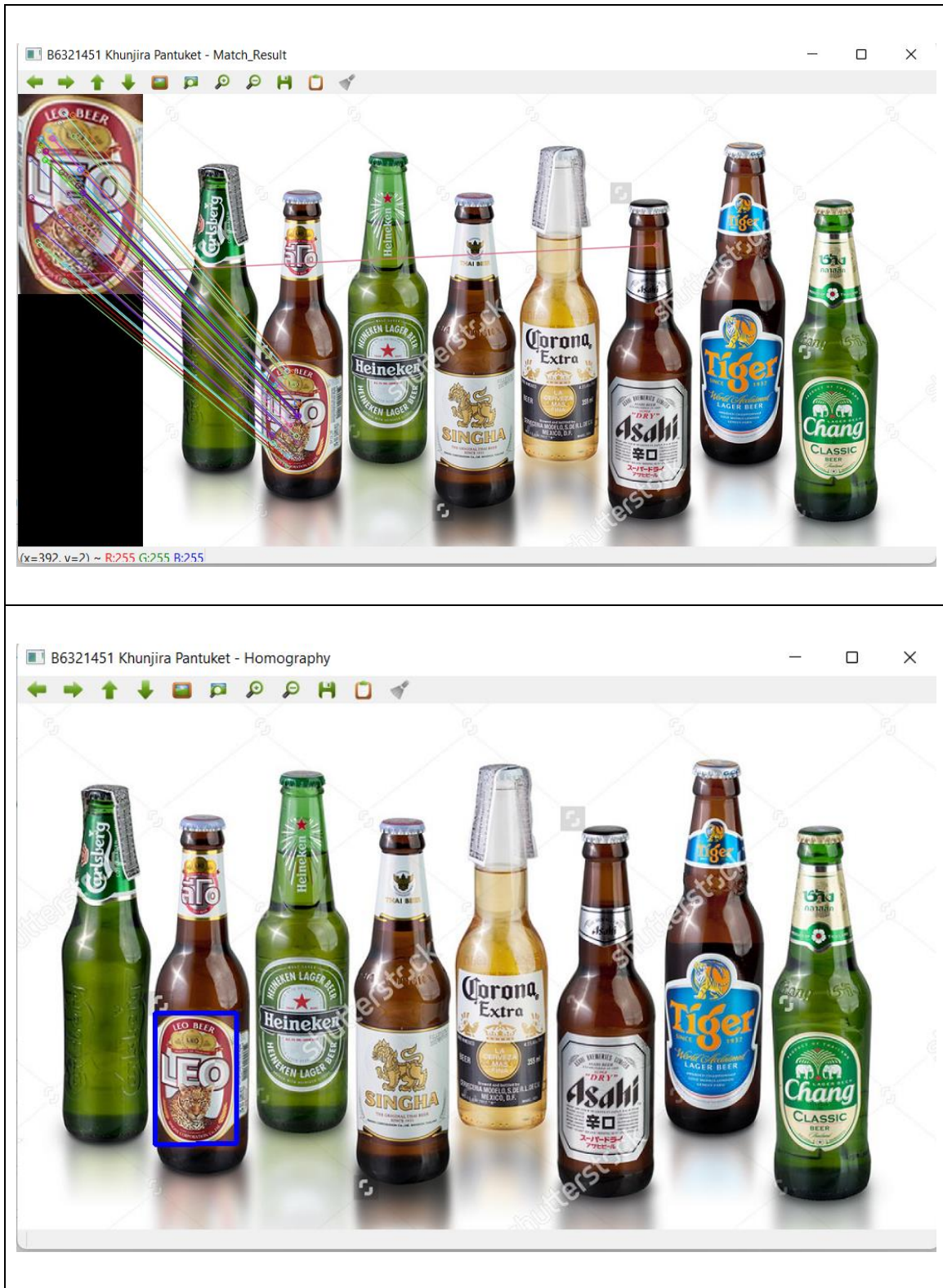
- กิจกรรม-102 -- ให้ตีกรอบผลลัพธ์การทำงาน



