



GÖRÜNTÜ İŞLEME DERS-5 YARDIMCI NOTLARI -2018

İmge Tipleri

1. rgb
2. gri seviye
3. binary (ikili)
4. indeksli

Görüntü işlemede kullanılan sınıflar

Name	Description
double	Double-precision, floating-point numbers in the approximate range $\pm 10^{308}$ (8 bytes per element).
single	Single-precision floating-point numbers with values in the approximate range $\pm 10^{38}$ (4 bytes per element).
uint8	Unsigned 8-bit integers in the range [0, 255] (1 byte per element).
uint16	Unsigned 16-bit integers in the range [0, 65535] (2 bytes per element).
uint32	Unsigned 32-bit integers in the range [0, 4294967295] (4 bytes per element).
int8	Signed 8-bit integers in the range [-128, 127] (1 byte per element).
int16	Signed 16-bit integers in the range [-32768, 32767] (2 bytes per element).
int32	Signed 32-bit integers in the range [-2147483648, 2147483647] (4 bytes per element).
char	Characters (2 bytes per element).
logical	Values are 0 or 1 (1 byte per element).

Uint8 ve logical sınıfları görüntü işleme süreçlerinde yaygın olarak kullanılmaktadır. Okunan resim formatları TIFF, png veya JPEG vb. olabilmektedir.

Tıbbi görüntüler vb. bazı bilimsel veri kaynaklarından daha hassas okuma yapabilmek amacıyla uint16 ya da int16 sınıfları kullanılabilir.

İMGE SINIFLARI VE SINIFLAR ARASI DÖNÜŞÜM

Genel olarak;
B=sinif_adi(A); şeklinde yapılır.

Name	Converts Input to:	Valid Input Image Data Classes
im2uint8	uint8	logical, uint8, uint16, int16, single, and double
im2uint16	uint16	logical, uint8, uint16, int16, single, and double
im2double	double	logical, uint8, uint16, int16, single, and double
im2single	single	logical, uint8, uint16, int16, single, and double
mat2gray	double in the range [0, 1]	logical, uint8, int8, uint16, int16, uint32, int32, single, and double
im2bw	logical	uint8, uint16, int16, single, and double

Örnek:

`%double veri tipinde bir matris oluşturalım.`

```
>> a=[255 89 157;62 50 121;237 64 0];
```

```
    255    89    157
     62    50    121
    237    64     0
```

```
>>class(a) %matrisin sınıfını (veri tipini) öğreniyoruz.
```

```
ans =
```

```
double
```

```
>> figure, imshow(a)
```



Görüldüğü üzere double class'ında sayıların herhangi bir renk karşılığı yok.

```
>> b=uint8(a)
```

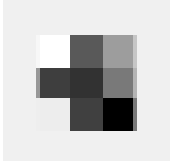
```
b =
```

```
3x3 uint8 matrix
```

```
    255    89    157
     62    50    121
    237    64     0
```

Dönüştürme işlemi

```
>> figure, imshow(b)
```



Görüldüğü üzere uint8 class'ında sayıların renk karşılıkları vardır.

```
>> a
```

```
a =
```

```
255    89   157
 62    50   121
237    64     0
```

```
% Mat2gray() fonksiyonu matris elemanlarını [0 .. 1] aralığına ölçekler.
```

```
>> d=mat2gray(a)
```

```
d =
```

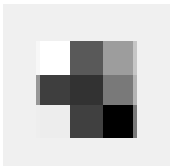
```
1.0000    0.3490    0.6157
0.2431    0.1961    0.4745
0.9294    0.2510         0
```

```
>> class(d)
```

```
ans =
```

```
'double'
```

```
>> figure, imshow(d)
```



[0 .. 1] aralığındaki double sayıların renk karşılıkları vardır.

```
 %[0 .. 1] aralığına ölçeklenen matris elemanları im2uint8() fonksiyonu ile [0 .. 255] arasına ölçekli hale getirilir.
```

```
>> e=im2uint8(d) %double f matrisi uint8 sınıfına çeviriyoruz.
```

```
e =
```

```
3×3 uint8 matrix
```

```
255    89    157
   62    50    121
  237    64     0
```

```
>> class(e) %yeni matrisin sınıfının da uint8 olduğunu görüyoruz.
```

```
ans =
```

```
uint8
```

```
>> A=[-2 2;0 4]
```

```
A =
```

```
 -2     2
   0     4
```

```
>> I = mat2gray(A)
```

```
I =
```

```
     0    0.6667
0.3333    1.0000
```

BİNARY (İKİLİ GÖRÜNTÜLER)

```
x=imread('headquarters-2.jpg');
subplot(1,2,1);
imshow(x)
bw=im2bw(x); %imge 8 bitlik sistemden 1 bitlik sisteme aktarılıyor.
subplot(1,2,2);
imshow(bw)
```



Burada `im2bw()` fonksiyonuna eşik değeri girilmedi. Bu tür durumlarda Matlab kendisi bir eşik değeri oluşturur.

