

# Assignment 03

Index No: 190108X

Name: Chathuranga M.M.P.

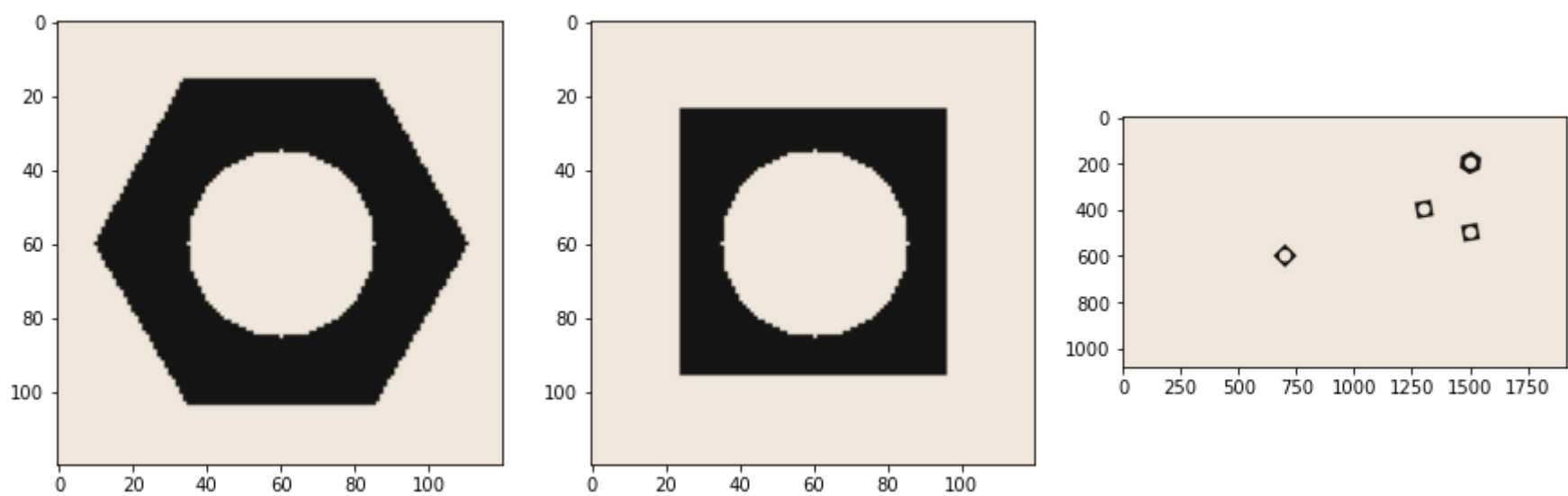
GitHub link: [https://github.com/ChathurangaMMP/EN2550\\_Image\\_Processing\\_Exercises/tree/master/Assignment%203](https://github.com/ChathurangaMMP/EN2550_Image_Processing_Exercises/tree/master/Assignment%203)

## Connected Component Analysis

```
In [1]: import cv2 as cv
import numpy as np
import matplotlib.pyplot as plt

hexnut_template = cv.imread('hexnut_template.png', cv.IMREAD_COLOR)
squarenut_template = cv.imread('squarenut_template.png', cv.IMREAD_COLOR)
conveyor_f100 = cv.imread('conveyor_f100.png', cv.IMREAD_COLOR)

fig, ax = plt.subplots(1,3,figsize=(15,15))
ax[0].imshow(cv.cvtColor(hexnut_template, cv.COLOR_RGB2BGR))
ax[1].imshow(cv.cvtColor(squarenut_template, cv.COLOR_RGB2BGR))
ax[2].imshow(cv.cvtColor(conveyor_f100, cv.COLOR_RGB2BGR))
plt.show()
```



## Otsu's thresholding

```
In [8]: hexnut_gray = cv.cvtColor(hexnut_template, cv.COLOR_BGR2GRAY)
squarenut_gray=cv.cvtColor(squarenut_template, cv.COLOR_BGR2GRAY)
conveyor_gray=cv.cvtColor(conveyor_f100, cv.COLOR_BGR2GRAY)

th1,hext = cv.threshold(hexnut_gray,0,255,cv.THRESH_BINARY+cv.THRESH_OTSU)
th2,square = cv.threshold(squarenut_gray,0,255,cv.THRESH_BINARY+cv.THRESH_OTSU)
th3,belt = cv.threshold(conveyor_gray,0,255,cv.THRESH_BINARY+cv.THRESH_OTSU)

print('Threshold value for hexnut template:',th1)
print('Threshold value for squarenut template:',th2)
print('Threshold value for conveyor belt:',th3)
```