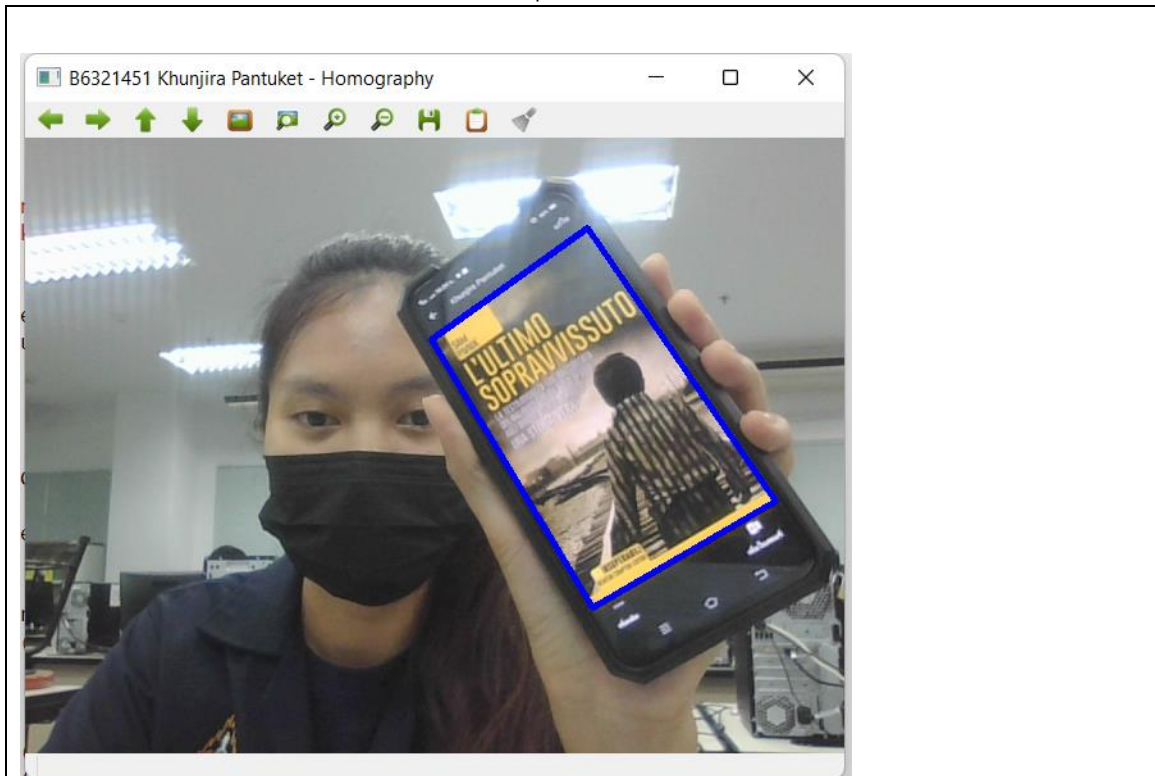
3. กิจกรรม-103 -- การตรวจจับคู่คุณลักษณะด้วยคำอธิบายคุณลักษณะ SIFT ของภาพจากกล้องวิดีโอ



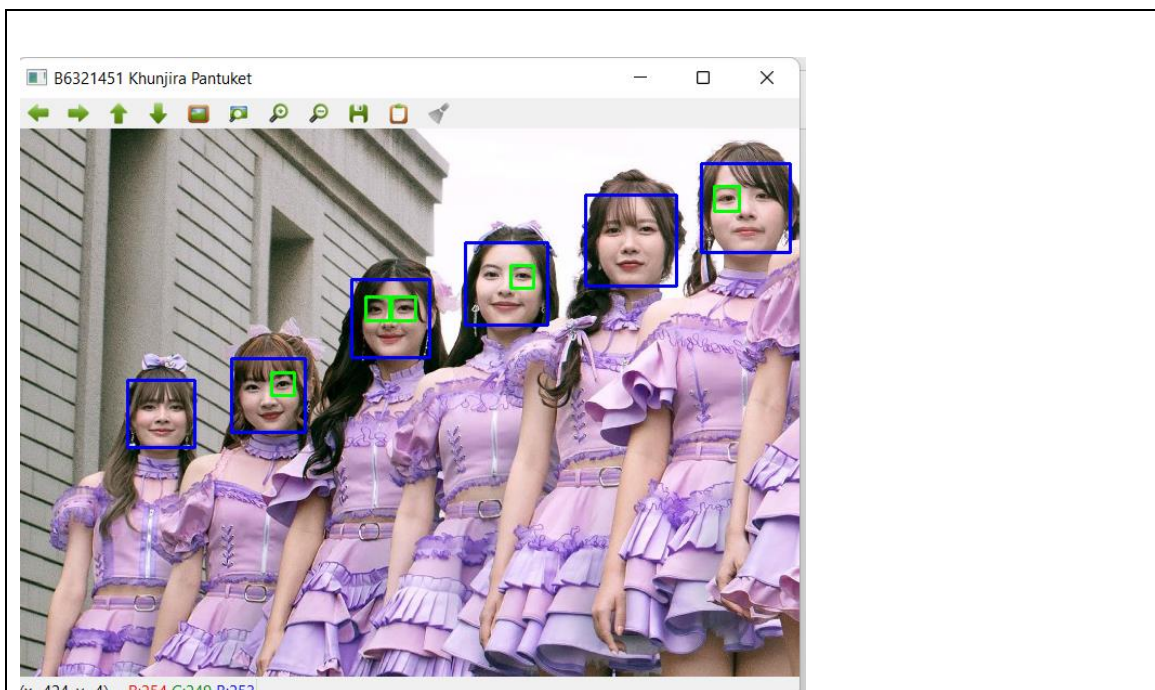
4. กิจกรรม-201 -- ค้นหา LEO จากภาพตัวอย่างด้านล่างนี้



