

## รายงาน

### เรื่อง ระบบตรวจจับและคัดแยกผลส้ม

#### โดย

นายณัฐพล จงเกษกรรม รหัสนักศึกษา B6304416

นายธีรวัฒน์ กุดกึ่ง รหัสนักศึกษา B6321031

นางสาวขวัญจิรา พันธเกตุ รหัสนักศึกษา B6321451

#### เสนอ

อาจารย์ ดร.วิชัย ศรีสุรักษ์

รายงานฉบับนี้เป็นส่วนหนึ่งของรายวิชา 558404 MACHINE VISION

สำนักวิชาวิศวกรรมศาสตร์

ภาคการศึกษาที่ 1 ปีการศึกษา 2565

มหาวิทยาลัยเทคโนโลยีสุรนารี

## คำนำ

รายงานฉบับนี้เป็นส่วนหนึ่งของวิชา Machine Vision (558404) โดยมีจุดประสงค์ เพื่อการศึกษา ความรู้ที่ได้จากเรื่อง Machine Vision ทั้งนี้ ในรายงานนี้มีเนื้อหาประกอบด้วยความรู้เกี่ยวกับ การตรวจจับ วัตถุ (Object Detection) การติดตาม (Tracking) ตลอดจนการประยุกต์ใช้ในการแยกแยะแต่ละชนิด ว่ามี ขนาดและรูปร่างแตกต่างกัน หรือมีสีต่างกันแบบไหน

คณะผู้จัดทำได้เลือกหัวข้อนี้ในการทำรายงาน เนื่องจากเป็นเรื่องที่น่าสนใจ รวมทั้งแสดงให้เห็นถึง ความสามารถของเครื่องจักรในการตรวจจับวัตถุตามที่เขียนโปรแกรมไว้ คณะผู้จัดทำต้องขอขอบคุณ อาจารย์ ดร. วิชัย ศรีสุรักษ์ ผู้ให้ความรู้ และแนวทางการศึกษา หวังว่ารายงานฉบับนี้จะให้ความรู้ และเป็นประโยชน์แก่ ผู้อ่านทุก ๆ ท่าน หากมีข้อเสนอแนะประการใด คณะผู้จัดทำขอรับไว้ด้วยความขอบพระคุณยิ่ง

คณะผู้จัดทำ

3 พ.ย. 2565

# สารบัญ

เรื่อง	หน้า
ทฤษฎี .....	1
OPENCV.....	1
ระบบสี HSV .....	1
การทำ CONTOUR.....	2
การทำงาน.....	3
รูปโปรแกรม JUPYTER NOTEBOOK .....	3
นับจำนวนรวม.....	3
การแยกสีจากภาพ.....	4
การแยกสีจาก video.....	5
การแยกขนาด.....	6
การแยกสีและขนาด.....	8
CODE PYTHON3 .....	10
นับจำนวนรวม.....	10
การแยกสีจากภาพ.....	11
การแยกสีจาก video.....	12
การแยกขนาด.....	13
การแยกสีและขนาด.....	15
ผลการทำงาน.....	18
สรุป.....	18
ข้อเสนอแนะ.....	19
อ้างอิง .....	20

## ทฤษฎี

โครงการนี้ได้คณะผู้จัดทำได้ศึกษาแนวคิดและทฤษฎีที่เกี่ยวข้อง เพื่อนำมา ประยุกต์ในการทำโครงการ ซึ่งประกอบด้วยเนื้อหาดังต่อไปนี้

### 1. OpenCV

OpenCV หรือ Open-Source Computer Vision Library ใช้ในการประมวลผลภาพและงานทางด้านการมองเห็นของคอมพิวเตอร์ (Computer Vision) Library นี้ถูกพัฒนาขึ้นด้วย ภาษา C และ C++ และยังมี interface ที่ไว้ เชื่อมต่อกับ tool อื่นด้วยเช่น Python, Ruby, Matlab เป็นต้น นอกจากนี้ OpenCV เป็น library ที่สร้างขึ้นเพื่อให้ผู้ใช้หรือนักพัฒนาสามารถใช้ฟังก์ชันใน library มาพัฒนาชิ้นงานที่มีความซับซ้อนโดยใช้เวลาเพียงไม่นาน OpenCV ประกอบด้วย Data Structure และ Algorithm

- Data Structure ใช้เก็บข้อมูลต่าง ๆ อาทิ เช่น รูปภาพ เมทริกซ์ พิกัด
- Algorithm เพื่อการประมวลผลต่าง โดยเฉพาะการประมวลผลทางรูปภาพ

ข้อจำกัดของ OpenCV คือ สามารถใช้ งานได้ เฉพาะเครื่องคอมพิวเตอร์ที่ใช้หน่วยประมวลผล (CPU) จาก Intel Corporation แต่ข้อจำกัดนี้ทำให้เกิดจุดเด่นเช่นกัน กล่าวคือ การประมวลผลต่าง ๆ จะใช้ความสามารถของหน่วยประมวลผลอย่างเต็มที่ทำให้โปรแกรมที่พัฒนาโดยใช้ OpenCV นี้มีประสิทธิภาพในการประมวลผลที่สูงมาก