Threshold value for hexnut template: 20.0  
Threshold value for squarenut template: 20.0  
Threshold value for conveyor belt: 20.0

```
In [10]: fig,ax=plt.subplots(1,3,figsize=(15,15))
ax[0].imshow(cv.cvtColor(hext,cv.COLOR_RGB2BGR))
ax[0].axis('off')
ax[1].imshow(cv.cvtColor(square,cv.COLOR_RGB2BGR))
ax[1].axis('off')
ax[2].imshow(cv.cvtColor(belt,cv.COLOR_RGB2BGR))
ax[2].axis('off')
plt.show()
```



## Morphological closing

```
In [16]: hex_binary = cv.threshold(hexnut_gray, 127, 255, cv.THRESH_BINARY)[1]
sqr_binary = cv.threshold(squarenut_gray, 127, 255, cv.THRESH_BINARY)[1]
belt_binary = cv.threshold(conveyor_gray, 127, 255, cv.THRESH_BINARY)[1]
kernel = np.ones((3,3),np.uint8)
hex_closing = cv.morphologyEx(np.invert(hex_binary), cv.MORPH_CLOSE, kernel)
square_closing = cv.morphologyEx(np.invert(sqr_binary), cv.MORPH_CLOSE, kernel)
belt_closing = cv.morphologyEx(np.invert(belt_binary), cv.MORPH_CLOSE, kernel)
```

```
In [18]: fig,ax=plt.subplots(1,3,figsize=(15,15))
ax[0].imshow(cv.cvtColor(np.invert(hex_closing),cv.COLOR_RGB2BGR))
ax[0].axis('off')
ax[1].imshow(cv.cvtColor(np.invert(square_closing),cv.COLOR_RGB2BGR))
ax[1].axis('off')
ax[2].imshow(cv.cvtColor(np.invert(belt_closing),cv.COLOR_RGB2BGR))
ax[2].axis('off')
plt.show()
```



## Connected components analysis for Hexnut template

```
In [19]: connectivity = 4
hex_num_labels,hex_labels,hex_stat,hex_cent = cv.connectedComponentsWithStats(hext, connectivity, cv.CV_32S)
print('There are {} connected components in Hexnut template.'.format(hex_num_labels))
print('Statistics:')
print(hex_stat)
print()
print('Centroids:')
print(hex_cent)
```

There are 3 connected components in Hexnut template.

Statistics:

```
[[ 10  16 101  88 4724]
 [  0   0 120 120 7715]
 [ 35  35  51  51 1961]]
```

Centroids:

```
[[59.83361558 59.22290432]
 [59.16863253 59.54257939]
 [60.         60.         ]]
```

## Connected components analysis for Squarenut template

```
In [20]: connectivity = 4
sqr_num_labels,sqr_labels,sqr_stat,sqr_cent = cv.connectedComponentsWithStats(square, connectivity, cv.CV_32S)
print('There are {} connected components in Squarenut template.'.format(sqr_num_labels))
print('Statistics:')
print(sqr_stat)
print()
print('Centroids:')
print(sqr_cent)
```

There are 3 connected components in Squarenut template.

Statistics:

```
[[ 24  24  72  72 3223]
 [  0   0 120 120 9216]
 [ 35  35  51  51 1961]]
```

Centroids:

```
[[59.19578033 59.19578033]
 [59.5         59.5         ]
 [60.         60.         ]]
```

## Connected components analysis for Conveyor belt

```
In [21]: connectivity = 4
belt_num_labels,belt_labels,belt_stat,belt_cent = cv.connectedComponentsWithStats(belt, connectivity, cv.CV_32S)
print('There are {} connected components in Conveyor belt.'.format(belt_num_labels))
print('Statistics:')
print(belt_stat)
```

```
print()
print('Centroids:')
print(belt_cent)
```

There are 6 connected components in Conveyor belt.

