

SORU :

Hareket sırasındaki yuvarlak yansıtıcıları bulan bir program yazınız.

YANIT:

Gürültüyü azaltmak için bwareafilt komutu ile büyük nesneleri resimden çıkartan program

```
RGB = imread('C:\nesneler_0001.jpg'); %Resim oku
I = rgb2gray(RGB); %gray e çevir
threshold = graythresh(I); %threshold degerini al
bw = im2bw(I,threshold); %binary yap
bw = bwareaopen(bw,50); %50 pixelden küçük nesneleri sil
bw2 = bwareafilt(bw,[70 200]); %70-200 arası nesneleri sil
imshowpair(bw,bw2,'montage'); %yanyana ekrana yazdır
```

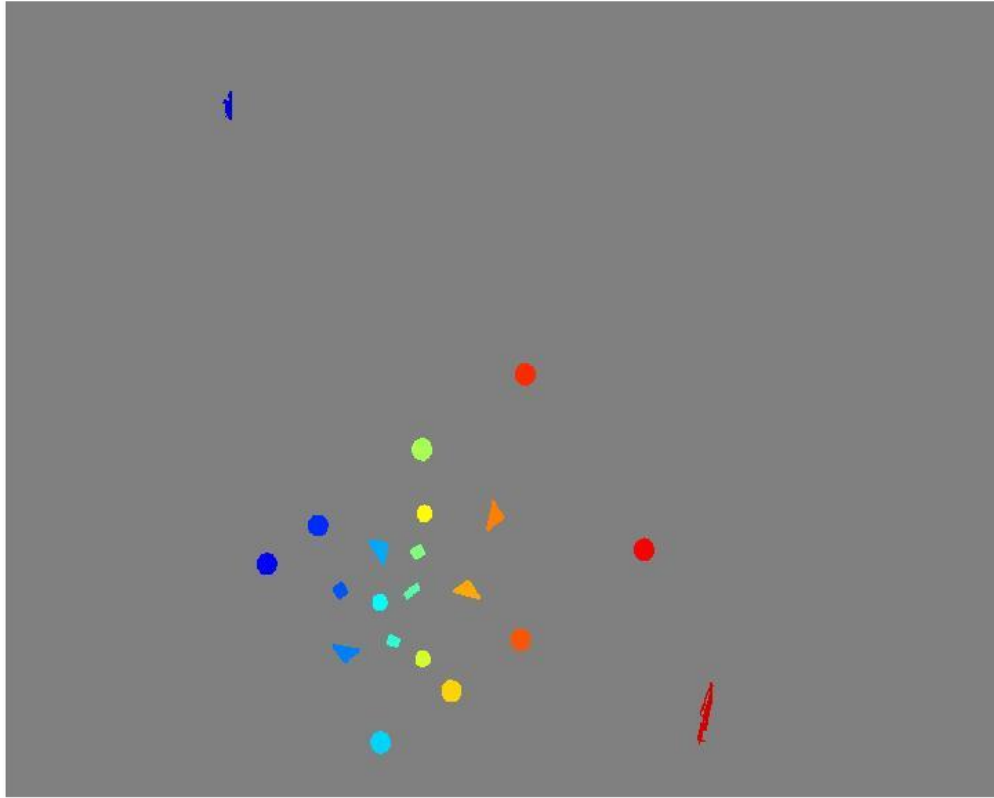


Gürültünün azaldığı yerler yeşille gösterilmiştir.

Nesneleri renk paleti ile renklendiren ve arka planı gri yapan program

```
RGB = imread('C:\nesneler_0001.jpg');
I = rgb2gray(RGB);
threshold = graythresh(I);
bw = im2bw(I,threshold);
bw = bwareaopen(bw,50);
bw = bwareafilt(bw,[70 200]);
```

```
[B,L]=bwboundaries(bw,'noholes'); %nesneleri bulup matris te
sayı vererek işaretliyor.
LRGB=label2rgb(L, @jet, [.5 .5 .5]); %bulunan nesneleri renk
paleti kullanarak etiketliyor.
figure,imshow(LRGB) %ekrana yazdır
```



