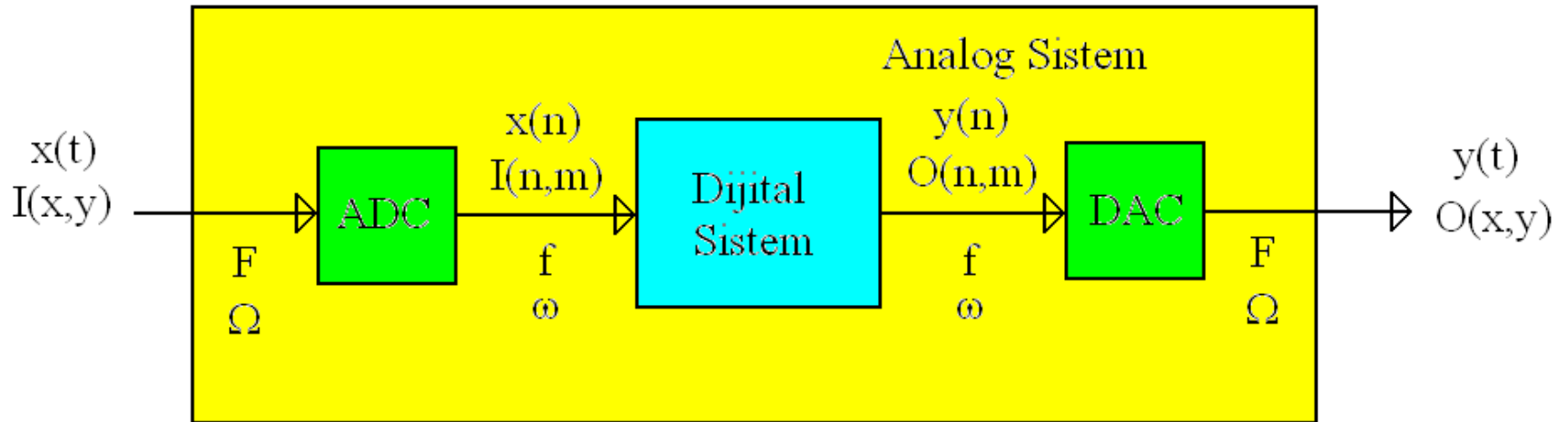


BMÜ-357 Sayısal Görüntü İşleme

MATLAB İLE GÖRÜNTÜ İŞLEME

Yrd. Doç. Dr. İlhan AYDIN

Sayısal Görüntü İşleme: Sensörlerden gelen görüntünün bilgisayara aktarılıp üzerinde herhangi bir işlem yapılması ve ardından görüntüleyici çıkışa iletilmesi



Görüntü işlemede temel kavramlar

- ▶ **Piksel (pixel)** : picture element sözcüklerinin birleştirilmesiyle oluşmuştur, görüntünün birim elemanını ifade eder.
- ▶ **Parlaklık (intensity)**: x ve y uzaysal boyutlar olmak üzere $I(x,y)$, x ve y koordinatlarındaki pikselin parlaklık değerini gösterir.
- ▶ **Ayrıklaştırma (Digitizing)**: Analog görüntünün sayısal sistemde ifade edilebilmesi için önce uzaysal boyutlarda sonlu sayıda ayırık parçaya bölünmesi (**örnekleme, sampling**), sonra da her bir parçadaki analog parlaklık değerinin belli sayıda ayırık sayısal seviyelerden biri ile ifade edilmesi (**kuantalama, quantizing**) gerekir.
- ▶ **Çözünürlük (Resolution)**: görüntünün kaç piksele bölündüğünü, yani kaç pikselle temsil edildiğini gösterir. Çözünürlük ne kadar yüksekse, görüntü o kadar yüksek frekansta örneklenmiş olur ve görüntüdeki ayrıntılar o kadar belirginleşir.
- ▶ **Uzaysal Frekanslar (Spatial Frequencies)**: Uzaysal boyutlarda belli bir mesafede parlaklık değerinin değişim sıklığını ifade ederler.

Analog ve Sayısal Görüntü

- Analog bir görüntü ve bu görüntünün örneklenmesi ile elde edilen sayısal görüntünün matrissel içeriği aşağıdaki şekilde verilmiştir.