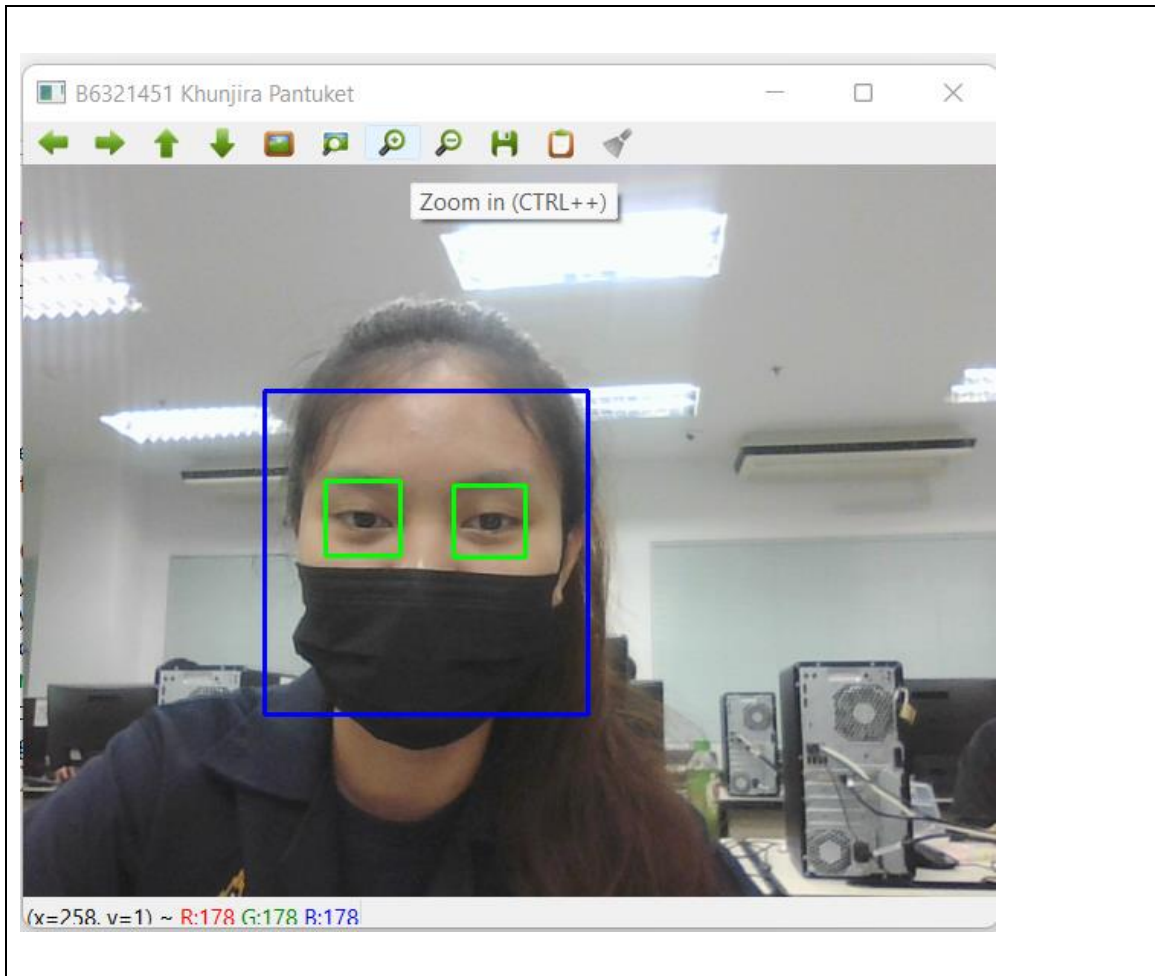
5. กิจกรรม-202 -- ให้ทำโปรแกรมเพื่อการตรวจจับวัตถุจากกล้องวีดีโอด้วย Homography-Detection