Statistics:

```
[[ 650 150 896 501 13938]
 [ 0 0 1920 1080 2051818]
 [ 1475 175 51 51 1961]
 [ 1275 375 51 51 1961]
 [ 1475 475 51 51 1961]
 [ 675 575 51 51 1961]]
```

Centroids:

```
[[1274.92050509 400.1106328 ]
 [ 956.24678115 540.8845999 ]
 [1500. 200. ]
 [1300. 400. ]
 [1500. 500. ]
 [ 700. 600. ]]
```

In [25]:

```
fig,ax=plt.subplots(1,3,figsize=(15,15))
ax[0].imshow(hex_labels)
ax[0].axis('off')
ax[1].imshow(sqr_labels)
ax[1].axis('off')
ax[2].imshow(belt_labels)
ax[2].axis('off')
plt.show()
```

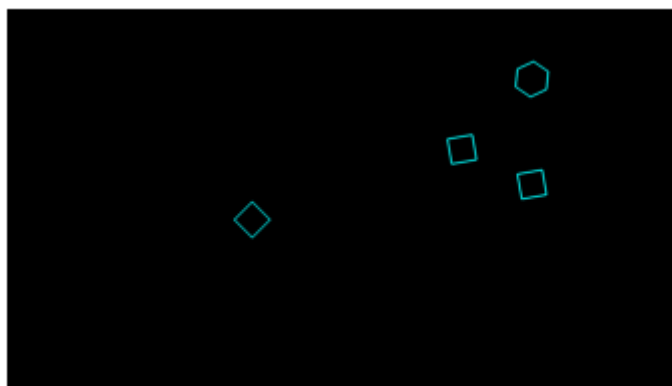


## Contour analysis

In [10]:

```
black_img=np.zeros(conveyor_f100.shape)
contours, hierarchy = cv.findContours(belt, cv.RETR_TREE, cv.CHAIN_APPROX_SIMPLE)
cnt = [contours[i] for i in range(1,9,2)]
cv.drawContours(black_img, cnt, -1, (0,255,150), 3)
plt.imshow(black_img)
plt.axis('off')
plt.show()
```

Clipping input data to the valid range for imshow with RGB data ([0..1] for floats or [0..255] for integers).



## Detecting Objects on a Synthetic Conveyor

In [11]:

```
cv.namedWindow('Conveyor', cv.WINDOW_NORMAL)
cap = cv.VideoCapture('conveyor.mp4')
f = 0
frames=[]
frame = []
while cap.isOpened():
    ret, frame = cap.read()
    frames.append(frame)
    if not ret:
        print("Can't receive frame (stream end?). Exiting.")
        break

    f += 1
    text = 'Frame:' + str(f)
    cv.putText(frame,text , (100, 100), cv.FONT_HERSHEY_COMPLEX, 1, (0,250,0), 1, cv.LINE_AA)
```

```

cv.imshow('Conveyor', frame)

if cv.waitKey(1) == ord('q'):
    break

cap.release()
cv.destroyAllWindows()

```

Can't receive frame (stream end?). Exiting.

In [12]: *# Count the number of matching hexagonal nuts in conveyor\_f100.png*

```

total_matches=0
for i in range(len(contours)):
    ret = cv.matchShapes(contours[i],contours[7],1,0.0)
    if ret==0.0:
        total_matches+=1
print('Number of matching hexagonal nuts = ',total_matches)

```

Number of matching hexagonal nuts = 1

In [15]:

```

total_nuts=0
frame_count=0
detected=[]
for frm in frames[:-1]:
    frame_total=0
    img_gray = cv.cvtColor(frm,cv.COLOR_BGR2GRAY)
    ret_f,thresh = cv.threshold(img_gray,0,255,cv.THRESH_BINARY+cv.THRESH_OTSU)
    contours_frame,hierarchy_frame = cv.findContours(thresh,cv.RETR_TREE, cv.CHAIN_APPROX_SIMPLE)

    for cnt in contours_frame:
        ret_count = cv.matchShapes(cnt,contours[7],1,0.0)
        if ret_count<=0.006 and abs(cv.contourArea(cnt)-cv.contourArea(contours[7]))<=800:
            temp=0
            for c in detected:
                if np.sum(c)-np.sum(cnt)<75000:
                    temp=1
            if temp==0:
                detected.append(cnt)
                total_nuts+=1
            frame_total+=1

    print('Number of matching hexagonal nuts in frame {} is {}'.format(frame_count+1,frame_total))
    print('Total number of matching hexagonal nuts upto frame {} is {}'.format(frame_count+1,total_nuts))
    frame_count+=1
    in_text= 'In frame : '+str(frame_total)
    upto_text='Upto frame : '+str(total_nuts)
    cv.putText(img=frm, text=in_text, org=(50, 70), fontFace=cv.FONT_HERSHEY_TRIPLEX, fontScale=2, color=(0, 0, 255),thickness=1)
    cv.putText(img=frm, text=upto_text, org=(50, 150), fontFace=cv.FONT_HERSHEY_TRIPLEX, fontScale=2, color=(0, 0, 255),thickness=1)

