



GÖRÜNTÜ İŞLEME DERS-6 YARDIMCI NOTLARI -2018

İMGE MANTIKSAL FONKSİYONLARI

İmge matrisleri arasında aritmetik işlemler yapılabildiği gibi

Mantıksal işlemler de yapılabilmektedir. Bu mantıksal işlemler piksel parlaklık değerleri arasında bit – bit karşılığında yapılır. Örneğin parlaklık değeri 255 olan bir pikselin 8 bit karşılığı "1 1 1 1 1 1 1 1" dir. Diğer pikselin parlaklık değeri 3 ise bunun da 8 bit karşılığı "0 0 0 0 0 0 1 1" dir. İkisini AND işlemine tabi tutarsak sonucu 3'dür. OR işlemine tabi tutarsak sonucu 255'dir.

Bu sonuçlar ışığında mantıksal işlemler genellikle bir görüntünün maskelenmesi amacıyla kullanılırlar.

Ancak mantıksal işlemlerde uyulması gereken bir takım kurallar vardır. Öncelikle iki görüntünün boyutları birbirine eşit olmalıdır. Ayrıca mantıksal işlem yapılacak piksel parlaklık değerlerinin 8 bit integer yani uint8 olması gerekir.

AND, NAND

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

AND

A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

NAND

Aritmetik ve mantıksal fonksiyonlar kullanılarak yapılan işlemler:

```
% lojik operatörler
clc;
clear all;
close all;
A=imread('cameraman.tif');
subplot(5,3,1)
imshow(A)
title('Image A');
B=imread('rice.png');
subplot(5,3,2)
imshow(B)
title ('Image B');
```

```
%Aritmetik operatörler
clc;
clear all;
close all;
A=imread('cameraman.tif');
B=imread('rice.png');
C=zeros(size(A));
for (x=100:200)
    for (y=100:200)
        C(x,y)=255;
```



AKÜ TEKNOLOJİ FAKÜLTESİ
MEKATRONİK MÜHENDİSLİĞİ BÖLÜMÜ



```
C=zeros(size(A));
for (x= 100: 200)
    for (y=100: 200)
        C(x,y)=255;
    end
end
subplot(5,3,3)
imshow(C)
title('Image C');
C=uint8(C);

lr1=bitand(A,C);
subplot(5,3,4)
imshow(lr1)
title('A and C');
lr2=bitand(B,C);
subplot(5,3,5)
imshow(lr2)
title('B and C');
lr3=bitand(A,B);
subplot(5,3,6)
imshow(lr3)
title('A and B');
lr4=bitor(A,C);
subplot(5,3,7)
imshow(lr4)
title('A or C');
lr5=bitor(B,C);
subplot(5,3,8)
imshow(lr5)
title('B or C');
lr6=bitor(A,B);
subplot(5,3,9)
imshow(lr6)
title('A or B');
lr7=bitxor(A,C);
subplot(5,3,10)
imshow(lr7)
title('A exor C');
lr8=bitxor(B,C);
subplot(5,3,11)
imshow(lr8)
title('B exor C');
lr9=bitxor(A,B);
subplot(5,3,12)
imshow(lr9)
title('A exor B');
lr10= bitcmp(A);
subplot(5,3,13)
imshow(lr10)
title('Not A');
lr11= bitcmp(B);
subplot(5,3,14)
imshow(lr11)
title('Not B');
```

```
end
end
C=uint8(C);

lr13=imadd(A,C);
figure,subplot(4,3,1)
imshow(uint8(lr13))
title('A + C');
lr14=imadd(B,C);
subplot(4,3,2)
imshow(uint8(lr14))
title('B + C');
lr15=imadd(A,B);
subplot(4,3,3)
imshow(uint8(lr15))
title('A + B');
lr16=imsubtract(A,C);
subplot(4,3,4)
imshow(uint8(lr16))
title('A - C');
lr17=imsubtract(B,C);
subplot(4,3,5)
imshow(uint8(lr17))
title('B - C');
lr18=imsubtract(A,B);
subplot(4,3,6)
imshow(uint8(lr18))
title('A - B');
lr19=immultiply(A,C);
subplot(4,3,7)
imshow(uint8(lr19))
title('A * C');
lr20=immultiply(B,C);
subplot(4,3,8)
imshow(uint8(lr20))
title('B * C');
lr21=immultiply(A,B);
subplot(4,3,9)
imshow(uint8(lr21))
title('A * B');
lr22=imdivide(A,C);
subplot(4,3,10)
imshow(uint8(lr22))
title('A/C');
lr23=imdivide(B,C);
subplot(4,3,11)
imshow(uint8(lr23))
title('B/C');
lr24=imdivide(A,B);
subplot(4,3,12)
imshow(uint8(lr24))
title('A/B');
```