6. Mission-301 -- ตรวจจับใบหน้าและตาโดยเลือกภาพที่มีบุคคล 5-10 คน

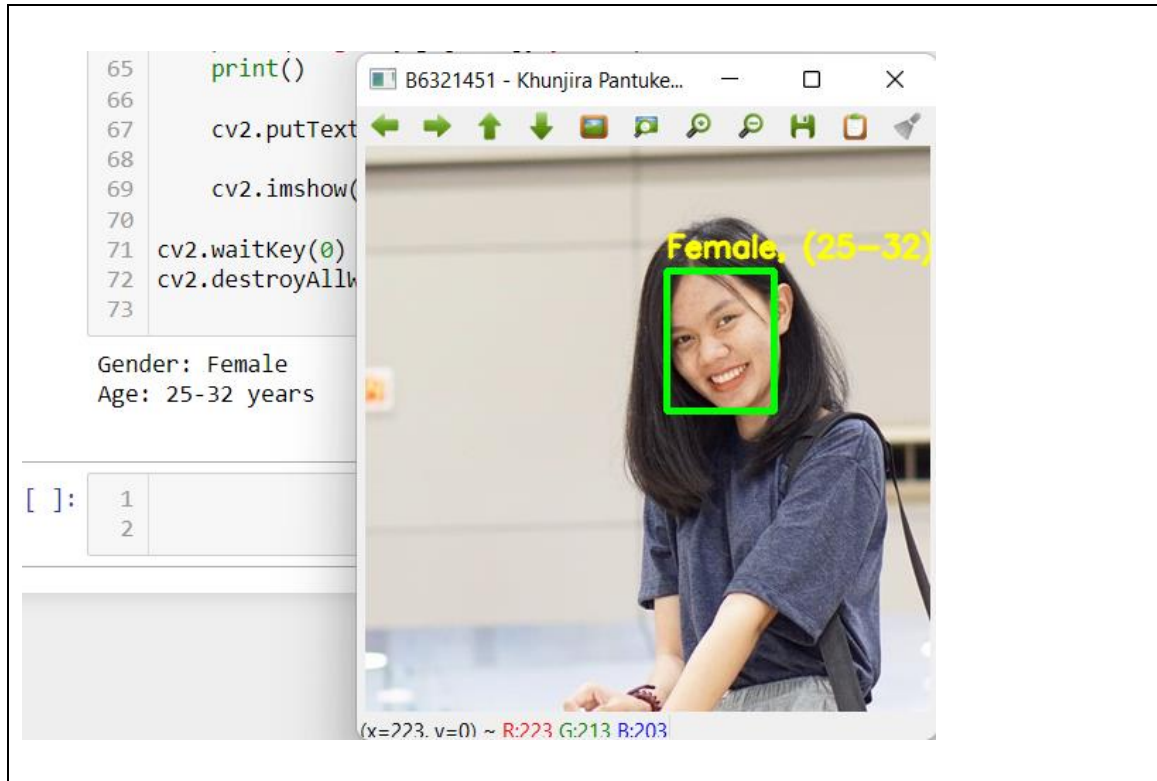


7. Mission-302 – ตรวจจับใบหน้าและตาจากภาพวิดีโอ