Gerçek resim



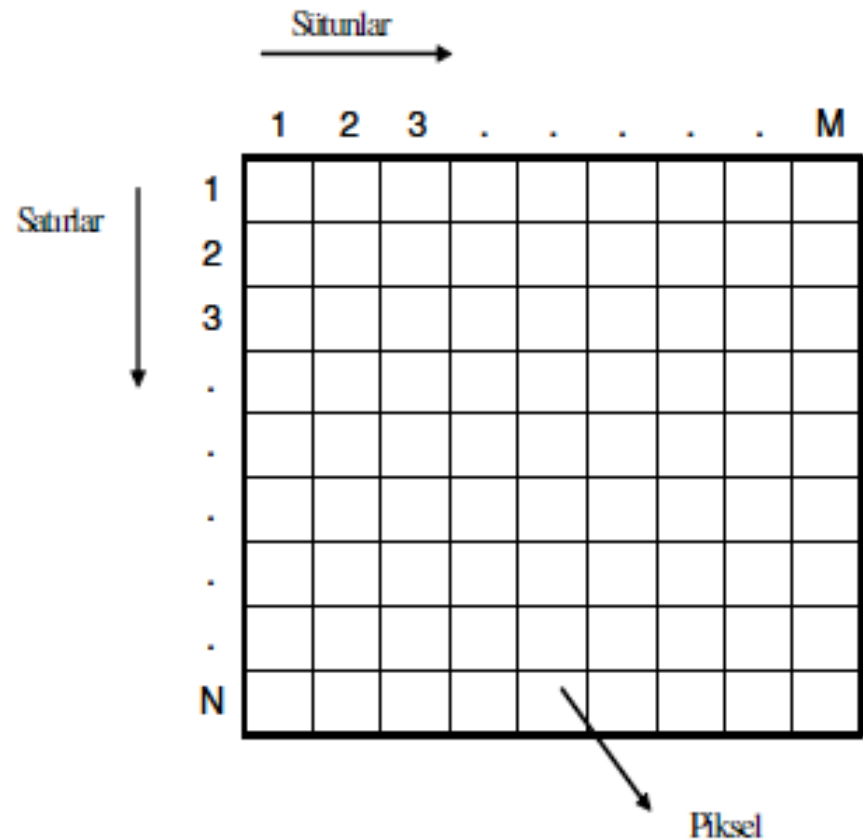
```
155 56 52 50 56 52 34 36 1 76 74 21 140 150 52 156 140 45 110 129 41 138 140 50 85  
156 11 74 95 14 35 97 23 95 74 25 73 82 19 87 75 21 40 45 7 35 38 6 84 54 19  
41 77 0 18 10 3 86 40 38 102 47 41 206 85 65 204 75 64 197 82 60 179 63 48 188 82  
48 168 48 40 52 68 21 60 68 11 40 51 17 28 37 0 28 29 0 83 50 15 2 0 1 43 14  
8 243 173 151 231 140 89 238 142 049 230 140 60 210 116 78 184 66 48 152 89 305 125 51  
27 104 41 25 85 46 8 30 27 0 288 28 2 29 28 1 40 28 18 13 10 1 224 167 112 249  
174 80 207 124 78 237 176 87 285 197 94 210 184 78 196 133 57 148 72 31 106 55 30 121  
61 12 128 30 24 131 48 35 55 21 1 12 5 0 14 16 11 5 0 0 157 126 85 244 205 125  
241 235 144 238 222 147 221 196 168 255 170 77 160 135 52 136 93 36 75 36 7 113 96 25  
156 63 33 101 52 21 31 14 7 5 19 0 20 14 12 250 214 112 342 115 106 245 227 120 239  
232 152 228 238 129 232 193 56 208 162 84 126 133 47 142 90 32 19 19 27 58 50 21 71  
116 48 114 61 29 75 49 24 10 8 5 11 50 1 227 102 62 248 221 122 141 226 129 240 219  
126 240 198 48 219 175 69 105 135 55 219 180 78 149 184 60 136 164 85 113 69 37 89 153  
80 122 74 56 80 51 18 16 37 47 86 37 30 213 177 15 208 288 195 240 218 125 238 208  
103 221 188 88 228 204 96 228 120 123 210 164 189 195 156 62 150 68 40 116 73 26 148 184  
48 109 59 34 75 48 18 27 33 33 48 185 186 216 177 96 223 168 11 228 228 111 226 213  
117 217 232 168 216 293 100 211 290 104 207 125 76 177 131 54 142 88 41 108 95 22 133  
50 22 90 55 10 75 50 17 0 10 2 54 76 74 100 111 102 218 104 108 228 228 152 228 330  
100 112 190 79 226 182 85 188 158 162 186 138 54 185 106 52 132 62 30 68 51 54 67 48  
15 51 46 18 10 15 8 11 6 0 68 30 51 54 90 90 220 186 87 217 100 105 214 177 96 298  
165 71 196 150 64 175 127 42 170 187 89 136 69 30 122 53 10 84 43 13 79 66 15 72 42  
14 10 13 4 12 8 0 68 104 119 58 96 109 120 135 115 186 134 62 180 148 66 163 130 78  
174 125 56 189 120 54 146 87 41 188 67 24 50 52 74 75 48 16 58 42 10 10 7 9 10 5  
0 18 11 3 66 111 116 70 169 122 216 103 66 57 71 62 162 111 66 161 96 31 152 101 51  
130 80 31 113 63 21 83 44 11 69 42 12 26 8 0 7 5 10 16 4 8 17 10 2 30 19 10  
58 58 66 53 80 64 58 01 102 60 60 110 54 60 70 25 60 65 21 14 25 63 41 25 11 2  
0 0 0 8 17 10 4 11 0 0 30 21 13 87 95 21 55 26 14 45 35 30
```

Sayısal resim

Sayısal Görüntü

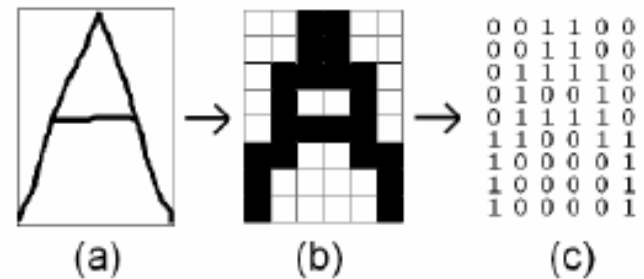
- $I(x, y)$ gibi bir fonksiyonla temsil edilen analog bir görüntü veya resimde
 - 'I' bir şiddet birimi (örneğin parlaklık),
 - x ve y ise görüntünün yatay ve dikey eksenindeki koordinatlarına karşı düşen değişkenlerdir.
- Sayısal görüntü ise, bu analog görüntünün M sütun ve N satırdan oluşacak şekilde örneklenmesi sonucu elde edilir.
- Satır ve sütunun kesiştiği her bölgeye **piksel** adı verilir.
- Sonuç olarak, sayısal görüntüye çevrilen resimde $N \times M$ adet piksel bulunur.