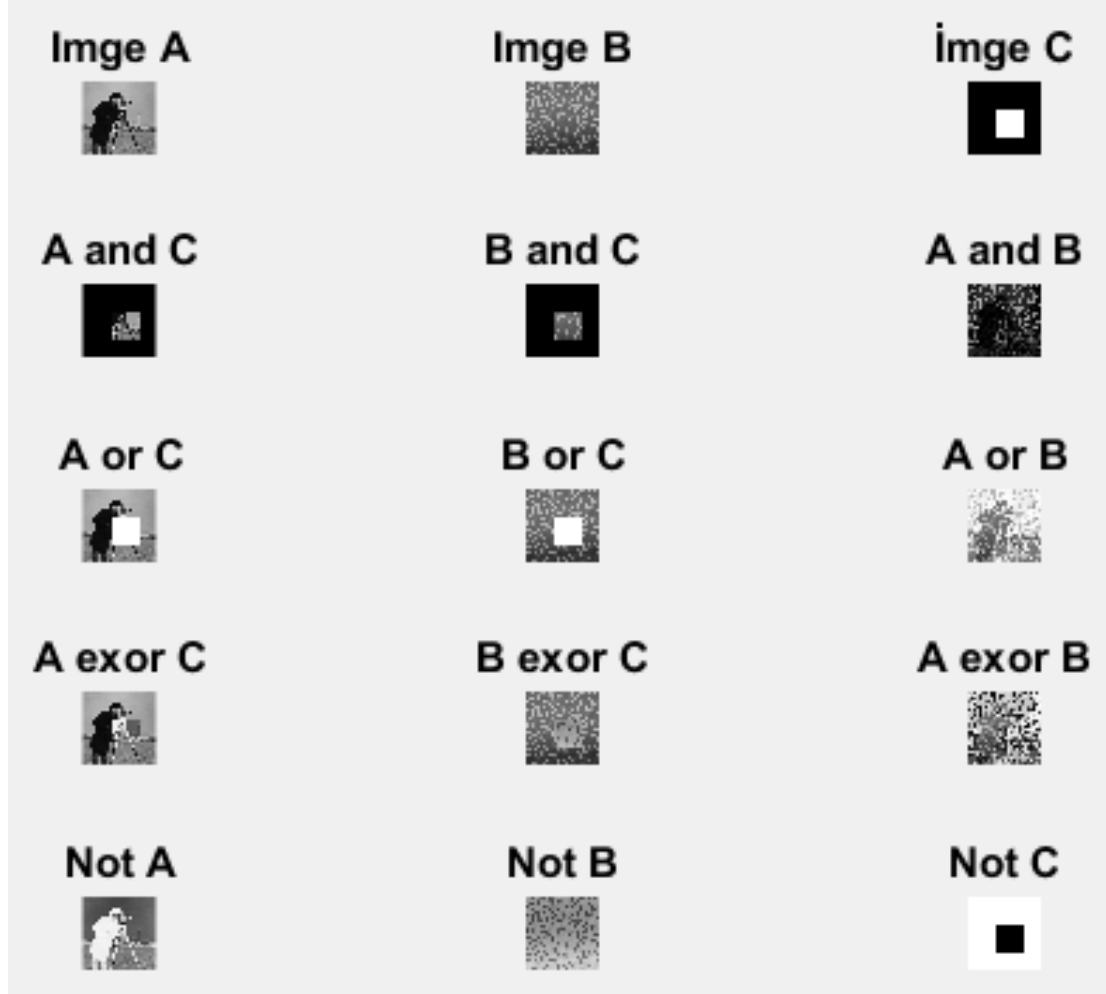


AKÜ TEKNOLOJİ FAKÜLTESİ
MEKATRONİK MÜHENDİSLİĞİ BÖLÜMÜ



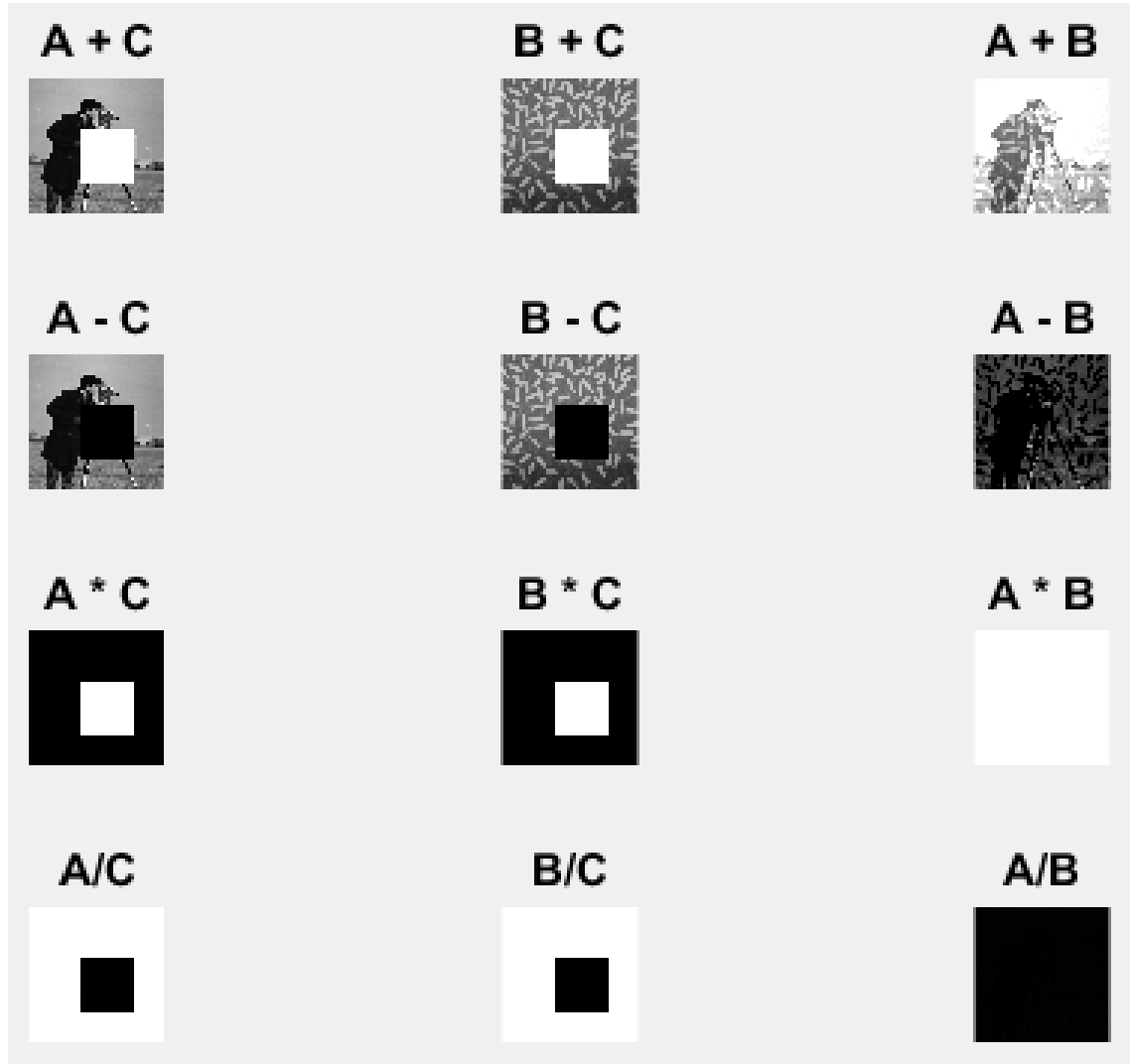
```
lr12= bitcmp(C);  
subplot(5,3,15)  
imshow(lr12)  
title('Not C');
```

Mantıksal işlemlerin sonuçları:





Aritmetik işlemlerin Sonuçları:



Mantıksal ve Aritmetik işlemler arasındaki benzerlikleri araştırabilirsiniz.