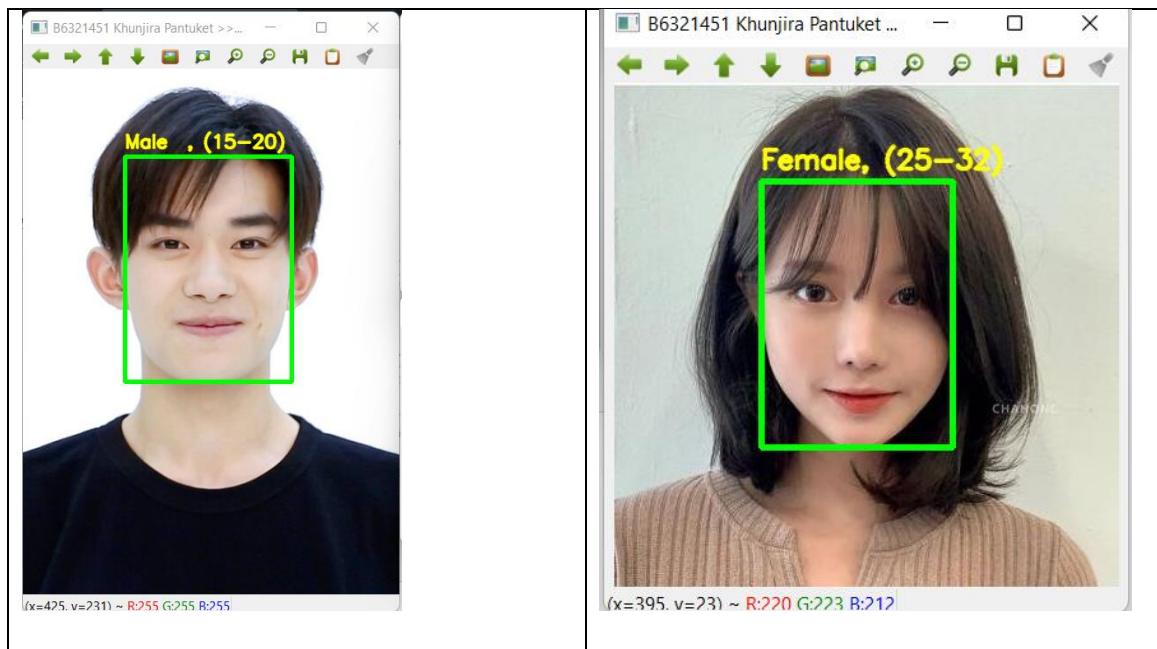


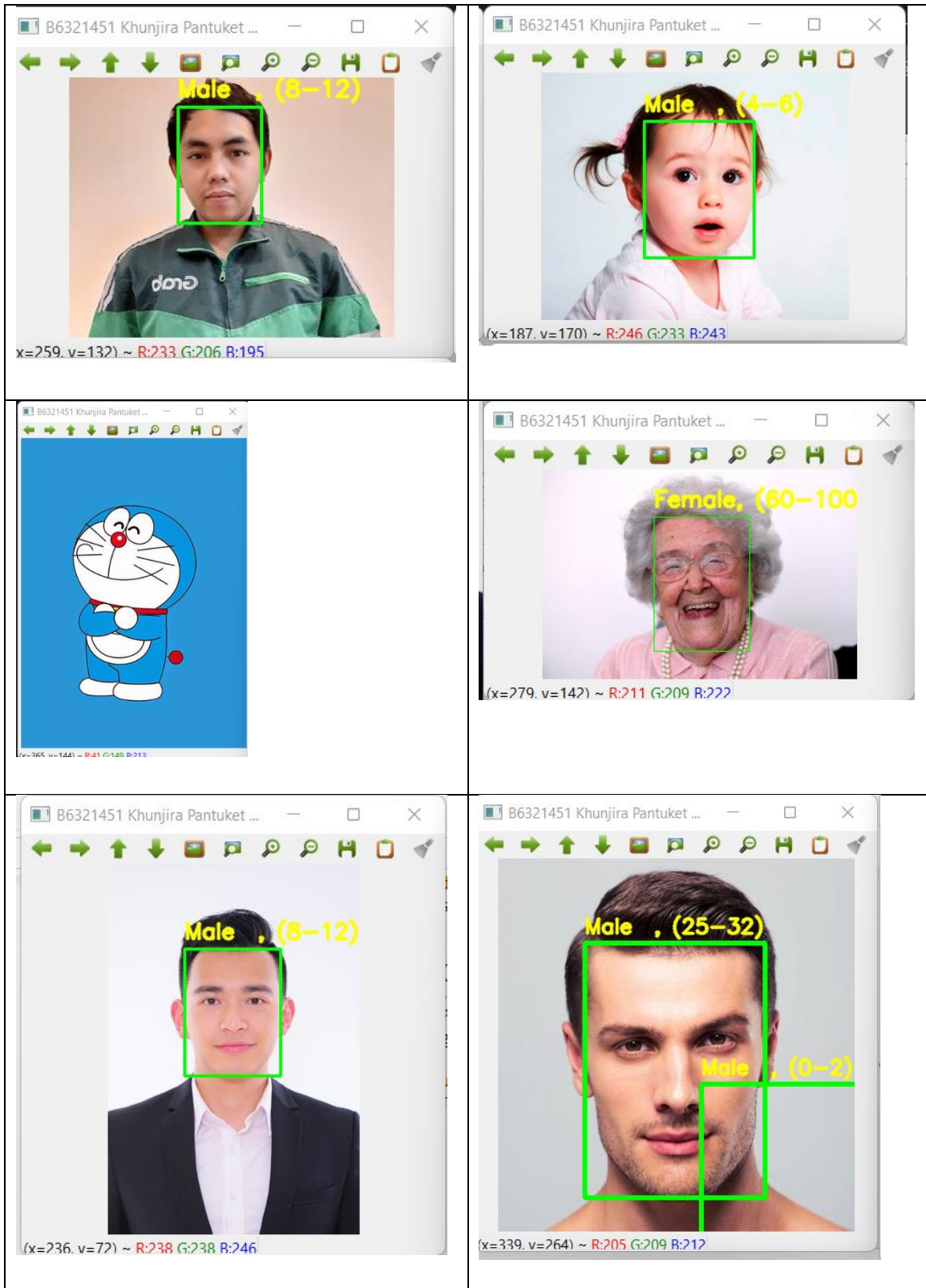
Wk0912.20221005,08 -- Gender and Age Detection by CNN