- 1 ve 0 değerleri sırasıyla aydınlık ve karanlık bölgeleri veya nesne ve zemini (nesnenin önünde veya üzerinde bulunduğu çevre zemini) temsil ederler.
- Sayısal (dijital) görüntü dosyaları renkli olarak genellikle 24 ya da 8 bit; gri-seviye görüntüler 1-2-4-6 ya da 8 bit olabilirler.

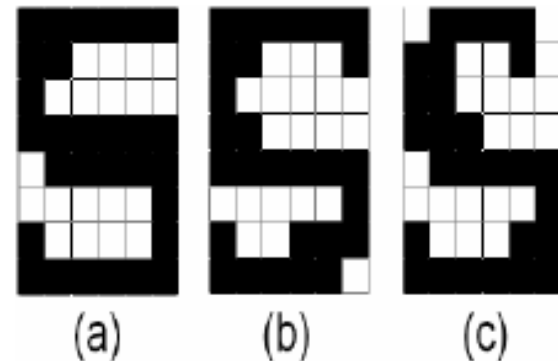


- Görsel karakterlerin sayısallaştırılması

1	1	1	1	1	1	1	1	1	1
1	0	0	0	1	1	0	0	0	1
1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	0	0	0	0	0	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1
1	1	0	1	1	1	1	0	1	1
1	0	0	0	1	1	0	0	0	1
1	1	1	1	1	1	1	1	1	1