### 2. ระบบสี HSV

ระบบสี HSV (Hue Saturation Value) เป็นการพิจารณาสีโดยใช้ Hue Saturation และ Value ซึ่ง Hue คือ ค่าสีของสีหลัก(แดง เขียวและน้ำเงิน)ในทางปฏิบัติจะอยู่ระหว่าง 0 และ 255 ซึ่งถ้า Hue มีค่าเท่ากับ 0 จะแทนสีแดงและเมื่อ Hue มีค่าเพิ่มขึ้นเรื่อย ๆ สีก็จะเปลี่ยนแปลง ไปตามสเปกตรัมของสีจนถึง 256 จึงจะกลับมาเป็นสีแดงอีกครั้งซึ่งสามารถแทนให้อยู่ในรูปของ องศาได้ดังนี้คือ สีแดง = 0 องศา สีเขียวเท่ากับ 120 องศา สีน้ำเงินเท่ากับ 240 องศา Hue สามารถ คำนวณได้จากระบบสี RGB ได้ดังนี้

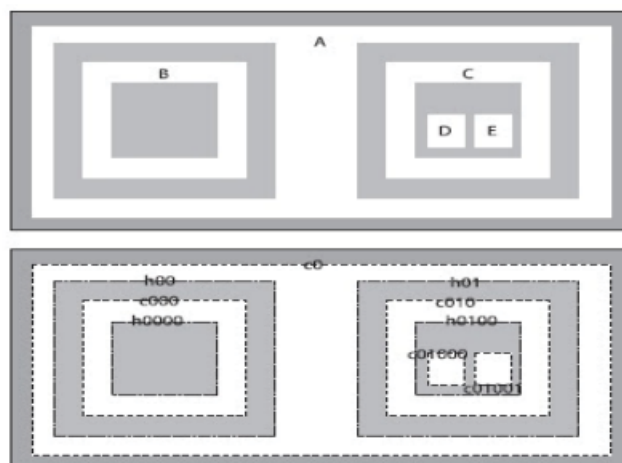
$$\text{red h} = \text{red} - \min(\text{red}, \text{green}, \text{blue})$$

$$\text{green h} = \text{green} - \min(\text{red}, \text{green}, \text{blue})$$

$$\text{blue h} = \text{blue} - \min(\text{red}, \text{green}, \text{blue})$$

### 3. การหา Contour

นิยามของ Contour คือลิสต์ของจุดที่แทน Curve ในรูปภาพ กรณีของ Contour ใน OpenCV จะเก็บไว้ในรูปของ Sequence ตามรูป



ภาพบนเป็นภาพที่ต้องการหา Contour ซึ่งประกอบด้วยส่วน A-E ส่วนด้านล่างคือ Contour ที่ OpenCV หาได้ (โดยฟังก์ชัน `cvFindContours`) ซึ่งกำกับไว้ด้วยคำว่า cX หรือ hX โดย c หมายถึง Contour, h หมายถึง Hole ตัวที่เป็นเส้นประ เป็น Exterior Boundaries ของพื้นที่สีขาว (ที่ไม่ใช่ 0 คือ กรอบของพื้นที่ขาว) ตัวเส้นจุดนั้นเป็น Interior Boundaries ของพื้นที่สีขาว หรือ Exterior Boundary พื้นที่ดำ (อยู่ในพื้นที่ขาว หรือเป็นกรอบของพื้นที่สีดำ) ซึ่ง OpenCV จะแยก ความแตกต่างระหว่างสองตัวนั้นโดย ฟังก์ชันที่ใช้ในการหา Contour ของ OpenCV ใน MATLAB คือ `cv.findContours` โดยเป็นการหา Contours ในภาพไบนารีโดยมีรูปแบบคำสั่ง ดังนี้

```
contours = cv.findContours(image)
```

```
contours = cv.findContours(image, 'OptionName', optionValue, ...)
```

โดยมีอินพุตเป็นภาพไบนารีและเอาต์พุตเป็น Contour ที่ตรวจพบ โดยแต่ละ Contour จะ ถูกเก็บ เป็นเวกเตอร์ของจุด, Option (Mode : External) รีเทิร์น Contour ภายนอกเพียงอย่างเดียวโดย ทุก Contour มีลำดับชั้นเป็น `hierarchy[i][2] = hierarchy[i][3]=-1`

## การทำงาน

รูปโปรแกรม Jupyter Notebook

### 1. นับจำนวนรวม

```
In [ ]: 1 #นับจำนวนรวม
2 #video_c
3 import cv2
4
5 cap = cv2.VideoCapture("./video/video_C2.avi")
6 fourcc = cv2.VideoWriter_fourcc(*"XVID")
7 out = cv2.VideoWriter("./video/video_CALL.avi", fourcc, 30, (540,960))
8
9 object_detector = cv2.createBackgroundSubtractorMOG2(history=100, varThreshold=40)
10 ret, frame = cap.read()
11 height, width = frame.shape[:2]
12 referenceLine = int(0.30*height)
13 positionText = (10,referenceLine-30)
14
15 ImageWidth = width
16 All = 0
17
18 BoxInline_y = 1
19 BoxInline_x = 1
20
21 while(cap.isOpened()):
22     ret, frame = cap.read()
23     height, width, _ = frame.shape
24
25     BoxInline_y = BoxInline_x
26     BoxInline_x = 0
27
28     roi = frame[0: 960,50: 490]
29     mask = object_detector.apply(roi)
30     _, mask = cv2.threshold(mask, 254, 255, cv2.THRESH_BINARY)
31     contours, _ = cv2.findContours(mask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
32     for cnt in contours:
33         area = cv2.contourArea(cnt)
34         if area > 9000:
35             x, y, w, h = cv2.boundingRect(cnt)
36             cv2.rectangle(roi, (x, y), (x + w, y + h), (0, 255, 0), 3)
37             if y < referenceLine and (y+h) > referenceLine:
38                 BoxInline_x = 1
39             if BoxInline_y==0 and BoxInline_x==1:
40                 All = All + 1
41
42     textShow = str(All)+'-'+str(BoxInline_x)+'-'+str(BoxInline_y)
43     textShow = 'All = ' + str(All)
44
45     cv2.line(frame,(0,referenceLine),(ImageWidth,referenceLine),(0, 0, 255), 3)
46     cv2.putText(frame,textShow , positionText, cv2.FONT_HERSHEY_PLAIN, 2, (0, 0, 255), 2)
47
48     cv2.imshow("M.Frame", frame)
49     cv2.imshow("Contours", mask)
50     key = cv2.waitKey(30)
51     out.write(frame)
52     if cv2.waitKey(1) & 0xFF == 27: # ESC Key
53         break
54
55 cap.release()
56 cv2.destroyAllWindows()
```

