



## GÖRÜNTÜ İŞLEME YARDIMCI NOTLARI -2023-



### GENEL FONKSİYON ve KOMUTLAR

#### Resim birleştirme (Yatay)

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
x=cv2.imread('balonlar2.jpg')
y=cv2.imread('anahtar.jpg')
print("x boyutu:"+str(x.shape))
print("y boyutu:"+str(y.shape))
#iki resmi aynı boyutlara getirmeliyiz.
y=cv2.resize(y,(226,223))
#iki resmi yan yana birleştirildi.
z=cv2.hconcat([x,y])
cv2.imshow('yatay birlesik resim',z)
cv2.waitKey(0)
cv2.destroyAllWindows()
```



#### Resim birleştirme (Yatay)

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
x=cv2.imread('balonlar2.jpg')
y=cv2.imread('anahtar.jpg')
print("x boyutu:"+str(x.shape))
print("y boyutu:"+str(y.shape))
#iki resmi aynı boyutlara getirmeliyiz.
y=cv2.resize(y,(226,223))
```

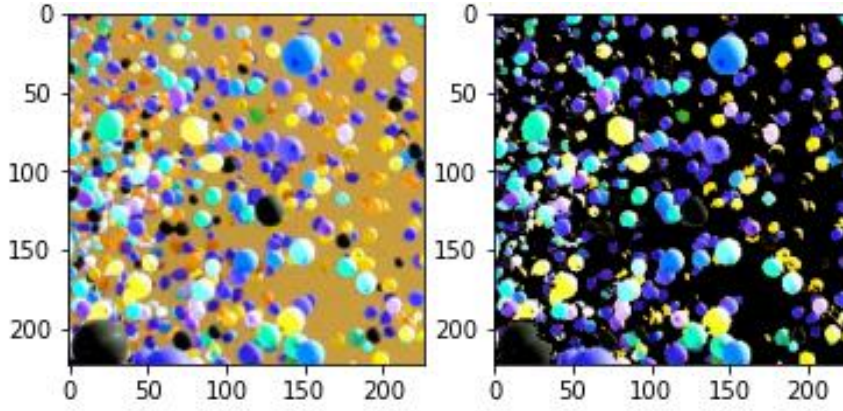
*#iki resmi yan yana birleştirildi.*

```
z=cv2.vconcat([x,y])  
cv2.imshow('yatay birlesik resim',z)  
cv2.waitKey(0)  
cv2.destroyAllWindows()
```



**Bir resimde piksel piksel renk aralığı seçme (for döngüsü)**

```
import cv2  
import numpy as np  
from matplotlib import pyplot as plt  
sayac=0  
x=cv2.imread('balon.jpg')  
plt.subplot(121)  
plt.imshow(x)  
for i in range(x.shape[0]):  
    for j in range(x.shape[1]):  
        if x[i,j,0]>100 and x[i,j,1]<200 and x[i,j,2]<200:  
            x[i,j,:]=0  
            sayac=sayac+1  
plt.subplot(122)  
plt.imshow(x)
```



Yukarıda, belirlenen aralıkta B, G ve R parlaklık değerine sahip olan pikseller siyah yapıldı.

### Resim üzerine şekiller çizme

Bir resim üzerine dörtgen, çizgi ve daire çizmek.

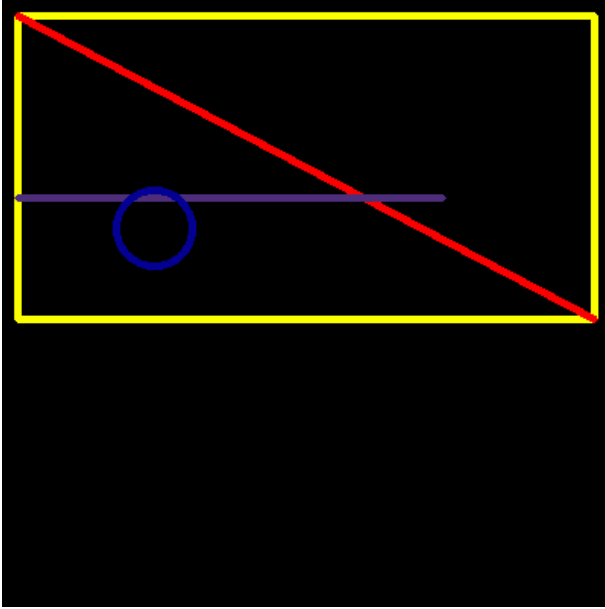
```
import cv2
import numpy as np
resim=np.zeros((400,400,3),dtype='uint8')
cv2.rectangle(resim,(10,10),(390,210),(0,255,251),3)
cv2.line(resim,(10,10),(390,210),(0,0,251),3)
cv2.line(resim,(10,230),(390,230),(123,45,78),3)
cv2.circle(resim,(200,350),25,(148,0,4),3)
cv2.circle(resim,(200,201),50,(244,135,244),-1)#içi dolu şekil
cv2.imshow('siyah',resim)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

Açıklamalı [GS1]: Başlangıç koordinatları

Açıklamalı [GS2]: Bitiş koordinatları

Açıklamalı [GS3]: renk kodu

Açıklamalı [GS4]: Kalınlık



## Aritmetik ve Mantıksal işlemler

### İki resim ve maske arasında mantıksal işlemler

İki resim seçelim ve bir de siyah-beyaz maske oluşturalım.

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
x=cv2.imread('balonlar2.jpg')
y=cv2.imread('anahtar.jpg')

print("x boyutu:"+str(x.shape))
print("y boyutu:"+str(y.shape))
#iki resmi aynı boyutlara getirmeliyiz.
y=cv2.resize(y, (226,223))
print("y boyutu:"+str(y.shape))
```

*#Bir maske oluşturalım.*

```
maske=np.zeros((223,226,3), np.uint8)  
maske[50:150,50:150]=255
```

*#MANTIKSAL İŞLEMLER*

```
xANDmaske=cv2.bitwise_and(x,maske)  
xORmaske=cv2.bitwise_or(x,maske)  
xANDy=cv2.bitwise_and(x,y)  
xORy=cv2.bitwise_or(x,y)  
x_NOT=cv2.bitwise_not(x)  
y_NOT=cv2.bitwise_not(y)  
maske_NOT=cv2.bitwise_not(maske)
```

```
plt.figure(1)  
plt.subplot(331)  
plt.imshow(x)  
plt.subplot(332)  
plt.imshow(y)  
plt.subplot(333)  
plt.imshow(xANDmaske)  
plt.subplot(334)  
plt.imshow(xORmaske)  
plt.subplot(335)  
plt.imshow(xANDy)  
plt.subplot(336)  
plt.imshow(xORy)  
plt.subplot(337)  
plt.imshow(x_NOT),plt.title('x Not')  
plt.subplot(338)  
plt.imshow(y_NOT),plt.title('y Not')
```

```
plt.figure(2)  
plt.subplot(221)  
plt.imshow(maske),plt.title('maske')  
plt.subplot(222)  
plt.imshow(maske_NOT),plt.title('maske Not')
```