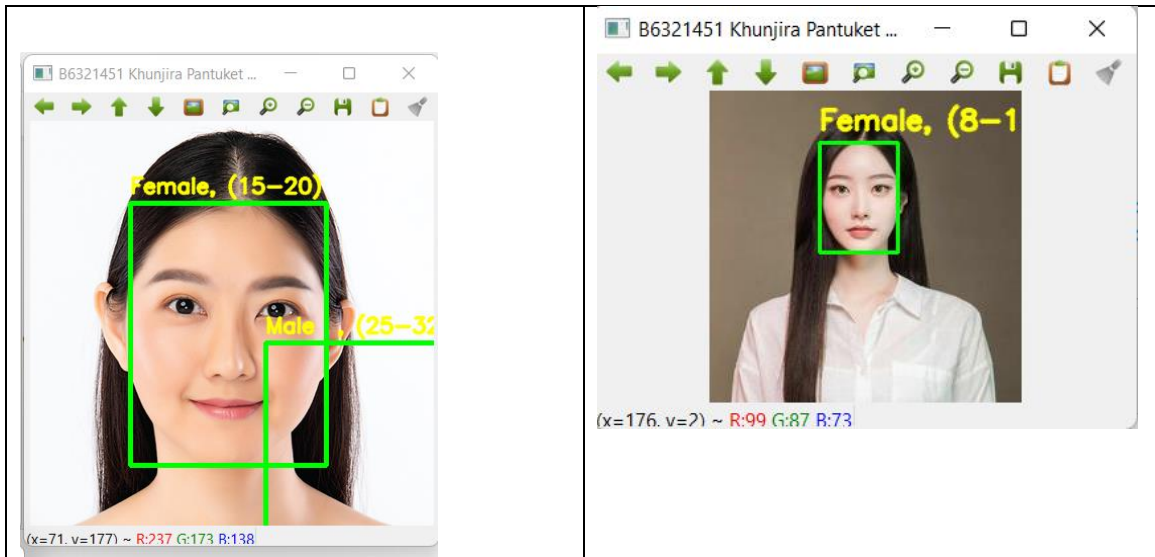
1. Test1: Single Fileทดสอบกับภาพนิ่งของตัวเอง



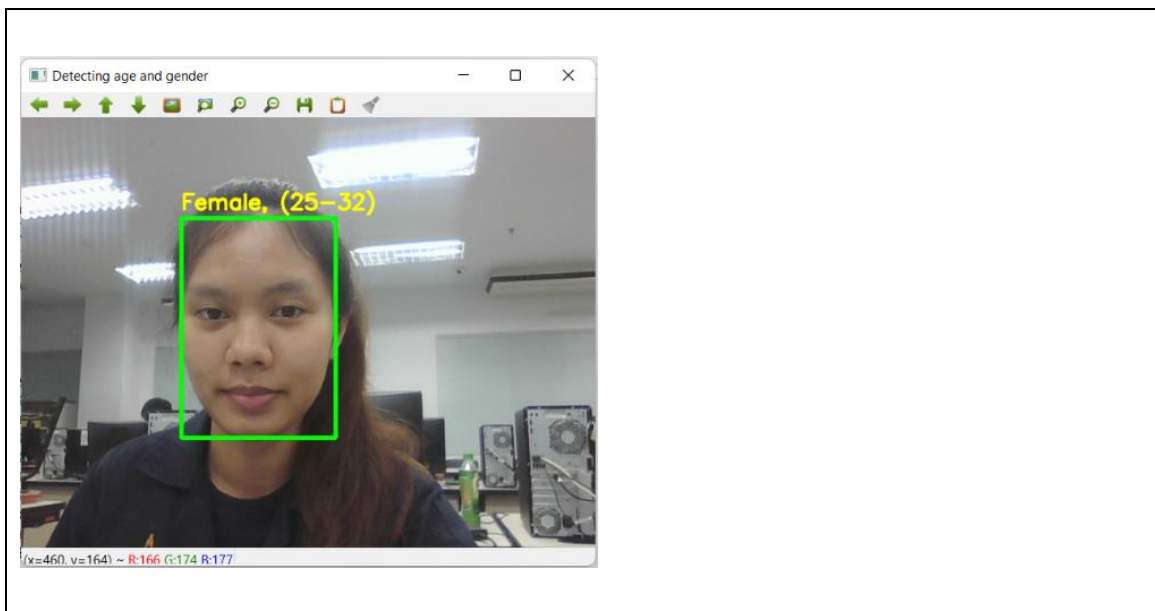
2. Test2: Multiple File ทดสอบกับภาพนิ่งของบุคคลที่เราสนใจจำนวนไม่ต่ำกว่า 10 ภาพ