## 2. การแยกสีจากภาพ

```

In [ ]: 1 #test แยกสี
        2 import cv2
        3
        4 cap = cv2.imread("./images/o/O07.png")
        5 cap = cv2.imread("./images/o/O17.PNG")
        6 cap = cv2.imread("./images/o/O27.PNG")
        7 #cap = cv2.imread("./images/o/O37.PNG")
        8
        9
        10 hsv_frame = cv2.cvtColor(cap, cv2.COLOR_BGR2HSV)
        11 height, width, _ = cap.shape
        12
        13 cx = int(width / 2)
        14 cy = int(height / 4)
        15
        16 # Pick pixel value
        17 pixel_center = hsv_frame[cy, cx]
        18
        19 hue_value = pixel_center[0]
        20 if hue_value < 28:
        21     color = "orange"
        22 elif hue_value < 50:
        23     color = "green"
        24 else:
        25     color = "Undefined"
        26
        27 print(pixel_center)
        28 #cv2.putText(cap, color, (10,50), 0,1, (255,0,0),2)
        29
        30 cv2.imshow("Frame", cap)
        31 cv2.waitKey(0)
        32 cv2.destroyAllWindows()
        33

```

## 3. การแยกสีจาก video

```
In [ ]: 1 #การแยกสีจาก video
2 import cv2
3
4 cap = cv2.VideoCapture("./video/video_C2.avi")
5
6 while True:
7     _, frame = cap.read()
8     hsv_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
9     height, width, _ = frame.shape
10
11     cx = int(width / 2)
12     cy = int(height / 4)
13
14     # Pick pixel value
15     pixel_center = hsv_frame[cy, cx]
16
17     hue_value = pixel_center[0]
18     if hue_value < 28:
19         color = "orange"
20     elif hue_value < 50:
21         color = "green"
22     else:
23         color = "Undefined"
24
25     print(pixel_center)
26     cv2.putText(frame, color, (10,50), 0,1, (255,0,0),2)
27     cv2.circle (frame, (cx, cy), 10, (255, 0, 0), 1)
28
29     cv2.imshow ("Frame", frame)
30     key = cv2.waitKey(15)
31     if cv2.waitKey(1) & 0xFF == 27: # ESC Key
32         break
33 cap.release
34 cv2.destroyAllWindows
35
```



## 4. การแยกขนาด

```

In [ ]: 1  #แยกขนาด
        2  import cv2
        3
        4  cap = cv2.VideoCapture("./video/video_C2.avi")
        5
        6  fourcc = cv2.VideoWriter_fourcc(*'XVID')
        7  out = cv2.VideoWriter('./video/video_C3.avi', fourcc, 30, (540,960))
        8
        9
        10 object_detector = cv2.createBackgroundSubtractorMOG2(history=100, varThreshold=40)
        11 ret, frame = cap.read()
        12 height, width = frame.shape[:2]
        13 referenceLine = int(0.30*height) # 90%
        14 positionText = (10,referenceLine-30)
        15 positionOl = (10,referenceLine-150)
        16 positionOm = (10,referenceLine-120)
        17 positionGm = (10,referenceLine-90)
        18 positionGs = (10,referenceLine-60)
        19 ImageWidth = width
        20 All = 0
        21 Orange_l = 0 #ส้มใหญ่
        22 Orange_m = 0 #ส้มเล็ก
        23 Green_m = 0 #เขียวใหญ่
        24 Green_s = 0 #เขียวเล็ก
        25 BoxInline_y = 1
        26 BoxInline_x = 1
        27 BoxInline_ol = 1
        28 BoxInline_om = 1
        29 BoxInline_gm = 1
        30 BoxInline_gs = 1
        31
        32 while(cap.isOpened()):
        33     ret, frame = cap.read()
        34     height, width, _ = frame.shape
        35     BoxInline_ol = BoxInline_x
        36     BoxInline_om = BoxInline_x
        37     BoxInline_gm = BoxInline_x
        38     BoxInline_gs = BoxInline_x
        39     BoxInline_y = BoxInline_x
        40     BoxInline_x = 0
        41
        42     roi = frame[0: 960,50: 490]
        43     mask = object_detector.apply(roi)
        44     _, mask = cv2.threshold(mask, 254, 255, cv2.THRESH_BINARY)
        45     contours, _ = cv2.findContours(mask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
        46