```

```

Number of matching hexagonal nuts in frame 1 is 0
Total number of matching hexagonal nuts upto frame 1 is 0
Number of matching hexagonal nuts in frame 2 is 0
Total number of matching hexagonal nuts upto frame 2 is 0
Number of matching hexagonal nuts in frame 3 is 0
Total number of matching hexagonal nuts upto frame 3 is 0
Number of matching hexagonal nuts in frame 4 is 0
Total number of matching hexagonal nuts upto frame 4 is 0
Number of matching hexagonal nuts in frame 5 is 0
Total number of matching hexagonal nuts upto frame 5 is 0
Number of matching hexagonal nuts in frame 6 is 0
Total number of matching hexagonal nuts upto frame 6 is 0
Number of matching hexagonal nuts in frame 7 is 0
Total number of matching hexagonal nuts upto frame 7 is 0
Number of matching hexagonal nuts in frame 8 is 0
Total number of matching hexagonal nuts upto frame 8 is 0
Number of matching hexagonal nuts in frame 9 is 0
Total number of matching hexagonal nuts upto frame 9 is 0
Number of matching hexagonal nuts in frame 10 is 0
Total number of matching hexagonal nuts upto frame 10 is 0
Number of matching hexagonal nuts in frame 11 is 0
Total number of matching hexagonal nuts upto frame 11 is 0
Number of matching hexagonal nuts in frame 12 is 0
Total number of matching hexagonal nuts upto frame 12 is 0
Number of matching hexagonal nuts in frame 13 is 0
Total number of matching hexagonal nuts upto frame 13 is 0
Number of matching hexagonal nuts in frame 14 is 0
Total number of matching hexagonal nuts upto frame 14 is 0
Number of matching hexagonal nuts in frame 15 is 0
Total number of matching hexagonal nuts upto frame 15 is 0
Number of matching hexagonal nuts in frame 16 is 0
Total number of matching hexagonal nuts upto frame 16 is 0
Number of matching hexagonal nuts in frame 17 is 0
Total number of matching hexagonal nuts upto frame 17 is 0
Number of matching hexagonal nuts in frame 18 is 0
Total number of matching hexagonal nuts upto frame 18 is 0
Number of matching hexagonal nuts in frame 19 is 0
Total number of matching hexagonal nuts upto frame 19 is 0
Number of matching hexagonal nuts in frame 20 is 0
Total number of matching hexagonal nuts upto frame 20 is 0
Number of matching hexagonal nuts in frame 21 is 0

```

[illegible]

[illegible]

[illegible]



[illegible]



[illegible]

[illegible]

[illegible]