3. Test3: Video Camera ทดสอบกับวิดีโอของตัวเอง



4. Test4: Video File ทดสอบกับวิดีโอของบุคคลที่เราสนใจ



5. อภิปรายข้อจำกัดของวิธีนี้ และอภิปรายแนวทางการนำไปใช้

- หากวิดีโอหรือรูปที่นำมาใช้ทดสอบ เห็นใบหน้าไม่ชัด เช่นใส่แมส อาจไม่สามารถระบุเพศ และอายุได้ถูกต้อง

Wk0913.20221005,08 -- Picture to Video

1. เลือกรูปภาพตามโปรเจ็คของตัวเอง
 - 1.1 ให้ครบชิ้นงานทั้ง 4 กลุ่มที่คัดแยก จะได้ 4 รูป
 - 1.2 และให้มีอื่นๆที่ไม่อยู่ใน 4 กลุ่ม อีกจำนวน 4 รูป
 - 1.3 จาก 1.1+1.2 จะได้จำนวน 8 รูป
 - 1.4 หมุนรูป 1.3 ที่มุม -30 องศา จะได้จำนวน +8 รูป
 - 1.5 หมุนรูป 1.3 ที่มุม -15 องศา จะได้จำนวน +8 รูป
 - 1.6 หมุนรูป 1.3 ที่มุม +15 องศา จะได้จำนวน +8 รูป
 - 1.7 หมุนรูป 1.3 ที่มุม +30 องศา จะได้จำนวน +8 รูป
 - 1.8 จะมีจำนวนทั้งหมด $8+8+8+8 = 40$ รูป
 - 1.9 ทำสำเนาซ้ำ 1.8 จะได้ทั้งหมด = 80 รูป
2. ใช้ทั้ง 80 รูป ทำ **Video_B1** แบบสไลด์ขึ้น หรือสไลด์ขวาง ตามที่สนใจ
3. ใช้ Video_B1 นับจำนวนรวม บันทึกเป็น **Video_B1_All_Count**
4. บันทึกผลในตาราง

ลำดับ	รายการ	จำนวนชิ้นงาน จริง	ผลการนับด้วย โปรแกรม	ผลต่าง	ถูกต้อง(%)
1	Video_A1	77	77	0	100%
2	Video_B1	80	80	0	100%

ข้อจำกัด ปัญหา ข้อเสนอแนะ ของ Video A

Video ไม่คมชัด การตกกระทบของแสงยังไม่ดีพอ ทำให้เกิดเงา อาจส่งผลถึงความแม่นยำของการนับได้

ข้อจำกัด ปัญหา ข้อเสนอแนะ ของ Video B

ข้อจำกัดของ Video_B มีน้อยมาก ยังไม่พบปัญหา เนื่องจาก Video มีความคมชัด ทำให้สามารถนับผลรวมของส้มได้แม่นยำ ไม่เกิดข้อผิดพลาด