```

```

46
47 for cnt in contours:
48     area = cv2.contourArea(cnt)
49     if area > 9000:
50         x, y, w, h = cv2.boundingRect(cnt)
51         cv2.rectangle(roi, (x, y), (x + w, y + h), (0, 255, 0), 3)
52         if y < referenceLine and (y+h) > referenceLine:
53             BoxInline_x = 1
54             print(area)
55             if area > 42000 and area < 56000:
56                 BoxInline_ol = 1
57             elif area > 18000 and area < 26000:
58                 BoxInline_om = 1
59             elif area > 18000 and area < 26000:
60                 BoxInline_gm = 1
61             elif area > 9000 and area < 15000:
62                 BoxInline_gs = 1
63         if BoxInline_y==0 and BoxInline_x==1:
64             All = All + 1
65             if BoxInline_ol == 1:
66                 Orange_l += 1
67             elif BoxInline_om == 1:
68                 Orange_m += 1
69             elif BoxInline_gm == 1:
70                 Green_m += 1
71             elif BoxInline_gs == 1:
72                 Green_s += 1
73
74     #textShow = str(All)+'='+str(BoxInline_x)+''+str(BoxInline_y)
75     textShow = 'All = ' + str(All)
76     textA = 'O_l = ' + str(Orange_l)
77     textB = 'O_m = ' + str(Orange_m)
78     textC = 'G_m = ' + str(Green_m)
79     textD = 'G_s = ' + str(Green_s)
80     cv2.line(frame,(0,referenceLine),(ImageWidth,referenceLine),(0, 0, 255), 3)
81     cv2.putText(frame,textShow , positionText, cv2.FONT_HERSHEY_PLAIN, 2, (0, 0, 255), 2)
82     cv2.putText(frame,textA , positionOl, cv2.FONT_HERSHEY_PLAIN, 2, (255, 0, 0), 2)
83     cv2.putText(frame,textB , positionOm, cv2.FONT_HERSHEY_PLAIN, 2, (0, 255, 255), 2)
84     cv2.putText(frame,textC , positionGm, cv2.FONT_HERSHEY_PLAIN, 2, (255, 255, 0), 2)
85     cv2.putText(frame,textD , positionGs, cv2.FONT_HERSHEY_PLAIN, 2, (255, 0, 255), 2)
86     cv2.imshow("M.Frame", frame)
87     cv2.imshow("Contours", mask)
88     key = cv2.waitKey(30)
89     #out.write(frame)
90     if cv2.waitKey(1) & 0xFF == 27: # ESC Key
91         break
92
93 cap.release()
94 cv2.destroyAllWindows()

```

## 5. การแยกสีและขนาด

```

In [ ]: 1 #แยกสี แยกขนาด
        2 import cv2
        3
        4 cap = cv2.VideoCapture("./video/video_C2.avi")
        5 fourcc = cv2.VideoWriter_fourcc(*"XVID")
        6 out = cv2.VideoWriter("./video_count/video_C8.avi", fourcc, 30, (540,960))
        7
        8 object_detector = cv2.createBackgroundSubtractorMOG2(history=100, varThreshold=40)
        9 ret, frame = cap.read()
        10 height, width = frame.shape[:2]
        11
        12 referenceLine = int(0.30*height) # 90%
        13 positionText = (10,referenceLine-30)
        14 positionOl = (10,referenceLine-150)
        15 positionOm = (10,referenceLine-120)
        16 positionGm = (10,referenceLine-90)
        17 positionGs = (10,referenceLine-60)
        18 ImageWidth = width
        19
        20 All = 0
        21 Orange_l = 0 #ส้มใหญ่
        22 Orange_m = 0 #ส้มเล็ก
        23 Green_m = 0 #เขียวใหญ่
        24 Green_s = 0 #เขียวเล็ก
        25 tarea = 0
        26
        27 BoxInline_y = 1
        28 BoxInline_x = 1
        29 BoxInline_ol = 1
        30 BoxInline_om = 1
        31 BoxInline_gm = 1
        32 BoxInline_gs = 1
        33
        34 while(cap.isOpened()):
        35     ret, frame = cap.read()
        36     height, width, _ = frame.shape
        37     hsv_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
        38     cx = int(width / 2)
        39     cy = referenceLine + 45
        40
        41     # Pick pixel value
        42     pixel_center = hsv_frame[cy, cx]
        43     bgr = frame[cy, cx]
        44     cb, cg, cr = int(bgr[0]), int(bgr[1]), int(bgr[2])
        45
        46     hue_value = pixel_center[0]
        47
        48     if hue_value < 28:
        49         color = "orange"
        50     elif hue_value < 50:
        51         color = "green"
        52     else:
        53         color = "blue"
        54
        55     #print(pixel_center)
        56
        57
        58     BoxInline_ol = BoxInline_x
        59     BoxInline_om = BoxInline_x
        60     BoxInline_gm = BoxInline_x
        61     BoxInline_gs = BoxInline_x
        62     BoxInline_y = BoxInline_x
        63     BoxInline_x = 0
        64
        65     roi = frame[0: 960,50: 490]
        66     mask = object_detector.apply(roi)
        67     _, mask = cv2.threshold(mask, 254, 255, cv2.THRESH_BINARY)
        68     contours, _ = cv2.findContours(mask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
        69