İMGE BOYUTLARININ DEĞİŞTİRİLMESİ

Aşağıdaki komutlar ile imge boyutları değiştirilebilir.

imresize() komutu

```
>> A=imread('cameraman.tif');  
>> B=imresize(A,2);  
>> imshow(B)  
>> B=imresize(A,[100,133]); satır 100 sütun 133 'e döndürür.
```



```
>> imshow(B)
>> B=imresize(A,[133 NaN]);%sadır sayısı 133 sütun sayısı ölçekli
>> imshow(B)
```

VİZE HAZIRLIK ÇALIŞMA SORULARI

1.) Kalemler.jpg adında RGB formatında bir resim dosyası veriliyor.

a) Bu resim dosyasının R ve B renk kanalında 80 ile 120 parlaklık değerleri arasında olan piksellerin sayısını bulunuz ve ayrı ayrı ekrana yazdırınız.

b) Bu resim dosyasının G renk kanalında 60 ile 90 parlaklık değerleri arasında bulunan piksellerin tümleyenlerini (complement) alarak aynı resim üzerinde gösteriniz ve resmi ekrana görüntüleyiniz.

Yanıt 1.)

```
clear;close all;clc;
x=imread('kalemler.jpg');
figure,imshow(x);
R_sayi=0;G_sayi=0;B_sayi=0;
%%
% a şıkkı
for i=1:size(x,1)
    for j=1:size(x,2)
        if x(i,j,1)>=80 && x(i,j,1)<=120
            R_sayi=R_sayi+1;
        end
    end
end

for i=1:size(x,1)
    for j=1:size(x,2)
        if x(i,j,2)>=80 && x(i,j,2)<=120
            G_sayi=G_sayi+1;
        end
    end
end

for i=1:size(x,1)
    for j=1:size(x,2)
        if x(i,j,3)>=80 && x(i,j,3)<=120
            B_sayi=B_sayi+1;
        end
    end
end
```



AKÜ TEKNOLOJİ FAKÜLTESİ
MEKATRONİK MÜHENDİSLİĞİ BÖLÜMÜ



```
end
end
end
fprintf('R=%d G=%d B=%d\n',R_sayi,G_sayi,B_sayi);

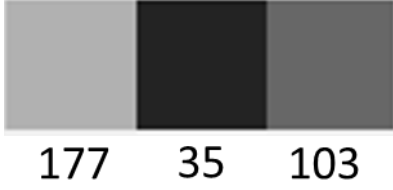
%%
%b şıkkı
% her Bir kanal (R,G,B) için değeri [60 90] aralığında olan
piksellerin
% tümleyenleri alınıyor.
for i=1:size(x,1)
    for j=1:size(x,2)
        if x(i,j,1)>=60 && x(i,j,1)<=90
            x(i,j,1)=255-x(i,j,1);
        end
    end
end

for i=1:size(x,1)
    for j=1:size(x,2)
        if x(i,j,2)>=60 && x(i,j,2)<=90
            x(i,j,2)=255-x(i,j,2);
        end
    end
end

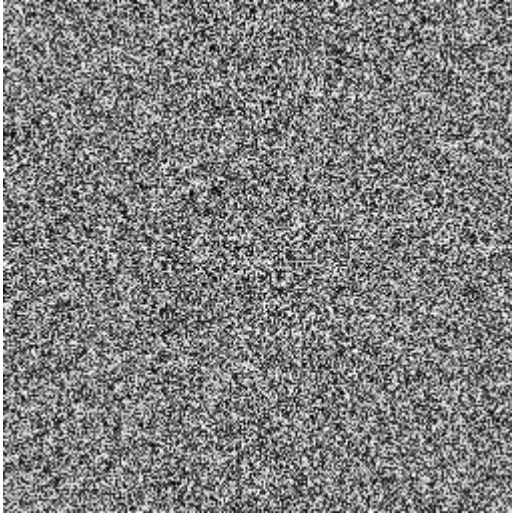
for i=1:size(x,1)
    for j=1:size(x,2)
        if x(i,j,3)>=60 && x(i,j,3)<=90
            x(i,j,3)=255-x(i,j,3);
        end
    end
end
figure,imshow(x);
```

2.)rastgele.jpg resminde aşağıda verilen örüntünün kaç defa geçtiğini ve başlangıç koordinatlarını bulan bir program yazınız.

Bulunacak Örüntü:



Rastgele.jpg



Yanıt 2.)

```
clear;close all;clc;
x=imread('rastgele.jpg');
figure,imshow(x);
sayac=0;k=1;
for i=1:size(x,1)
    for j=1:size(x,2)-2
        if x(i,j)==177 && x(i,j+1)==35 && x(i,j+2)==103
            sayac=sayac+1;
            baslangicKoord(k,1)=i;
            baslangicKoord(k,2)=j;
            k=k+1;
        end
    end
end
end
fprintf('Bulunan örüntü sayısı=%d\n',sayac);
fprintf('Koordinatlar: Satır=%d
Sütun=%d\n',baslangicKoord(1,1),baslangicKoord(1,2));
```



Bulunan örüntü sayısı=1

Koordinatlar: Satır=11 Sütun=7

3.) RGB bir resimdeki kırmızı renkli objeleri bulmak için resmin kırmızı kanalından resmin gri seviyesini çıkartmak gerekir. Buna göre kalemler.jpg resmindeki kırmızı renkli objeleri aşağıdaki gibi siyah-beyaz resim olarak bulan programı yazınız. (Not: eşik değeri uint8 50 ya da double 0.1 alınız.)