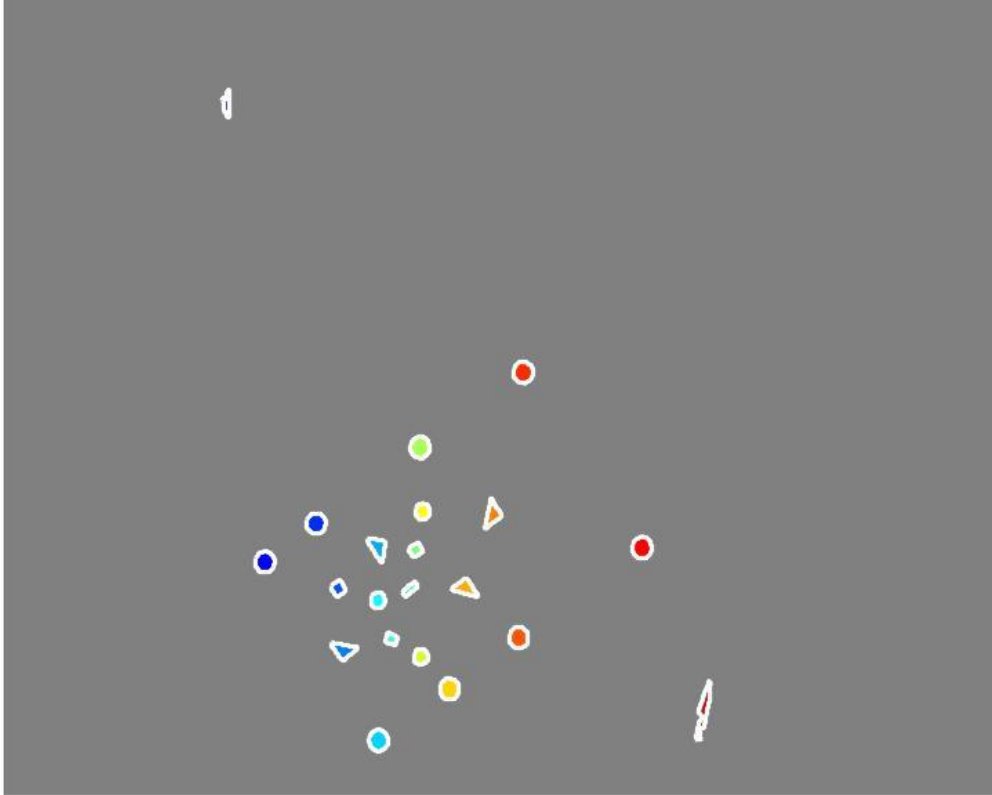
Nesnelerin kenarlarını çizdiren program

```
RGB = imread('C:\nesneler_0001.jpg');
I = rgb2gray(RGB);
threshold = graythresh(I);
bw = im2bw(I,threshold);
bw = bwareaopen(bw,50);
bw = bwareafilt(bw,[70 200]);
[B,L]=bwboundaries(bw,'noholes');
LRGB=label2rgb(L, @jet, [.5 .5 .5]);
figure,imshow(LRGB)
hold on %önceki resmi sabit tut
for k = 1:length(B)
```

```

boundary = B{k};
plot(boundary(:,2),boundary(:,1),'w', 'LineWidth',2) %beyaz
'w' ve 2 çizgi kalınlığı ile sınırları çiz
end