```

```

70 for cnt in contours:
71     area = cv2.contourArea(cnt)
72     if area > 9000:
73         x, y, w, h = cv2.boundingRect(cnt)
74         cv2.rectangle(roi, (x, y), (x + w, y + h), (0, 255, 0), 3)
75         if y < referenceLine and (y+h) > referenceLine:
76             BoxInline_x = 1
77             tarea = area
78             #print(area)
79             if area > 9000 and area < 15000:
80                 BoxInline_gs = 1
81             elif area > 18000 and area < 26000:
82                 if color == "orange":
83                     BoxInline_om = 1
84                 else:
85                     BoxInline_gm = 1
86             else :
87                 BoxInline_ol = 1
88         if BoxInline_y==0 and BoxInline_x==1:
89             All = All + 1
90             if BoxInline_ol == 1:
91                 Orange_l += 1
92             elif BoxInline_om == 1:
93                 Orange_m += 1
94             elif BoxInline_gm == 1:
95                 Green_m += 1
96             elif BoxInline_gs == 1:
97                 Green_s += 1
98
99         error = All - (Orange_l+Orange_m+Green_m+Green_s)
100
101         #textShow = str(All)+' '+str(BoxInline_x)+' '+str(BoxInline_y)
102         textShow = 'All = ' + str(All)
103         textA = 'O_L = ' + str(Orange_l)
104         textB = 'O_M = ' + str(Orange_m)
105         textC = 'G_M = ' + str(Green_m)
106         textD = 'G_S = ' + str(Green_s)
107         textE = 'error = ' + str(error)
108         textF = 'area = ' + str(tarea)
109         cv2.line(frame,(0,referenceLine),(ImageWidth,referenceLine),(0, 0, 255), 3)
110         cv2.putText(frame,textShow , positionText, cv2.FONT_HERSHEY_PLAIN, 2, (0, 0, 255), 2)
111         cv2.putText(frame,textA , positionOl, cv2.FONT_HERSHEY_PLAIN, 2, (0, 128, 255), 2)
112         cv2.putText(frame,textB , positionOm, cv2.FONT_HERSHEY_PLAIN, 2, (0, 255, 255), 2)
113         cv2.putText(frame,textC , positionGm, cv2.FONT_HERSHEY_PLAIN, 2, (0, 153, 76), 2)
114         cv2.putText(frame,textD , positionGs, cv2.FONT_HERSHEY_PLAIN, 2, (0, 102, 0), 2)
115         cv2.putText(frame,textE , (10,700), cv2.FONT_HERSHEY_PLAIN, 2, (0, 0, 255), 2)
116         cv2.putText(frame,textF , (10,650), cv2.FONT_HERSHEY_PLAIN, 2, (0, 0, 255), 2)
117
118         cv2.putText(frame, color, (10,50), 0,1, (cb, cg, cr),2)
119
120         cv2.imshow("M.Frame", frame)
121         #cv2.imshow("roi", roi)
122         key = cv2.waitKey(15)
123         out.write(frame)
124         if cv2.waitKey(1) & 0xFF == 27: # ESC Key
125             break
126
127 cap.release()
128 cv2.destroyAllWindows()

```

## Code Python3

## 1. นับจำนวนรวม

```

#นับจำนวนรวม
#video_c
import cv2

cap = cv2.VideoCapture("./video/video_C2.avi")
fourcc = cv2.VideoWriter_fourcc(*'XVID')
out = cv2.VideoWriter('./video/video_CAll.avi', fourcc, 30, (540,960))

object_detector = cv2.createBackgroundSubtractorMOG2(history=100,
varThreshold=40)
ret, frame = cap.read()
height, width = frame.shape[:2]
referenceLine = int(0.30*height)
positionText = (10,referenceLine-30)

ImageWidth = width
All = 0

BoxInline_y = 1
BoxInline_x = 1

while(cap.isOpened()):
    ret, frame = cap.read()
    height, width, _ = frame.shape

    BoxInline_y = BoxInline_x
    BoxInline_x = 0

    roi = frame[0: 960,50: 490]
    mask = object_detector.apply(roi)
    _, mask = cv2.threshold(mask, 254, 255, cv2.THRESH_BINARY)
    contours, _ = cv2.findContours(mask, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)
    for cnt in contours:
        area = cv2.contourArea(cnt)
        if area > 9000:
            x, y, w, h = cv2.boundingRect(cnt)
            cv2.rectangle(roi, (x, y), (x + w, y + h), (0, 255, 0), 3)
            if y < referenceLine and (y+h) > referenceLine:
                BoxInline_x = 1
    if BoxInline_y==0 and BoxInline_x==1:
        All = All + 1

    textShow = str(All)+'='+str(BoxInline_x)+' '+str(BoxInline_y)
    textShow = 'All = ' + str(All)

    cv2.line(frame,(0,referenceLine),(ImageWidth,referenceLine),(0, 0,
255), 3)

```

```

    cv2.putText(frame, textShow, positionText, cv2.FONT_HERSHEY_PLAIN,
2, (0, 0, 255), 2)

    cv2.imshow("M.Frame", frame)
    cv2.imshow("Contours", mask)
    key = cv2.waitKey(30)
    out.write(frame)
    if cv2.waitKey(1) & 0xFF == 27: # ESC Key
        break

cap.release()
cv2.destroyAllWindows()