- Sayısal bir görüntüye değer atanması ve saklanması örneği



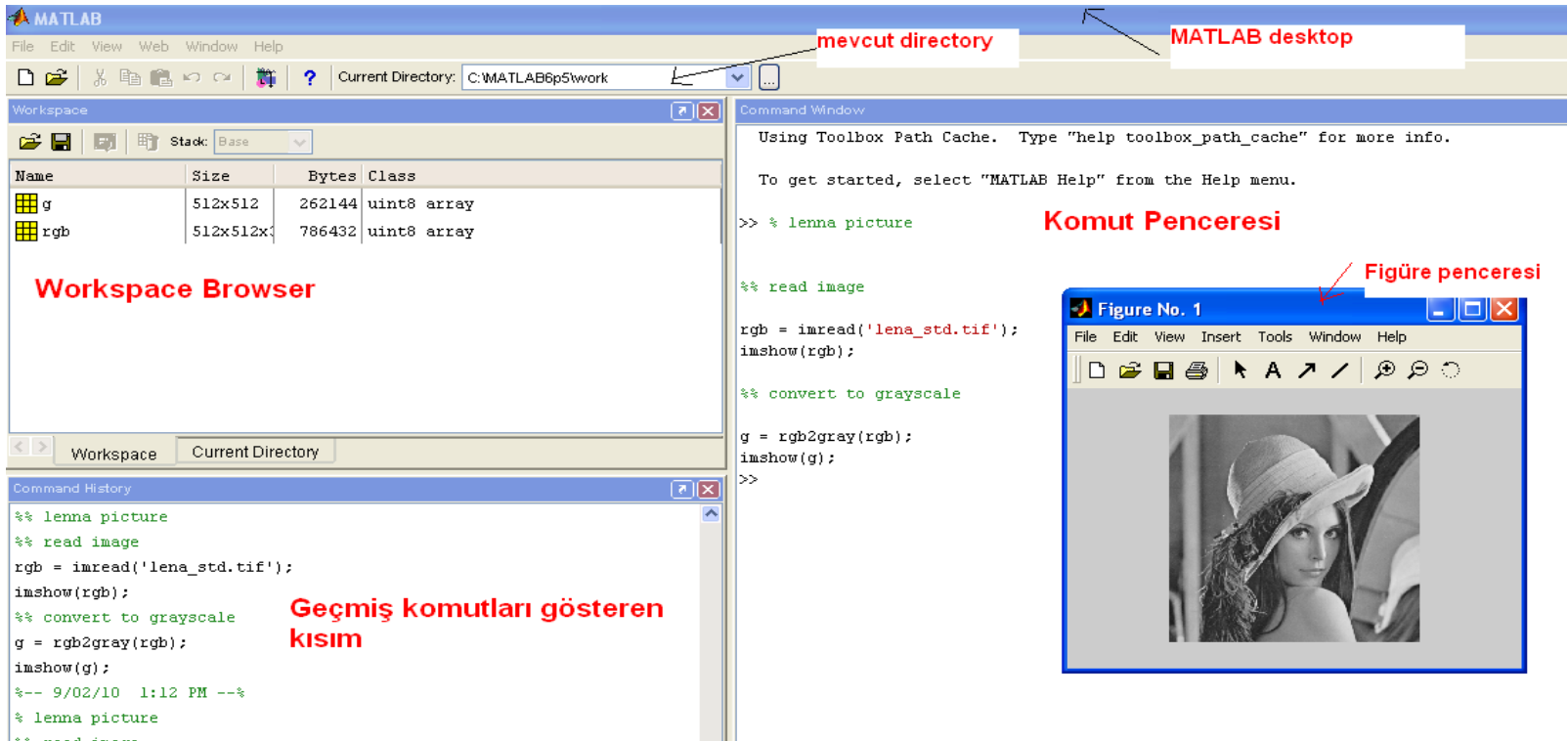
- S karakteri için 3 ayrı örnek

Sayısal Görüntü İşlemede MATLAB'ın kullanımı

- MATLAB (**MAT**rix **LAB**oratory), 1985'de C.B Moler tarafından, özellikle matris temelli matematik ortamında kullanılmak üzere geliştirilmiş etkileşimli bir paket programlama dilidir.
- MATLAB mühendislik hesaplamalarında; sayısal hesaplama, veri çözümleri ve grafik işlemleri için genel amaçlı bir programdır. Bununla beraber özel amaçlı modüler paketlere de sahiptir. CONTROL TOOLBOX, SIGNAL TOOLBOX, IMAGE PROCESSING TOOLBOX v.b gibi paket programlar (CACSD), denetim sistemlerinin tasarımında çok etkili araçlardır.
- Ayrıca WINDOWS ortamında çalışan SIMULINK, etkileşimli benzetim programlarının hazırlanması ve çalıştırılmasında büyük kolaylıklar sağlamaktadır.
- Bu derste MATLAB 2012 sürümünü kullanacağız. Öğrencilerin bu sürümü bilgisayarlarında çalışır hale getirmesi gerekmektedir.

MATLAB PAKET Programı Çalışma Ortamı

- MATLAB Desktop:** Matlab ana uygulama alanıdır. Program açıldığında gelen ilk kısımdır. Default görünüm olarak aşağıdaki gibidir.
- Command Window (Komut penceresi):** Matlab komutlarının ve denklemlerin, >> i şaretinin yazıldığı kısımdır. Sonuçlar ve komutlar burada görüntülenir.
- Workspace Browser:** Command Window'da tanımlanan değişkenler hakkında bilgi verilen yerdir. Değişken üzerine çift tıklandığında detay verir.
- Current Directory:** Şu anda çalışılan directory ismidir. C:\Matlab\Work directorysi default directory'dir. Çalışacağınız image dosyaları buraya kopyalanmışsa bu dosyaları sadece ismiyle işlem sokabilirsiniz.
- Command History Window (Geçmiş komutların gösterildiği yer):** Önceden yazılmış komutların görüldüğü yerdir. Buradan seçilen komutlar, komut penceresine kopyalanarak yeniden çalıştırılabilir.

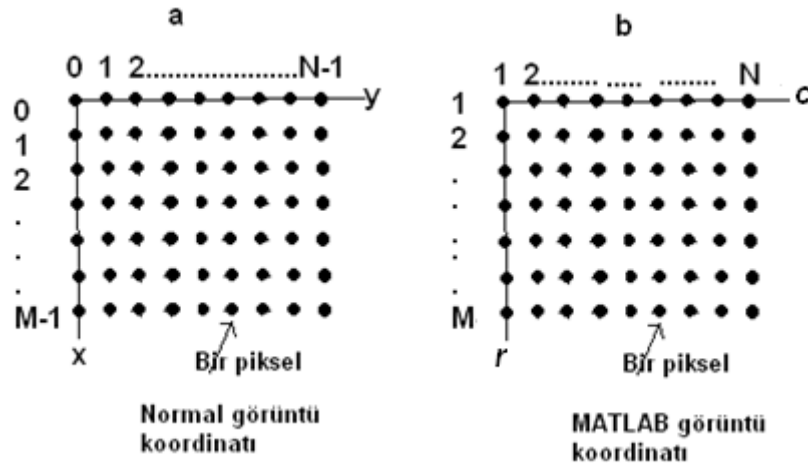


M-File oluşturmak

- Yazdığınız program parçalarını sonradan çalıştırabilmek için M-File haline getirebilirsiniz.
- Bunun için komut penceresinde
- `>> edit` yazarak veya **File** menüsünden **New** komutu ile; bir editör sayfası açarak program parçası yazılır. Ve belli bir isim ve `.m` uzantılı olarak saklanır.
- Bir M-file çağırmak için Matlab desktopta **file** menüsünden **open** komutu kullanılır.
- M-file doğrudan çalıştırmak için komut satırına `m file`'ın ismini yazmak yeterlidir.

Görüntü gösterimi

Bir sayısal görüntü, analog resmin örnekleme ve kuantalanması sonucunda elemanları reel sayılardan oluşan bir matrix formunda ifade edilir. Yani $f(x,y)$ şeklindeki bir sayısal image, M satır N sütundan oluşmuş MxN elemanlı bir matristir.



$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \dots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

Sayısallaştırılmış görüntüyü ifade eden matris formu

$$f = \begin{bmatrix} f(1,1) & f(1,2) & \dots & f(1,N) \\ f(2,1) & f(2,2) & \dots & f(2,N) \\ \vdots & \vdots & \dots & \vdots \\ f(M,1) & f(M,2) & \dots & f(M,N) \end{bmatrix}$$

Sayısal görüntüyü ifade eden MATLAB Matris formu

MATLAB'ın desteklediği önemli resim formatları

Format Name	Description	Recognized Extensions
TIFF	Tagged Image File Format	.tif, .tiff
JPEG	Joint Photographic Experts Group	.jpg, .jpeg
GIF	Graphics Interchange Format	.gif
BMP	Windows Bitmap	.bmp
PNG	Portable Network Graphics	.png
XWD	X Window Dump	.xwd

Formatlar arası dönüşüm komutları

Image format conversion

(Within the parenthesis you type the name of the image you wish to convert.)