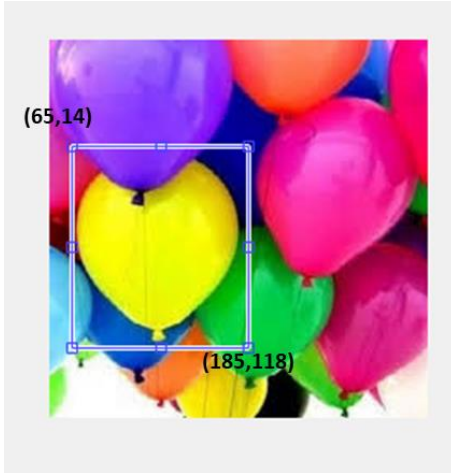


Yanıt 3.)

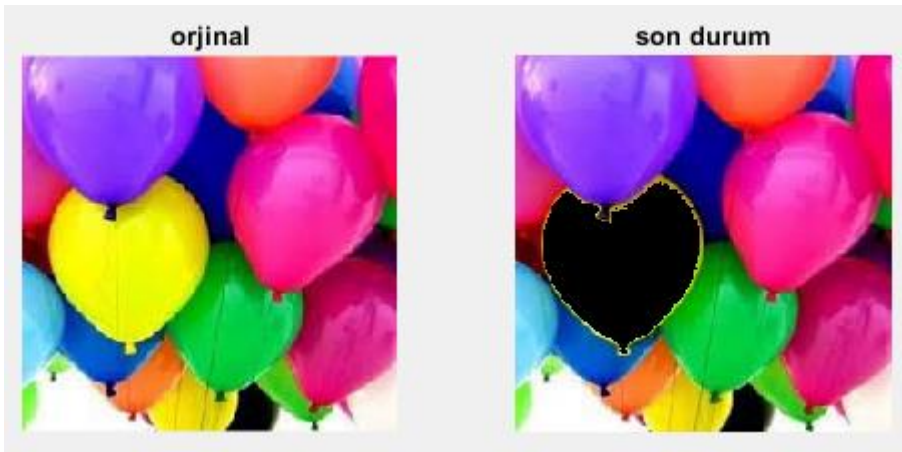
```
a=imread('kalemler.jpg');  
r=a(:,:,1);%kırmızı bant ayrıldı  
gri=rgb2gray(a);%resim gri seviye yapıldı  
KirmiziObjeler=imsubtract(r,gri); %kırmızı banttan gri seviye  
çıkartılarak sadece kırmızı olanlar bulunuyor.  
BW=KirmiziObjeler>50;  
subplot(1,2,1);imshow(a);%resimler görüntüleniyor.  
subplot(1,2,2);imshow(BW);%resimler görüntüleniyor.
```

4.) Aşağıdaki resimde verilen alan içerisinde kalan sarı balonu siyaha boyayacak olan programı iç içe for döngüleri kullanarak yazınız. Sarı renk için şart $R>180$; $G>150$; $20<B<100$ 'dür. Resmin ilk ve son halini aşağıdaki şekilde ekrana getiriniz. (25p)

Not: Verilen koordinatlar satır ve sütun cinsindedir.



Olması gereken son durum:



Yanıt 4.)

```
x=imread('balon.jpg');  
subplot(1,2,1);imshow(x);title('orjinal')  
for i=1:size(x,1)  
    for j=1:size(x,2)  
        if (i>=65 && i<=185) && (j>=14 && j<=118)  
            if x(i,j,1)>180 && x(i,j,2)>150 && x(i,j,3)>=0 &&  
x(i,j,3)<=100  
                x(i,j,:)=0;  
            end  
        end  
    end  
end  
subplot(1,2,2);imshow(x);title('son durum')
```



**AKÜ TEKNOLOJİ FAKÜLTESİ
MEKATRONİK MÜHENDİSLİĞİ BÖLÜMÜ**