```



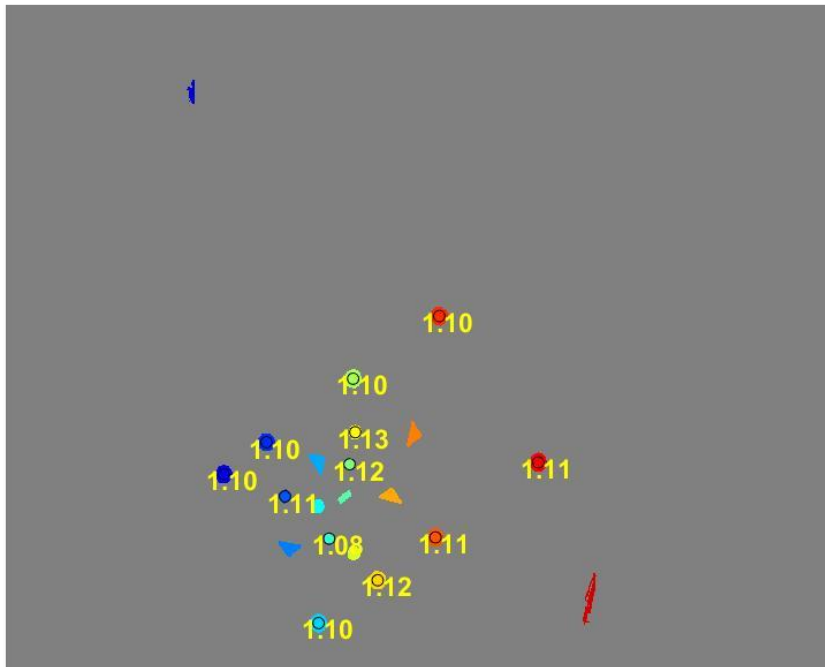
Yuvarlakları işaretleyen ve yanlarına yuvarlaklık katsayılarını yazan program.

```

RGB = imread('C:\nesneler_0001.jpg');
I = rgb2gray(RGB);
threshold = graythresh(I);
bw = im2bw(I,threshold);
bw = bwareaopen(bw,50);
bw = bwareafilt(bw,[70 200]);
[B,L]=bwboundaries(bw,'noholes');
LRGB=label2rgb(L, @jet, [.5 .5 .5]);
figure,imshow(LRGB)
hold on

```

```
ratioLow = 0.85; %yuvarlaklık katsayısı 0.85 den az
olmayacak
ratioUp = 1.15; %yuvarlaklık katsayısı 1.15 den fazla
olmayacak
for k = 1:length(B)
boundary = B{k};
stats = regionprops(L, 'Area', 'Centroid', 'Perimeter');
%nesne ile ilgili parametreleri alıyoruz.
perimeter = stats(k).Perimeter; %alan parametresini
degiskene aktarıyor
area = stats(k).Area; %çevre parametresini degiskene
aktarıyor
ratio = 4*pi*area / perimeter^2; %yuvarlaklık katsayısını
hesaplıyor
ratio_string = sprintf('%2.2f', ratio); %yuvarlaklık
katsayısını istediğimiz sayı formatında değışkene yazıyor
if (ratio >= ratioLow) && (ratio <= ratioUp) %koşul
centroid = stats(k).Centroid; %orta noktasını buluyor
plot(centroid(1), centroid(2), 'ko'); %orta noktaya işaret
text(centroid(1) - 15, centroid(2) +
5, ratio_string, 'Color', 'y', 'FontSize', 14, 'FontWeight', 'bold
'); %orta noktanın 15 birim kenarına yuvarlaklık katsayı
değerini yazıyor
end
end
```



Tam program

```

for i=1:21
string = sprintf('C:\\nesneler_000%d.jpg', i);
RGB = imread(string);
I = rgb2gray(RGB);
threshold = graythresh(I);
bw = im2bw(I,threshold);
bw = bwareaopen(bw,50);
bw = bwareafilt(bw,[70 200]);
[B,L]=bwboundaries(bw,'noholes');
LRGB=label2rgb(L, @jet, [.5 .5 .5]);
figure,imshow(LRGB)
hold on
ratioLow = 0.85;
ratioUp = 1.15;
for k = 1:length(B)
boundary = B{k};
stats = regionprops(L,'Area','Centroid','Perimeter');
perimeter = stats(k).Perimeter;
area = stats(k).Area;
ratio = 4*pi*area / perimeter^2;

```

```
ratio_string = sprintf('%2.2f', ratio);
if (ratio >= ratioLow) && (ratio <= ratioUp)
centroid = stats(k).Centroid;
    plot(centroid(1), centroid(2), 'ko');
    text(centroid(1) - 15, centroid(2) +
5, ratio_string, 'Color', 'y', 'FontSize', 14, 'FontWeight', 'bold
');
end
end
end
```

* Takip değerleri artış düzeni

1	2
3	4

Takip değerleri [1-21 arası]