Operation:

Matlab command:

Convert between intensity/indexed/RGB format to binary format.

`dither()`

Convert between intensity format to indexed format.

`gray2ind()`

Convert between indexed format to intensity format.

`ind2gray()`

Convert between indexed format to RGB format.

`ind2rgb()`

Convert a regular matrix to intensity format by scaling.

`mat2gray()`

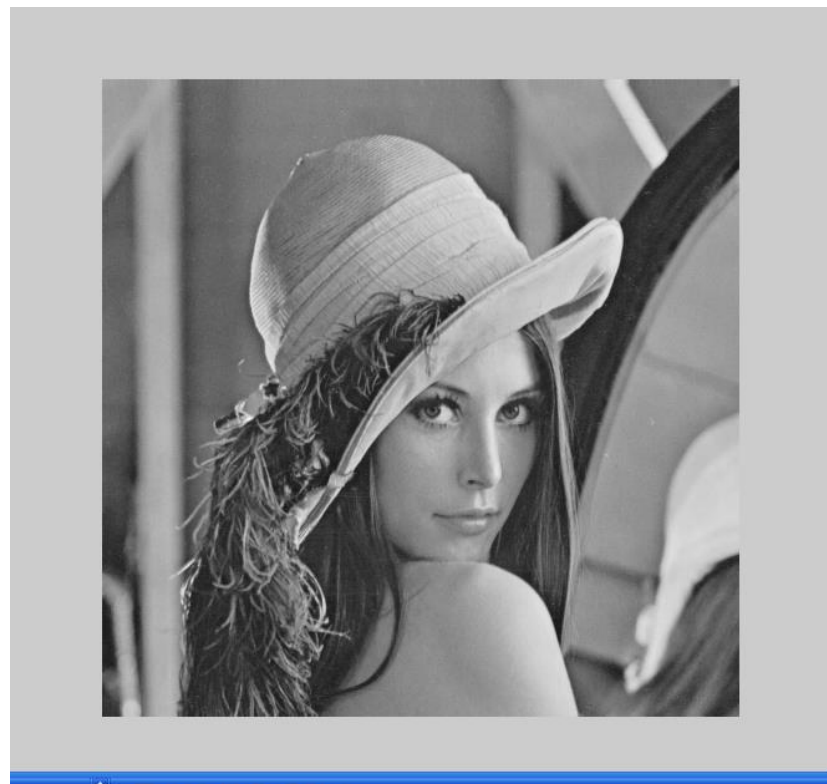
Convert between RGB format to intensity format.

`rgb2gray()`

Convert between RGB format to indexed format.

`rgb2ind()`

```
>> a=imread('lena_std.tif');  
>> imshow(a);  
>> b=rgb2gray(a);  
>> imshow(b);
```



Rgb2gray fonksiyonunu kullanmadan renkli resmi griye dönüştürme

```
Im=imread('lena_std.tif');  
figure,imshow(Im);  
title('Original Image');  
%0.2989 * R + 0.5870 * G + 0.1140 * B  
GIm=uint8(zeros(size(Im,1),size(Im,2)));  
for m=1:size(Im,1)  
    for n=1:size(Im,2)  
        GIm(m,n)=0.2989*Im(m,n,1) +  
            0.5870*Im(m,n,2) + 0.1140*Im(m,n,3);  
    end  
end
```

For döngüleri kullanmadan dönüşüm:

```
GIm=0.2989*Im(:,:,1)+0.5870*Im(:,:,2)+0.1140*Im(:,:,3);
```

MATLAB'DA Görüntü Okuma

- MATLAB'da görüntü datasının f dizisine aktarılması için;

>> f = *imread ('filename')* : ilgili resim dosyası work dizininde ise

>> f = *imread ('D:\myimages\rose.jpg')* ; ilgili resim dosyası başka dizinde ise.

- Bir resim datasının satır ve sütun olarak boyutunu görmek için ;

>> size(f)

ans=

1024 1024

- Resim datasını M satır ve R sütuna atamak için

>> [M,N] = size(f)

M=

512

N=

512

- Whos komutu resim datasına ait detaylı bilgi verir.

>> whos g

Name	Size	Bytes	Class
g	512x512	262144	uint8 array

Resmin Görüntülenmesi (Displaying Images)

>> `imshow(f,G)`

>> `imshow(f, [low,high])` komutu;

low değerinden eşit veya küçük değerleri siyah, high değerine eşit veya büyük değerleri beyaz olarak gösterir.

>> `imshow(f, [])` komutu;

f dizisinin en küçük değerini low değişkenine set eder. High değeri ise f dizisinin en büyük değeridir.



`imsow(g)`



`imshow(g,[50,100])`

```
f = imread('rose_512.tif');
```

```
whos f
```

```
imshow(f)
```

Name	Size	Bytes	Class
f	512x512	262144	uint8 array

Grand total is 262144 elements using
262144 bytes

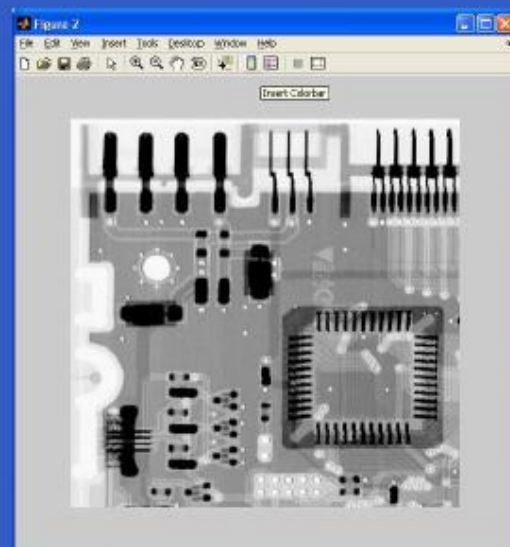


MATLABDA iki görüntüyü beraber görüntülemek

figure fonksiyonu kullanılarak gerçekleştirilir.