Total number of matching hexagonal nuts upto frame 343 is 4  
 Number of matching hexagonal nuts in frame 344 is 2  
 Total number of matching hexagonal nuts upto frame 344 is 4  
 Number of matching hexagonal nuts in frame 345 is 2  
 Total number of matching hexagonal nuts upto frame 345 is 4  
 Number of matching hexagonal nuts in frame 346 is 2  
 Total number of matching hexagonal nuts upto frame 346 is 4  
 Number of matching hexagonal nuts in frame 347 is 1  
 Total number of matching hexagonal nuts upto frame 347 is 4  
 Number of matching hexagonal nuts in frame 348 is 1  
 Total number of matching hexagonal nuts upto frame 348 is 4  
 Number of matching hexagonal nuts in frame 349 is 1  
 Total number of matching hexagonal nuts upto frame 349 is 4  
 Number of matching hexagonal nuts in frame 350 is 1  
 Total number of matching hexagonal nuts upto frame 350 is 4  
 Number of matching hexagonal nuts in frame 351 is 1  
 Total number of matching hexagonal nuts upto frame 351 is 4  
 Number of matching hexagonal nuts in frame 352 is 1  
 Total number of matching hexagonal nuts upto frame 352 is 4  
 Number of matching hexagonal nuts in frame 353 is 1  
 Total number of matching hexagonal nuts upto frame 353 is 4  
 Number of matching hexagonal nuts in frame 354 is 1  
 Total number of matching hexagonal nuts upto frame 354 is 4  
 Number of matching hexagonal nuts in frame 355 is 1  
 Total number of matching hexagonal nuts upto frame 355 is 4  
 Number of matching hexagonal nuts in frame 356 is 1  
 Total number of matching hexagonal nuts upto frame 356 is 4  
 Number of matching hexagonal nuts in frame 357 is 0  
 Total number of matching hexagonal nuts upto frame 357 is 4  
 Number of matching hexagonal nuts in frame 358 is 0  
 Total number of matching hexagonal nuts upto frame 358 is 4  
 Number of matching hexagonal nuts in frame 359 is 0  
 Total number of matching hexagonal nuts upto frame 359 is 4  
 Number of matching hexagonal nuts in frame 360 is 0  
 Total number of matching hexagonal nuts upto frame 360 is 4  
 Number of matching hexagonal nuts in frame 361 is 0  
 Total number of matching hexagonal nuts upto frame 361 is 4  
 Number of matching hexagonal nuts in frame 362 is 0  
 Total number of matching hexagonal nuts upto frame 362 is 4  
 Number of matching hexagonal nuts in frame 363 is 0  
 Total number of matching hexagonal nuts upto frame 363 is 4  
 Number of matching hexagonal nuts in frame 364 is 0  
 Total number of matching hexagonal nuts upto frame 364 is 4  
 Number of matching hexagonal nuts in frame 365 is 0  
 Total number of matching hexagonal nuts upto frame 365 is 4  
 Number of matching hexagonal nuts in frame 366 is 0  
 Total number of matching hexagonal nuts upto frame 366 is 4  
 Number of matching hexagonal nuts in frame 367 is 0  
 Total number of matching hexagonal nuts upto frame 367 is 4  
 Number of matching hexagonal nuts in frame 368 is 0  
 Total number of matching hexagonal nuts upto frame 368 is 4  
 Number of matching hexagonal nuts in frame 369 is 0  
 Total number of matching hexagonal nuts upto frame 369 is 4  
 Number of matching hexagonal nuts in frame 370 is 0  
 Total number of matching hexagonal nuts upto frame 370 is 4  
 Number of matching hexagonal nuts in frame 371 is 0  
 Total number of matching hexagonal nuts upto frame 371 is 4  
 Number of matching hexagonal nuts in frame 372 is 0  
 Total number of matching hexagonal nuts upto frame 372 is 4  
 Number of matching hexagonal nuts in frame 373 is 0  
 Total number of matching hexagonal nuts upto frame 373 is 4  
 Number of matching hexagonal nuts in frame 374 is 0  
 Total number of matching hexagonal nuts upto frame 374 is 4  
 Number of matching hexagonal nuts in frame 375 is 0  
 Total number of matching hexagonal nuts upto frame 375 is 4  
 Number of matching hexagonal nuts in frame 376 is 0  
 Total number of matching hexagonal nuts upto frame 376 is 4  
 Number of matching hexagonal nuts in frame 377 is 0  
 Total number of matching hexagonal nuts upto frame 377 is 4  
 Number of matching hexagonal nuts in frame 378 is 0  
 Total number of matching hexagonal nuts upto frame 378 is 4  
 Number of matching hexagonal nuts in frame 379 is 0  
 Total number of matching hexagonal nuts upto frame 379 is 4  
 Number of matching hexagonal nuts in frame 380 is 0  
 Total number of matching hexagonal nuts upto frame 380 is 4

In [14]:

```

# Writing the video

frame_array = frames[:-1]
shape = (1080, 1920, 3)

# Your code here

out = cv.VideoWriter('./conveyor_result_190108X.mp4', cv.VideoWriter_fourcc(*'h264'), 30, (shape[1], shape[0]))

for i in range(len(frame_array)):
    cv.imshow('Frame', frame_array[i])
    if cv.waitKey(1) == ord('q'):
        break
    out.write(frame_array[i])
  
```

```
out.release()  
cv.destroyAllWindows()
```