```

## 2. การแยกสีจากภาพ

```

#test การแยกสี
import cv2

cap = cv2.imread("./images/o/007.png")
cap = cv2.imread("./images/o/017.PNG")
cap = cv2.imread("./images/o/027.PNG")
#cap = cv2.imread("./images/o/037.PNG")

hsv_frame = cv2.cvtColor(cap, cv2.COLOR_BGR2HSV)
height, width, _ = cap.shape

cx = int(width / 2)
cy = int(height / 4)

pixel_center = hsv_frame[cy, cx]
hue_value = pixel_center[0]
if hue_value < 28:
    color = "orange"
elif hue_value < 50:
    color = "green"
else:
    color = "Undefined"
print(pixel_center)
cv2.putText(cap, color, (10,50), 0,1, (255,0,0),2)

cv2.imshow("Frame", cap)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

### 3. การแยกสีจาก video

```
#การแยกสีจาก video
import cv2

cap = cv2.VideoCapture("./video/video_C2.avi")

while True:
    _, frame = cap.read()
    hsv_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
    height, width, _ = frame.shape

    cx = int(width / 2)
    cy = int(height / 4)

    # Pick pixel value
    pixel_center = hsv_frame[cy, cx]

    hue_value = pixel_center[0]
    if hue_value < 28:
        color = "orange"
    elif hue_value < 50:
        color = "green"
    else:
        color = "Undefined"

    print(pixel_center)
    cv2.putText(frame, color, (10,50), 0,1, (255,0,0),2)
    cv2.circle (frame, (cx, cy), 10, (255, 0, 0), 1)

    cv2.imshow ("Frame", frame)
    key = cv2.waitKey(15)
    if cv2.waitKey(1) & 0xFF == 27: # ESC Key
        break
cap.release
cv2.destroyAllWindows
```

## 4. การแยกขนาด

```

#แยกขนาด
import cv2

cap = cv2.VideoCapture("./video/video_C2.avi")

fourcc = cv2.VideoWriter_fourcc(*'XVID')
out = cv2.VideoWriter('./video/video_C3.avi', fourcc, 30, (540,960))

object_detector = cv2.createBackgroundSubtractorMOG2(history=100,
varThreshold=40)
ret, frame = cap.read()
height, width = frame.shape[:2]
referenceLine = int(0.30*height) # 90%
positionText = (10,referenceLine-30)
positionOl = (10,referenceLine-150)
positionOm = (10,referenceLine-120)
positionGm = (10,referenceLine-90)
positionGs = (10,referenceLine-60)
ImageWidth = width
All = 0
Orange_l = 0 #ส้มใหญ่
Orange_m = 0 #ส้มเล็ก
Green_m = 0 #เขียวใหญ่
Green_s = 0 #เขียวเล็ก
BoxInline_y = 1
BoxInline_x = 1
BoxInline_ol = 1
BoxInline_om = 1
BoxInline_gm = 1
BoxInline_gs= 1

while(cap.isOpened()):
    ret, frame = cap.read()
    height, width, _ = frame.shape
    BoxInline_ol = BoxInline_x
    BoxInline_om = BoxInline_x
    BoxInline_gm = BoxInline_x
    BoxInline_gs = BoxInline_x
    BoxInline_y = BoxInline_x
    BoxInline_x = 0

    roi = frame[0: 960,50: 490]
    mask = object_detector.apply(roi)
    _, mask = cv2.threshold(mask, 254, 255, cv2.THRESH_BINARY)
    contours, _ = cv2.findContours(mask, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)

    for cnt in contours:
        area = cv2.contourArea(cnt)

```



```

    if area > 9000:
        x, y, w, h = cv2.boundingRect(cnt)
        cv2.rectangle(roi, (x, y), (x + w, y + h), (0, 255, 0), 3)
        if y < referenceLine and (y+h) > referenceLine:
            BoxInline_x = 1
            print(area)
            if area > 42000 and area < 56000:
                BoxInline_ol = 1
            elif area > 18000 and area < 26000:
                BoxInline_om = 1
            elif area > 18000 and area < 26000:
                BoxInline_gm = 1
            elif area > 9000 and area < 15000:
                BoxInline_gs = 1
    if BoxInline_y==0 and BoxInline_x==1:
        All = All + 1
        if BoxInline_ol == 1:
            Orange_l += 1
        elif BoxInline_om == 1:
            Orange_m += 1
        elif BoxInline_gm == 1:
            Green_m += 1
        elif BoxInline_gs == 1:
            Green_s += 1

    #textShow = str(All)+'='+str(BoxInline_x)+','+str(BoxInline_y)
    textShow = 'All = ' + str(All)
    textA = 'O_l = ' + str(Orange_l)
    textB = 'O_m = ' + str(Orange_m)
    textC = 'G_m = ' + str(Green_m)
    textD = 'G_s = ' + str(Green_s)
    cv2.line(frame,(0,referenceLine),(ImageWidth,referenceLine),(0, 0,
255), 3)
    cv2.putText(frame,textShow , positionText, cv2.FONT_HERSHEY_PLAIN,
2, (0, 0, 255), 2)
    cv2.putText(frame,textA , positionOl, cv2.FONT_HERSHEY_PLAIN, 2,
(255, 0, 0), 2)
    cv2.putText(frame,textB , positionOm, cv2.FONT_HERSHEY_PLAIN, 2, (0,
255, 255), 2)
    cv2.putText(frame,textC , positionGm, cv2.FONT_HERSHEY_PLAIN, 2,
(255, 255, 0), 2)
    cv2.putText(frame,textD , positionGs, cv2.FONT_HERSHEY_PLAIN, 2,
(255, 0, 255), 2)
    cv2.imshow("M.Frame", frame)
    cv2.imshow("Contours", mask)
    key = cv2.waitKey(30)
    #out.write(frame)
    if cv2.waitKey(1) & 0xFF == 27: # ESC Key
        break