>> imshow(f), **figure**, imshow(g) formunda yazılarak

```
>> f=imread('rose_512.tif');  
>> g=imread('cktboard.tif');  
>> imshow(f), figure, imshow(g)
```



Görüntünün yazılması (Writing image)

İmage'ler diske **imwrite** fonksiyonu ile yazılır. Filename'den sonra MATLAB'ın desteklediği hangi formatta yazılacaksa muhakkak belirtilmelidir. Aşağıda örnekler verilmiştir.

Eğer filename isimli dosya çalışılacak dizine yazılacaksa yol tarif edilmesine gerek yoktur.

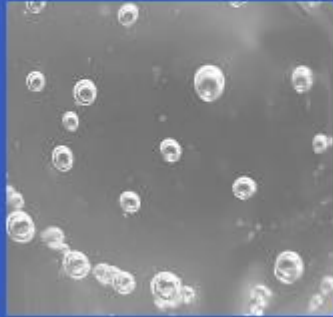
imwrite komutuyla JPG formatında kayıt yapılacaksa, 'quality', q yazılır. Burada q 0 ile 100 arasında bir tamsayıdır. Sıkıştırma oranını gösterir. En düşük sayı kalitesiz kayıt anlamındadır.

```
imwrite(f, 'filename')
```

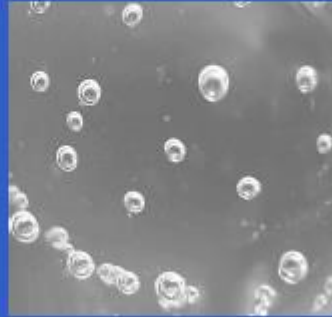
- `imwrite(f, 'patient10_run1', 'tif')`
- `imwrite(f, 'patient10_run1.tif')`

```
imwrite(f, 'filename.jpg', 'quality', q)
```

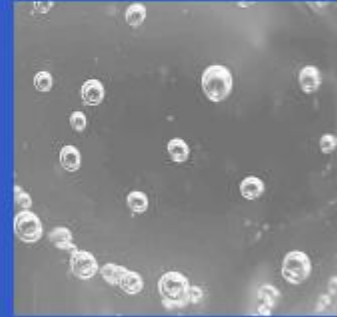
Writing Images



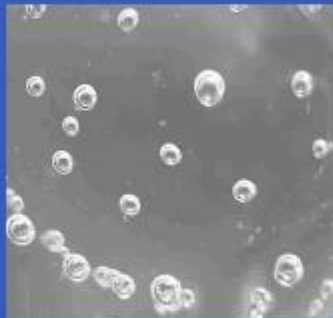
$q = 100$



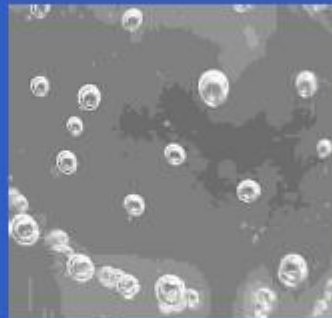
$q = 50$



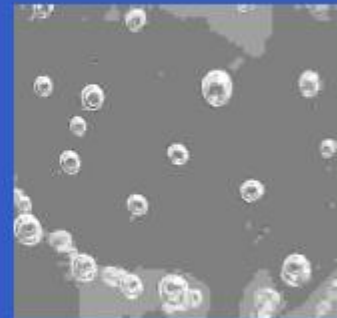
$q = 25$



$q = 15$



$q = 5$



$q = 0$

Size fonksiyonu

size(imagematrix)

```
>> size(f)
```

```
ans =
```

```
494 600
```

```
>> [M,N]=size(f);
```

```
>> whos f
```

Name	Size	Bytes	Class
------	------	-------	-------

f	494x600	296400	uint8 array
---	---------	--------	-------------

Grand total is 296400 elements using 296400 bytes

Image'in detaylı information'nu öğrenmek

```
imfinfo filename
```

```
>> imfinfo bubbles25.jpg
```

```
ans =
```

```
      Filename: 'bubbles25.jpg'  
      FileModDate: '02-Feb-2005 09:34:50'  
      FileSize: 13354  
      Format: 'jpg'  
      FormatVersion: ''  
      Width: 720  
      Height: 688  
      BitDepth: 8  
      ColorType: 'grayscale'  
      FormatSignature: ''  
      NumberOfSamples: 1  
      CodingMethod: 'Huffman'  
      CodingProcess: 'Sequential'  
      Comment: {}
```

- Buradaki Filesize kısmındaki rakam , resmin sıkıştırılmış haldeki kapladığı hafıza boyutunu (bayt olarak) ifade eder.
- Resmin orijinal büyüklüğü aşağıdaki gibi hesaplanır;
$$\text{Width}(720) \times \text{Height}(688) \times \text{Bith depth}(8) / 8 = 495360 \text{ bayt}$$

Resim bilgisindeki filesize (sıkıştırılmış resmin) kapladığı yerdir.
Buna göre sıkıştırma oranı;

Compression rate = Orijinal Boyut/ sıkıştırılmış boyut

$$\text{Sıkıştırma oranı} = 495360 / 13354 = 37.09$$

olarak bulunur. Bu işlem için matlab komutları bir sonraki slaytta verilmiştir.

