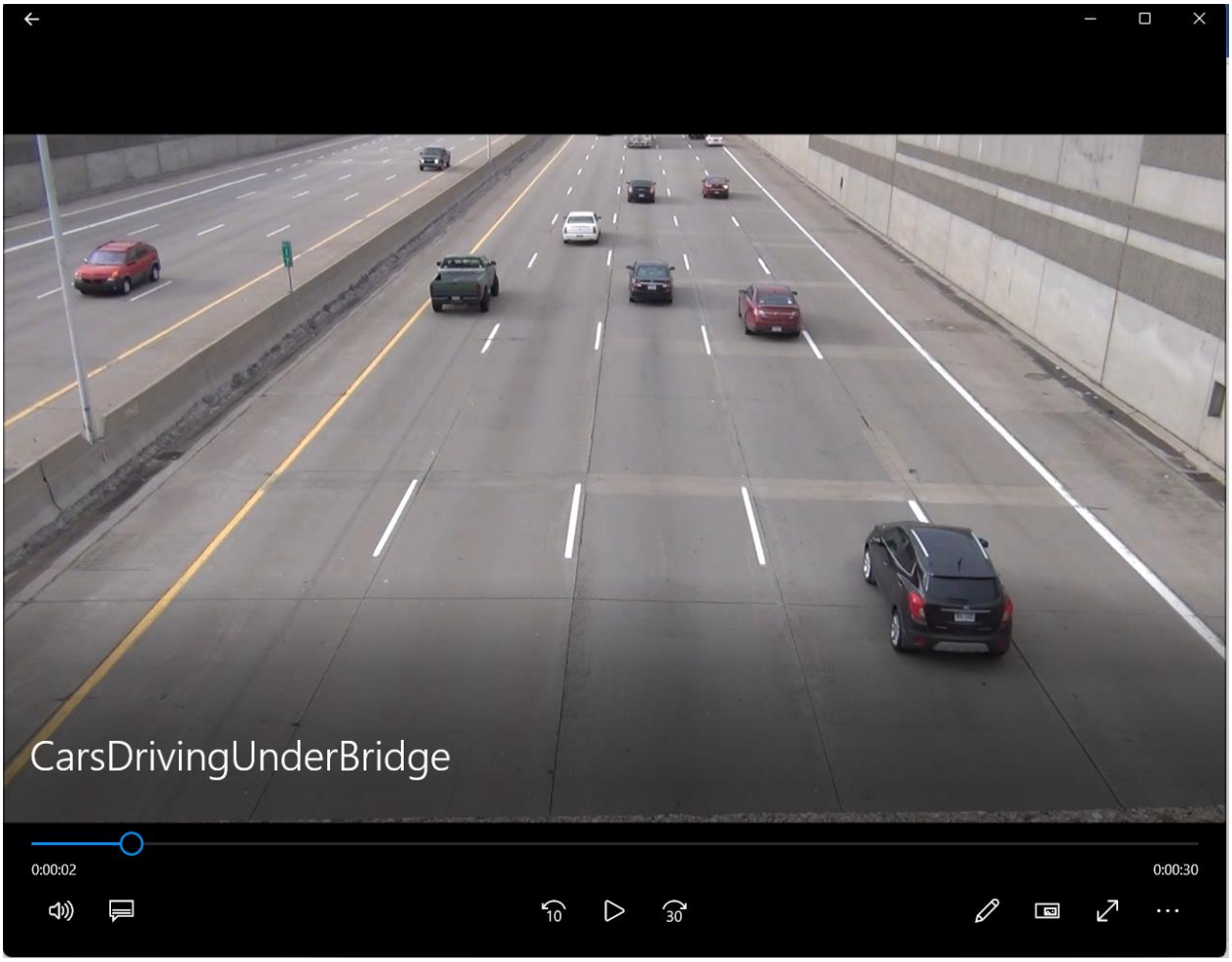
รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