cap.release()
cv2.destroyAllWindows()

```

## 5. การแยกสีและขนาด

```

#แยกสี แยกขนาด
import cv2

cap = cv2.VideoCapture("./video/video_C2.avi")
fourcc = cv2.VideoWriter_fourcc(*'XVID')
out = cv2.VideoWriter('./video_count/video_C8.avi', fourcc, 30,
(540,960))

object_detector = cv2.createBackgroundSubtractorMOG2(history=100,
varThreshold=40)
ret, frame = cap.read()
height, width = frame.shape[:2]

referenceLine = int(0.30*height) # 90%
positionText = (10,referenceLine-30)
positionOl = (10,referenceLine-150)
positionOm = (10,referenceLine-120)
positionGm = (10,referenceLine-90)
positionGs = (10,referenceLine-60)
ImageWidth = width

All = 0
Orange_l = 0 #ส้มใหญ่
Orange_m = 0 #ส้มเล็ก
Green_m = 0 #เขียวใหญ่
Green_s = 0 #เขียวเล็ก
tarea = 0

BoxInline_y = 1
BoxInline_x = 1
BoxInline_ol = 1
BoxInline_om = 1
BoxInline_gm = 1
BoxInline_gs= 1

while(cap.isOpened()):
    ret, frame = cap.read()
    height, width, _ = frame.shape
    hsv_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)
    cx = int(width / 2)
    cy = referenceLine + 45

    # Pick pixel value
    pixel_center = hsv_frame[cy, cx]
    bgr = frame[cy, cx]
    cb, cg, cr = int(bgr[0]), int(bgr[1]), int(bgr[2])

    hue_value = pixel_center[0]

    if hue_value < 28:

```

```

        color = "orange"
    elif hue_value < 50:
        color = "green"
    else:
        color = "blue"

    #print(pixel_center)

    BoxInline_ol = BoxInline_x
    BoxInline_om = BoxInline_x
    BoxInline_gm = BoxInline_x
    BoxInline_gs = BoxInline_x
    BoxInline_y = BoxInline_x
    BoxInline_x = 0

    roi = frame[0: 960,50: 490]
    mask = object_detector.apply(roi)
    _, mask = cv2.threshold(mask, 254, 255, cv2.THRESH_BINARY)
    contours, _ = cv2.findContours(mask, cv2.RETR_TREE,
cv2.CHAIN_APPROX_SIMPLE)

    for cnt in contours:
        area = cv2.contourArea(cnt)
        if area > 9000:
            x, y, w, h = cv2.boundingRect(cnt)
            cv2.rectangle(roi, (x, y), (x + w, y + h), (0, 255, 0), 3)
            if y < referenceLine and (y+h) > referenceLine:
                BoxInline_x = 1
                #print(area)
                if area > 9000 and area < 15000:
                    BoxInline_gs = 1
                elif area > 18000 and area < 26000:
                    if color == "orange":
                        BoxInline_om = 1
                    else:
                        BoxInline_gm = 1
                else :
                    BoxInline_ol = 1
    if BoxInline_y==0 and BoxInline_x==1:
        All = All + 1
        if BoxInline_ol == 1:
            Orange_l += 1
        elif BoxInline_om == 1:
            Orange_m += 1
        elif BoxInline_gm == 1:
            Green_m += 1
        elif BoxInline_gs == 1:
            Green_s += 1

    #textShow = str(All)+'='+str(BoxInline_x)+','+str(BoxInline_y)
    textShow = 'All = ' + str(All)
    textA = 'O_L = ' + str(Orange_l)

```

```

textB = 'O_M = ' + str(Orange_m)
textC = 'G_M = ' + str(Green_m)
textD = 'G_S = ' + str(Green_s)