5. Upload **Video_B1** ขึ้น YouTube แล้วแปะลิงค์ที่รายงานและ กลุ่มโปรเจ็ค

<https://youtu.be/ukcOwXdIETI>

6. Upload **Video_B1_All_Count** ขึ้น YouTube แล้วแปะลิงค์ที่รายงานและ กลุ่มโปรเจ็ค

<https://youtu.be/WsA9Fia3MdQ>

7. Capture โปรแกรมการทำงานและผลการทำงาน

```

In [2]: 1 import cv2
2 cap = cv2.VideoCapture("D:\Machine\Week9\video_B_new7.mp4")
3
4 fourcc = cv2.VideoWriter_fourcc(*'MP4V')
5 frame_width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
6 frame_height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))
7 out = cv2.VideoWriter('./im_m1/B1_All_Count.avi', fourcc, 30, (frame_width, frame_height))
8 #out = cv2.VideoWriter('./im_m1/B1_All_Count.avi', fourcc, 30, (350, 450)) ##
9
10 object_detector = cv2.createBackgroundSubtractorMOG2()
11
12 ret, frame = cap.read()
13 height, width = frame.shape[:2]
14 referenceLine = int(0.60*height) # 90%
15 ImageWidth = width
16 #print(height,width)
17
18 positionText = (30,referenceLine-30)
19 nCar = 0
20 BoxInline_y = 1
21 BoxInline_x = 1
22
23 while(cap.isOpened()):
24     ret, frame = cap.read()
25     if frame is None:
26         break
27
28     BoxInline_y = BoxInline_x
29     BoxInline_x = 0
30
31     mask = object_detector.apply(frame)
32     contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
33     for cnt in contours:
34         # Calculate area and remove small elements
35         area = cv2.contourArea(cnt)
36         if area > 1000:
37             cv2.drawContours(frame, [cnt], -1, (0, 255, 0), 2)
38             x, y, w, h = cv2.boundingRect(cnt)
39             cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 0, 255), 3)
40
41             if y < referenceLine and (y+h) > referenceLine:
42                 BoxInline_x = 1
43
44     if BoxInline_y==0 and BoxInline_x==1:
45         nCar = nCar + 1
46
47     textShow = str(nCar)+'-'+str(BoxInline_x)+'-'+str(BoxInline_y)
48     cv2.line(frame,(0,referenceLine),(ImageWidth,referenceLine),(255, 0, 0), 3)
49     cv2.putText(frame,textShow , positionText, cv2.FONT_HERSHEY_PLAIN, 4, (255, 0, 0), 2)
50
51     cv2.imshow("Mask", mask)
52     cv2.imshow("Frame", frame)
53     out.write(frame)
54     if cv2.waitKey(1) & 0xFF == 27: # ESC Key
55         break
56
57 cap.release()
58 cv2.destroyAllWindows()

```



8. คัดลอกโปรแกรมการทำงาน

```

import cv2
cap = cv2.VideoCapture("D:\\Machine\\Week9\\video_B_new7.mp4")

fourcc = cv2.VideoWriter_fourcc(*"MP4V")
frame_width = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
frame_height = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))
out = cv2.VideoWriter('./im_m1/B1_All_Count.avi', fourcc, 30, (frame_width, frame_height))
#out = cv2.VideoWriter('./im_m1/B1_All_Count.avi', fourcc,30, (350,450)) ##

object_detector = cv2.createBackgroundSubtractorMOG2()

ret, frame = cap.read()
height, width = frame.shape[:2]
referenceLine = int(0.60*height) # 90%
ImageWidth = width
#print(height,width)

positionText = (30,referenceLine-30)
nCar = 0
BoxInline_y = 1
BoxInline_x = 1

while(cap.isOpened()):
    ret, frame = cap.read()
    if frame is None:
        break

    BoxInline_y = BoxInline_x
    BoxInline_x = 0

    mask = object_detector.apply(frame)
    contours, _ = cv2.findContours(mask, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
    for cnt in contours:

```

```

# Calculate area and remove small elements
area = cv2.contourArea(cnt)
if area > 1000:
    cv2.drawContours(frame, [cnt], -1, (0, 255, 0), 2)
    x, y, w, h = cv2.boundingRect(cnt)
    cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 0, 255), 3)

    if y < referenceLine and (y+h) > referenceLine:
        BoxInline_x = 1

if BoxInline_y==0 and BoxInline_x==1:
    nCar = nCar + 1

textShow = str(nCar)+'-'+str(BoxInline_x)+'-'+str(BoxInline_y)
cv2.line(frame,(0,referenceLine),(ImageWidth,referenceLine),(255, 0, 0), 3)
cv2.putText(frame,textShow , positionText, cv2.FONT_HERSHEY_PLAIN, 4, (255, 0, 0), 2)

cv2.imshow("Mask", mask)
cv2.imshow("Frame", frame)
out.write(frame)
if cv2.waitKey(1) & 0xFF == 27: # ESC Key
    break
cap.release()
cv2.destroyAllWindows()

```