%Eşik değeri verilerek belirli bir değere kadar piksel parlaklık değerleri 0 alınır, belirli bir değerden sonra 1 alınır.

```
>>B =  
    0         0.6667  
    0.3333    1.0000
```

```
>> A=im2bw(B,0.6)
```

Eşik değeri: 0.6

```
A =  
    0     1  
    0     1
```

Farklı bir yöntem ile resimlerin ikilik resime dönüştürülmesi:

```
>> B=[1 2;3 4]
```

```
B =  
    1     2  
    3     4
```

```
>> A=B>2 %2 'den büyük olan değerler 1 küçükler 0 değerine atanır.
```

```
A =  
    0     0  
    1     1
```

%Matrislere ekrandan bakıldığında eşitliğin sağında bulunan şartı doğrulayan kümenin "1" yapıldığını görürüz.

```
a=imread('cameraman.tif');  
subplot(1,2,1);  
imshow(a);  
b=a<50;  
subplot(1,2,2);  
imshow(b);
```



%Yukarıdaki örnekte uint8 50 olarak seçilen eşik değeri 130 olarak seçip örneği tekrarlayalım:

```
a=imread('cameraman.tif');  
subplot(1,2,1);  
imshow(a);  
b=a<130;  
subplot(1,2,2);  
imshow(b);
```



Görüldüğü üzere ikilik görüntüleri elde ederken verilen eşik değerin önemi büyüktür. Siluetin elde edilmesi için eşik değerin doğru olarak belirlenmesi gerekir. Yukarıdaki örnekte verilen değer daha doğru görünmektedir.

ÇALIŞMA ÖRNEĞİ

1. Kendi kodlarımızla bir RGB imgeyi gri seviye imge haline getirecek fonksiyonu yazınız.

```
function griMatris=griYap(rgbResim)
if size(rgbResim,3)>1 %eğer girişe RGB olmayan bir resim gelirse
    for i=1:size(rgbResim,1) %rgbResim satırlarını tarar
        for j=1:size(rgbResim,2) %rgbResim sütunlarını tarar
            griMatris(i,j)=(rgbResim(i,j,1)+rgbResim(i,j,2)+rgbResim(i,j,3))./3;
        end
    end
else
    griMatris=rgbResim;%RGB olmayan bir resim ise değiştirmeden dışarı aktar
end
end
```

2. Kendi kodlarımızla bir RGB imgeyi ikilik imge haline getirecek fonksiyonu yazınız.

```
function bw=bwYap(resim,esik)
for i=1:size(resim,1) %resim satırlarını tarar
    for j=1:size(resim,2) %resim sütunlarını tarar
        if resim(i,j)>esik %resim matrisinin her bir elemanını eşik değer
            ile karşılaştırır.
                bw(i,j)=1;
            else
                bw(i,j)=0;
            end
        end
    end
end
end
```

%Fonksiyonu deneyelim: (Command Window)

```
>>x=imread('headquarters.jpg');
>>y=griYap(x);
>>bwYap(y,50);
```

ÇALIŞMA ÖRNEĞİ:

3. Bir resmin sağa-sola ya da yukarı-aşağı ötelenmesini sağlayacak kodları yazınız.

Bir imgeyi x ya da y eksenleri boyunca kaydırma işlemleri for döngüleri ile yapılabilir. Aşağıdaki örnekte 15 piksel x yönünde kaydırma yapılmıştır.

```
x=imread('headquarters-2.jpg');
x=rgb2gray(x); %önce resim gri seviyeye dönüştürülüyor.
y=uint8(zeros(size(x,1),size(x,2)));%yeni resim için boş bir matris
açılıyor. Bu yapılmasa da olur
ancak yapıldığında program hızlı
çalışacaktır.
```

```
for i=1:size(x,1)
    for j=1:size(x,2)-15
        y(i,j+15)=x(i,j);
    end
end
%Görüntüleme başlıyor.
subplot(1,2,1);
imshow(x);
title('orjinal görüntü');
subplot(1,2,2);
imshow(y);
title('ötelenmiş görüntü')
```

Sütun tarama değeri sınır değeri aşmasın diye 15 piksel azaltılıyor.

Resim 15 piksel sağa doğru öteleniyor.



İMGE ARİTMETİK FONKSİYONLARI

Function	Description
imadd	Adds two images; or adds a constant to an image.
imsubtract	Subtracts two images; or subtracts a constant from an image.
immultiply	Multiplies two image, where the multiplication is carried out between pairs of corresponding image elements; or multiplies a constant times an image.
imdivide	Divides two images, where the division is carried out between pairs of corresponding image elements; or divides an image by a constant.
imabsdiff	Computes the absolute difference between two images.
imcomplement	Complements an image.

ÇALIŞMA ÖRNEĞİ

4.) İki RGB resmin Aritmetik işlemlerini gerçekleştiriniz.

```
x=imread('headquarters.jpg');
y=imread('headquarters-Defect.jpg');
subplot(2,3,1);
imshow(x);
title('Orjinal Resim1');
subplot(2,3,2);
imshow(y);
```

```

title('Orjinal Resim2')
%*****
a=imsubtract(x,y);
subplot(2,3,3);
imshow(a);
title('Çıkartma İşlemi');
%*****
b=imadd(x,y);
subplot(2,3,4);
imshow(b);
title('Toplama İşlemi');
%*****
c=immultiply(x,y);
subplot(2,3,5);
imshow(c);
title('Çarpma İşlemi');
%*****
d=imdivide(x,y);
subplot(2,3,6);
imshow(d);
title('Bölme İşlemi');

```

İki resim arasındaki aritmetik işlemler sadece `imsubtract()`, `imadd()` gibi fonksiyonlarla değil `+`, `-`, `*`, `/` gibi operatörlerle de yapılabilir.



Çıkartma işleminde iki resim arasındaki fark olan pencerenin ortaya çıktığına dikkat ediniz.

Aritmetik işlemlere ek olarak bir resmin tümleyenini alma işlemi:

```

x=imread('headquarters.jpg');
tumleyen=255-x;%veya tumleyen=imcomplement(x) kullanılabilir.
figure, imshow(tumleyen);

```