İmfinfo fonksiyonu bilgilerini bir değişkene atayarak işlem yapmak

İmage teknik bilgilerini bir değişkene atadıktan sonra, image'in sıkıştırma oranı v.b işlemleri yapabiliriz. Aşağıda örnek görülmektedir.

```
>> K=iminfo('bubbles25.jpg');  
>> image_bytes=K.Width*K.Height*K.BitDepth/8;  
>> compressed_bytes=K.FileSize;  
>> compression_ratio=image_bytes/compressed_bytes
```

```
compression_ratio =
```

```
37.0945
```

TIF uzantılı görüntüler için en genel **imwrite** komut yazılımı aşağıdaki gibidir.

Packbits compression: Binary olmayan görüntüler için.

Ccitt compression: Binary görüntüler için.

Colres rowres: iki tamsayı değer ifade eder. Görüntünün birim yüzeyindeki (genellikle inç'tir) satır ve sütundaki dot (pixel) sayısını belirtir. **Res res** olarakta belirtilebilir.

```
imwrite(g, 'filename.tif', ...  
        'compression', 'parameter', ...  
        'resolution', [colres rowres])
```

'parameter':	'none'	no compression
	'packbits'	packbits compression
	'ccitt'	ccitt compression

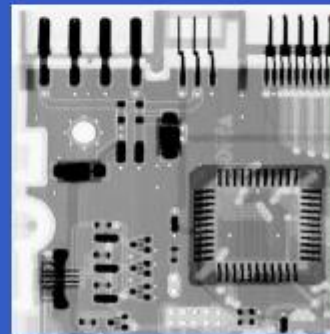
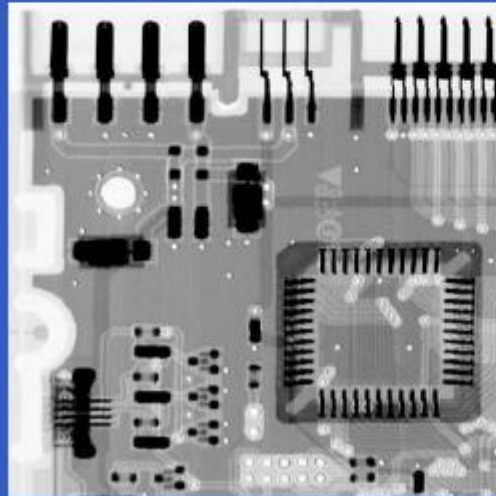
[colres rowres] contains two integers that give the column and row resolution in dots-per-unit (the default values are [72 72]).

- **Örnek:** 8 bit gri seviyeli , 200 dpi, jpg formatında bir görüntü olsun. Bu image 450x450 piksel'dir. Dolayısıyla resmin boyutu 2.25x2.25 inç olur. Bu resim f değişkenine atanmış olsun. Biz bu resmi tiff formatında, sıkıştırmadan, **sf** ismiyle saklamak isteyelim. Daha sonra 450x450 piksel sayısını değiştirmeden boyutunu 1.5x1.5 inç yapalım.
- 450 x 450 piksellik bir resmin çözünürlüğü 200 dpi ise bu resim;
 $450 \text{ dot} / 200 \text{ dpi} = 2.25 \times 2.25 \text{ inç}$ boyutlarında olur.
- 450 x 450 piksellik bir resmi 1.5 x 1.5 inç boyutunda ifade edersek; bir doğrultudaki piksel sayısı $450 / 1.5 = 300 \text{ dpi}$ olur.
- Veya 2.5 x 2.5 inç boyutundan 1.5 x 1.5 inç boyutuna düşürülürse, resmin resolasyonu
 $(2.25 / 1.5) \times 200 = 300 \text{ dpi}$ olur .

Bu işlemleri gerçekleştiren MATLAB kodu.

Round fonksiyonu, ilgili argüman değeri en yakın tamsayıya yuvarlatılır. Önemli not; imwrite komutuyla görüntünün pixel sayısı değiştirilemez, görüntünün boyutu (scalası) değiştirilebilir.

```
>> f=imread('cktboard.tif');  
>> res=round(200*2.25/1.5);  
>> imwrite(f, 'sf.tif', 'compression', ...  
          'none', 'resolution', res)
```



MATLAB görüntüsünü print etme işlemi; resim çerçevesindeki menüyü kullanarak veya aşağıdaki fonksiyonu kullanarak yapılır. Eğer sadece print komutu yazılırsa; enson resim print edilir.

```
print -fno -dfileformat -rresno filename
```

<i>no</i>	figure number in figure window
-----------	--------------------------------

<i>fileformat</i>	file format (in the earlier table)
-------------------	------------------------------------

<i>resno</i>	resolution in dpi
--------------	-------------------

Veri sınıfları (Data classes): Her ne kadar tamsayı koordinatlarda çalışılsa da piksel değerleri kendi içerisinde tamsayı olmayabilir. Dolayısıyla MATLAB'da desteklenen data sınıfları aşağıdaki tabloda verilmiştir. Bu sınıflardan herhangi birisiyle pixel değerleri ifade edilebilir. Bunlardan unit8 sınıfı gri seviye piksellerin ifadesi için en çok kullanılanıdır.