Lab 18 Track & Counting Cars/People

คำสั่ง

1. หาภาพที่มีพื้นหลังคงที่ / กล้องไม่เคลื่อนที่ (Stationary CAM) และมีวัตถุเคลื่อนที่ในเฟรมเพื่อให้สามารถใช้วิธีการ Background Subtraction ตรวจจับและนับได้ หรือใช้ vdo ที่ให้จากLabนี้



รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

2. ทำ Background Subtraction / Blob Tracking / และนับวัตถุที่ผ่านไปผ่านมา โดยสามารถปรับใช้ Code จาก ใน Lab นี้ ไปปรับพื้นที่ตรวจจับ/นับให้เหมาะสม (สามารถปรับใช้ code ที่ให้มาด้านล่าง หรือใช้ cvbloblib)

Code :

```python
import math
import cv2 as cv
import numpy as np
from numpy import random

capture = cv.VideoCapture('CarsDrivingUnderBridge.mp4')
#capture = cv.VideoCapture('circle_move.mp4')


backSub = cv.createBackgroundSubtractorMOG2(history=None,varThreshold=None,detectShadows=True)
print("===========SHOW DEFAULT BackgroundSubtractor PARAMETER==========") #Print default value
print(f"getHistory={backSub.getHistory()}\ngetNMixtures={backSub.getNMixtures()}
\ngetDetectShadows={backSub.getDetectShadows()}\nvarThreshold={backSub.getVarThreshold()}")
#backSub.setHistory(600) # Amount of mean sigma of BG
#backSub.setNMixtures(2) # Number of Gaussian Distribution
backSub.setDetectShadows(True) # Detect Shadow
backSub.setVarThreshold(100) # Threshold Motion

class TrackedBlob:
    id = None
    color = None
    alive = True # if this flag false = marked to remove
```

รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

```
    def __init__(self,blob_id,xywh):
        self.id = blob_id
        (self.x,self.y,self.w,self.h) = xywh
        c = random.randint(255, size=(3))
        self.color = [c[0].item(),c[1].item(),c[2].item()]
        self.path = []
        self.nlost = 0
        self.allLife = 1
        self.matchedStatus = True
        self.alreadyCounted = False # use for count function
    def XYWH(self):
        return [self.x,self.y,self.w,self.h]
    def calDiffBlobWithContour(self, contour_xywh, weightPosition = 0.5, weightScale = 0.5):
        (xContour,yContour,wContour,hContour) = contour_xywh # incoming object
        return (math.sqrt((xContour - self.x)**2 + (yContour - self.y)**2) * weightPosition) +
        (math.sqrt((wContour - self.w)**2 + (hContour - self.h)**2) * weightScale)
    def updateBlob(self, status, xywh=[0,0,0,0]):
        if (status): # Matched
            self.nlost = 0
            self.allLife += 1
            self.path.append(xywh)
            (self.x,self.y,self.w,self.h) = xywh # update corrent position
        else : # Not Matched
            self.nlost += 1
            if(self.nlost >= 4): # if not matched xx frame will be marked to remove
                self.alive = False
    def setMatched(self):
        self.matchedStatus = True
    def resetMatched(self):
        self.matchedStatus = False
    def getMatched(self):
        return self.matchedStatus
    def getNLost(self):
        return self.nlost
    def setCounted(self):
        self.alreadyCounted = True
    def getCounted(self):
        return self.alreadyCounted
    def getAllLife(self):
        return self.allLife
    def getPath(self):
        return self.path
```

รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

```
63
64  class BlobTracker:
65      def __init__(self, distanceThreshold = 60):
66          self.TrackedBlob_Table = []
67          self.distanceThreshold = distanceThreshold
68          self.lastID = 0
69      def setDistanceThreshold(self, distanceThreshold = 60):
70          self.distanceThreshold = distanceThreshold
71      def getTrackedBlob_Table(self):
72          return self.TrackedBlob_Table
73      def trackXYWHs(self, ContourXYWHs): # ContourXYWHs -> Incomming Objects Table
74          # if no TrackedBlob in Table
75          if(len(self.TrackedBlob_Table)==0):
76              for XYWH in ContourXYWHs:
77                  self.addNewTrackedBlob(XYWH)
78          # Matching Incomming Objects with all
79          else :
80              for XYWH in ContourXYWHs: # accessing (incoming object) in each contour to update
                or new tracked blob
81                  minDistance = 1000000000 #          -1 = not match with anyone
82                  minDistanceTrackedBlobID = -1 #     -1 = not match with anyone
83                  for TrackedBlobIter in self.TrackedBlob_Table:
84                      if(not TrackedBlobIter.getMatched()): # if didnot match yet
85                          diffValue = TrackedBlobIter.calDiffBlobWithContour(XYWH)
86                          #print(f'diffValue {diffValue} of Blob={XYWH} with TrackedBlob#
                            {TrackedBlobIter.id}={TrackedBlobIter.XYWH()}({TrackedBlobIter.color})')
87                          if(diffValue <= self.distanceThreshold and diffValue < minDistance): #
                            ถ้าเข้าเกณฑ์ และมีค่าน้อยสุดก็ marked ไว้ก่อน
88                              minDistanceTrackedBlobID = TrackedBlobIter.id
89                              minDistance = diffValue
90                  # if matched some TrackedBlob -> update
91                  if(minDistanceTrackedBlobID!=-1):
92                      self.updateTrackedBlob(minDistanceTrackedBlobID, XYWH)
93                  else:
94                      self.addNewTrackedBlob(XYWH)
95          # if some TrackedBlob has not be updated -> set Lost
96          for TrackedBlobIter in self.TrackedBlob_Table:
97              if(not TrackedBlobIter.getMatched()): # if didnot match yet
98                  TrackedBlobIter.updateBlob(False)
99          # remove all dead (not alive)
100         numTrackedBlob = len(self.TrackedBlob_Table)
101         i=0
```

รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

```
        while i < numTrackedBlob:
            if(not self.TrackedBlob_Table[i].alive): # if didnot match yet
                self.TrackedBlob_Table.pop(i) # remove
                numTrackedBlob-=1 # move down counter
                i-=1 # move down iterator
            i+=1
        # reset all Match Status
        for TrackedBlobIter in self.TrackedBlob_Table:
            TrackedBlobIter.resetMatched()
    def addNewTrackedBlob(self, XYWH):
        self.TrackedBlob_Table.append(TrackedBlob(self.lastID,XYWH))
        self.lastID += 1
    def updateTrackedBlob(self, id, XYWH):
        for TrackedBlobIter in self.TrackedBlob_Table:
            if TrackedBlobIter.id == id:
                TrackedBlobIter.updateBlob(True,XYWH) # update
                TrackedBlobIter.setMatched() # marked already Mateched
                break
    def drawTrackedBlobs(self, image, fontSize=1.0, thickness=2, drawPath=True):
        for TrackedBlobIter in self.TrackedBlob_Table:
            if TrackedBlobIter.getNLost()==0 and TrackedBlobIter.alive : # and TrackedBlobIter.
            getAllLife()>1'''
                color = TrackedBlobIter.color
                (x,y,w,h) = TrackedBlobIter.XYWH()
                cv.rectangle(image, (x,y), (x+w,y+h), color, thickness)
                cv.putText(image, str(TrackedBlobIter.id), (x+2,y+2), cv.FONT_HERSHEY_SIMPLEX,
                fontSize, color, thickness)
                # plot path
                if(drawPath):
                    histXYWHs = TrackedBlobIter.getPath()
                    histCenter = []
                    for i in range(len(histXYWHs)) :
                        (xHist,yHist,wHist,hHist) = histXYWHs[i]
                        histCenter.append((xHist+(wHist//2),yHist+(hHist//2)))
                        if(i!=0):
                            cv.line(image,histCenter[i-1],histCenter[i],color)

class BlobExtractor:
    def __init__(self):
        self.contours = None
        self.hierarchy = None
        self.XYWHs = None
```

รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

```
142         self.colors = None
143     def execute(self,segmented_bin_img):
144         edge16S_img = cv.Laplacian(segmented_bin_img, cv.CV_16S, ksize=3)
145         edge_img = cv.convertScaleAbs(edge16S_img)
146         self.contours, self.hierarchy = cv.findContours(edge_img, cv.RETR_EXTERNAL, cv.
            CHAIN_APPROX_SIMPLE)
147         self.XYWHs = [ cv.boundingRect(contour) for contour in self.contours]
148     def filterMinArea(self,min):
149         ''' filter out(remove) countours that  have area < min'''
150         temp_contours = []
151         for i,cnt in enumerate(self.contours):
152             (x,y,w,h) = cv.boundingRect(cnt)
153             if((w*h)>=min):
154                 temp_contours.append(cnt) # [x,y,w,h]
155         self.contours = temp_contours.copy()
156         self.XYWHs = [ cv.boundingRect(contour) for contour in self.contours]
157     def filterInArea(self,XYWH):
158         ''' filter only countour in Area XYWH'''
159         temp_contours = []
160         (xmin,ymin) = XYWH[:2]
161         xmax = xmin + XYWH[2]
162         ymax = ymin + XYWH[3]
163         for i,cnt in enumerate(self.contours):
164             (x,y,w,h) = cv.boundingRect(cnt)
165             if( (x>=xmin and x<=xmax) and (y>=ymin and y<=ymax)):
166                 temp_contours.append(cnt) # [x,y,w,h]
167         self.contours = temp_contours.copy()
168         self.XYWHs = [ cv.boundingRect(contour) for contour in self.contours]
169     def getContours(self):
170         return self.contours
171     def getXYWHs(self):
172         return self.XYWHs
173
174 distanceThreshold = 60
175 mainBlobTracker = BlobTracker(distanceThreshold); # create main tracker
176 cv.namedWindow('Frame',cv.WINDOW_NORMAL)
177 def changeDistanceThreshold(x):
178     global distanceThreshold
179     distanceThreshold = cv.getTrackbarPos('DistanceThreshold','Frame')
180     mainBlobTracker.setDistanceThreshold(distanceThreshold)
181 cv.createTrackbar('DistanceThreshold', 'Frame', distanceThreshold, 400,
    changeDistanceThreshold)
```

รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

```
182
183 y_startCount = 20 # เริ่มเส้นนับ
184 y_endCount = 200 # จบเส้นนับ
185 blobCount = 0 # number of blob which passed line
186
187 while True:
188     ret, frame = capture.read()
189     id_frame = capture.get(cv.CAP_PROP_POS_FRAMES)
190     if frame is None:
191         break
192
193     fgMask = backSub.apply(frame,learningRate=0.005) # learningRate
194
195     cv.rectangle(frame, (10, 2), (100,20), (255,255,255), -1)
196     cv.putText(frame, str(id_frame), (15, 15), cv.FONT_HERSHEY_SIMPLEX, 0.5 , (0,0,0))
197
198     _,fgMask = cv.threshold(fgMask, 200, 255, cv.THRESH_BINARY)
199     kernel = cv.getStructuringElement(cv.MORPH_RECT, (9,9))
200     fgMask = cv.dilate(fgMask, kernel, iterations=1)
201
202     Blob = BlobExtractor()
203     Blob.execute(fgMask)
204     Blob.filterMinArea(2000)
205     FocusArea = [100,100,950,700] # บริเวณที่แสดง Blob
206     Blob.filterInArea(FocusArea)
207
208
209     (H,W) = fgMask.shape
210     contours_img = np.zeros((H,W,3),dtype=np.uint8)
211     if(len(Blob.getContours())>=1):
212         cv.drawContours(contours_img, Blob.getContours(), -1, (255,0,0))
213
214     # plot tracking area
215     (xFA,yFA,wFA,hFA) = FocusArea
216     cv.rectangle(frame, (xFA,yFA), (xFA+wFA,yFA+hFA), (255,255,255)), cv.rectangle
        (contours_img, (xFA,yFA), (xFA+wFA,yFA+hFA), (255,255,255))
217     cv.putText(frame, 'Tracking Zone', (xFA+5, yFA+20), cv.FONT_HERSHEY_SIMPLEX, .7 ,(255,255,
        255), 1),cv.putText(contours_img, 'Tracking Zone', (xFA+5, yFA+20), cv.
        FONT_HERSHEY_SIMPLEX, .7 ,(255,255,255), 1)
218
219     # plot counting line
220     (fHeight,fWidth) = frame.shape[:2]
```

รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

```
221     cv.putText(frame, 'Counting Zone', (50, y_startCount+30), cv.FONT_HERSHEY_SIMPLEX, 1 ,(0,
        255,255), 2),cv.putText(contours_img, 'Counting Zone', (50, y_startCount+30), cv.
        FONT_HERSHEY_SIMPLEX, 1 ,(0,255,255), 2)
222     cv.line(frame,(0,y_startCount),(fWidth,y_startCount),(0,255,255),2),    cv.line(frame,(0,
        y_endCount),(fWidth,y_endCount),(0,255,255),2)
223     cv.line(contours_img,(0,y_startCount),(fWidth,y_startCount),(0,255,255),2),    cv.line
        (contours_img,(0,y_endCount),(fWidth,y_endCount),(0,255,255),2)
224
225     if(id_frame>100):
226         # blob tracking
227         mainBlobTracker.trackXYWHs(Blob.getXYWHs())
228         mainBlobTracker.drawTrackedBlobs(frame,fontSize=1,drawPath=True)
229         mainBlobTracker.drawTrackedBlobs(contours_img,fontSize=1,drawPath=True)
230         # blob counting
231         TrackedBlob_Table = mainBlobTracker.getTrackedBlob_Table()
232         for TrackedBlobIter in TrackedBlob_Table:
233             (_,y,_,_) = TrackedBlobIter.XYWH()
234             # if be in y_startCount && y_endCount
235             if(y>=y_startCount and y<=y_endCount):
236                 # if alive, is not counted and active
237                 if(TrackedBlobIter.getNLost()==0 and (not TrackedBlobIter.getCounted()) and
                    TrackedBlobIter.alive and TrackedBlobIter.getAllLife()>2):
238                     blobCount+=1
239                     TrackedBlobIter.setCounted()
240         cv.putText(frame, 'Counter : '+str(blobCount), (500, 35), cv.FONT_HERSHEY_SIMPLEX, 1.2
            , (128,128,255),2),cv.putText(contours_img, 'Counter : '+str(blobCount), (500, 30), cv.
            FONT_HERSHEY_SIMPLEX, 1 , (128,128,255))
241     cv.imshow('Frame', frame)
242     cv.imshow('FG_Mask', fgMask)
243     cv.imshow('Contours', contours_img)
244
245     keyboard = cv.waitKey(10)
246     if keyboard == 'q' or keyboard == 27:
247         break
248
```

รหัสนักศึกษา 1624041400008-2 ชื่อ-นามสกุล นายโชคชัย แจ่มน้อย

กำหนดส่ง 5/11/2564

3. ทำ Screenshot ผลลัพธ์ที่เป็น Motion หรืออัด VDO ผลลัพธ์ใส่ GoogleDrive

ผลลัพธ์ :

https://drive.google.com/file/d/12qnEeADXUs8F5Y4lq21O8bHPGvjxQTEe/view?usp=sharing