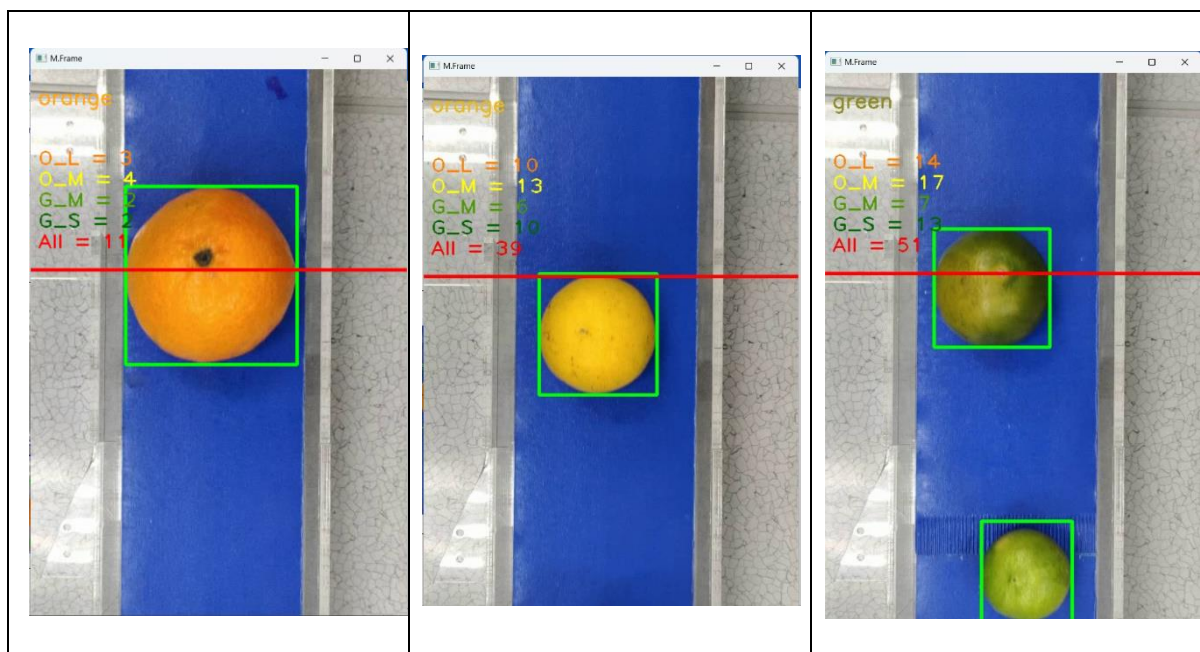
cv2.line(frame,(0,referenceLine),(ImageWidth,referenceLine),(0, 0,
255), 3)
cv2.putText(frame,textShow , positionText, cv2.FONT_HERSHEY_PLAIN,
2, (0, 0, 255), 2)
cv2.putText(frame,textA , positionO1, cv2.FONT_HERSHEY_PLAIN, 2, (0,
128, 255), 2)
cv2.putText(frame,textB , positionOm, cv2.FONT_HERSHEY_PLAIN, 2, (0,
255, 255), 2)
cv2.putText(frame,textC , positionGm, cv2.FONT_HERSHEY_PLAIN, 2, (0,
153, 76), 2)
cv2.putText(frame,textD , positionGs, cv2.FONT_HERSHEY_PLAIN, 2, (0,
102, 0), 2)
cv2.putText(frame, color, (10,50), 0,1, (cb, cg, cr),2)

cv2.imshow("M.Frame", frame)
#cv2.imshow("roi", roi)
key = cv2.waitKey(15)
out.write(frame)
if cv2.waitKey(1) & 0xFF == 27: # ESC Key
    break

cap.release()
cv2.destroyAllWindows()

```

## ผลการทำงาน



วิดีโอผลการทำงาน

<https://youtu.be/vzIKmsZ92fs>

บันทึกผลในตาราง

รายการ	จำนวน ชิ้นงานจริง	ผลการนับด้วย โปรแกรม	ผลต่าง	ถูกต้อง(%)
ผลรวม	100	103	-3	97.00%
ส้มลูกใหญ่ สีส้ม	28	29	-1	96.42%
ส้มลูกกลาง สีเหลือง	34	35	-1	97.06%
ส้มลูกกลาง สีเขียว	17	16	1	94.12%
ส้มลูกเล็ก สีเขียว	21	23	-2	90.48%

## สรุป

การทำงานของโปรแกรมมีความคลาดเคลื่อนเล็กน้อย เนื่องจากโปรแกรมที่ทำการตรวจจับสีจะจับเฟรมแรก ณ จุดตรงกลางของส้ม แล้วส้มบางชิ้นมีสีของเปลือกที่ใกล้เคียงกัน ตัวอย่างเช่น ส้มสีเขียวที่ใกล้สุก อาจจะมีเปลือกสีเหลืองเล็กน้อย แสงที่ตกกระทบก็มีส่วนทำให้สีคลาดเคลื่อนได้เช่นกัน

## ข้อเสนอแนะ

โปรเจกต์นี้เหมาะสำหรับใช้ตรวจจับวัตถุที่มีรูปร่างคล้ายคลึงกันและมีลักษณะที่เฉพาะตัว ไม่สามารถตรวจจับความเป็นเอกลักษณ์ของวัตถุได้ เช่น ส้ม ที่มีสี ขนาด และพื้นผิวใกล้เคียงกัน เป็นไปได้ยากมากที่จะหาจุดตำหนิที่เหมือนกันของส้มแต่ละลูก

หากต้องการหาความเป็นเอกลักษณ์ควรใช้การตรวจจับวัตถุแบบ Keypoint Matching หรือใช้ YOLO - Real-Time Object Detection

## อ้างอิง

- asmitapatidar2023. Count number of Object using Python-OpenCV. [Online]. 2016, Available from: <https://www.geeksforgeeks.org/count-number-of-object-using-python-opencv/> [ 30 ตุลาคม 2565 ]
- iBook Engineering. ระบบสี HSV [Online]. 2015, Available from: [http://ibookengineering.blogspot.com/2015/07/hsv\\_13.html](http://ibookengineering.blogspot.com/2015/07/hsv_13.html) [ 30 ตุลาคม 2565 ]
- Pysource. Simple Color recognition with Opencv and Python. [Online]. 2021, Available from: <https://youtu.be/t71sQ6WY7L4> [ 30 ตุลาคม 2565 ]