TABLO-1

Name	Description
double	Double-precision, floating-point numbers in the approximate range -10^{308} to 10^{308} (8 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).
single	Single-precision floating-point numbers with values in the approximate range -10^{38} to 10^{38} (4 bytes per element).
char	Characters (2 bytes per element).
logical	Values are 0 or 1 (1 byte per element).

İMAGE TİPLERİ

Matlab toolbox aşağıdaki görüntü tiplerini destekler

- Intensity images
- Binary images
- Indexed images
- RGB images

İNTENSITY IMAGE

- Bir *intensity image*, veri matrisi şeklindedir. *Matris elemanları piksellerin parlaklık şiddetini gösterir.*
- Bir gri seviye görüntünün uint8 veya uint16 sınıfında gösterilimi, elemanların [0,255] ve [0,65535] arasında tamsayılarla ifadesi anlamındadır.
- Eğer image *double* sınıfında ise, bu değerler kayan-nokta sayılardır. *Double* sınıflı scala değerli gri seviye görüntülerde eleman değerleri 0,1 arasında değişir.

Binary İmage

- Bir ikili görüntünün elemanları, 0 veya 1 gibi lojik dizidir.
- *Bir sayısal dizi, aşağıdaki şekilde binary formata dönüştürülür.*

$$B=\text{logical}(A)$$

- Bir dizinin binary olup olmadığını test etmek için *islogical* fonksiyonu kullanılır.

function: islogical(C)

C bir lojik dizi ise bu fonksiyon 1 gösterir.
Değilse 0 gösterir.

DATA Sınıfları arasında dönüşüm

$B = \text{data_class_name}(A)$

`data_class_name`, Tablo-1'in 1.sütnundakilerden biri olabilir.

- Eğer C double sınıfından bir dizi ise; ki onun tüm elemanlarının değeri 0,255 sahasındadır. Bu dizinin uint8 dizisine dönüştürülmesi için,

$D = \text{uint8}(C)$

- Eğer double sınıfındaki dizinin herhangi bir değeri 0, 255 sahasının dışına taşmışsa ve bu dizide uint8 sınıfına dönüştürülmüşse;

Matlab;

0' dan küçük değerleri 0'a,

255 'ten büyük değerleri ise 255'e

dönüştürür.

İmage tipleri ve veri sınıfları arasındaki dönüşüm

B=data_class_name(A) ;

data_class_name, Tablo-2'nin 1.sütündakilerden birisi olabilir.

Tablo-2

Name	Converts Input to:	Valid Input Image Data Classes
im2uint8	uint8	logical, uint8, uint16, and double
im2uint16	uint16	logical, uint8, uint16, and double
mat2gray	double (in range [0,1])	double
im2double	double	logical, uint8, uint16, and double
im2bw	logical	uint8, uint16, and double

İmage tipleri ve veri sınıfları arasındaki dönüşüm

2 x 2 görüntü dizisi f (double sınıfı) aşağıdaki gibidir.

```
>> f=[-0.5 0.5; 0.75 1.5]
```

f =

-0.5000	0.5000
0.7500	1.5000

Bu dizi *im2uint8* tipi diziye çevrilecekse;

```
>> g = im2uint8(f)
```

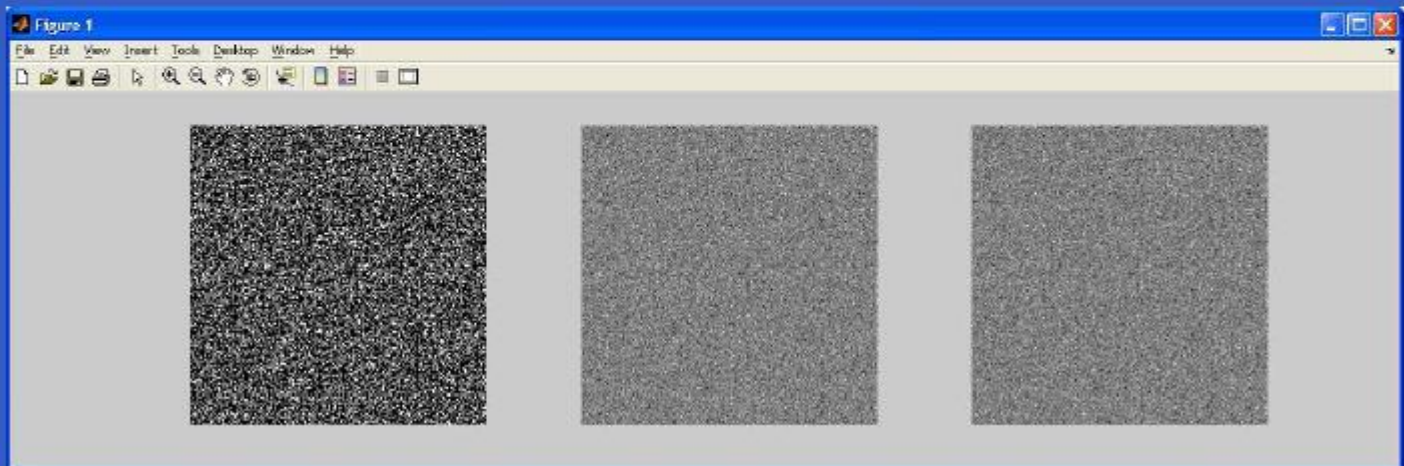
g =

0	128
191	255

Yapılan iş: Giriş datasındaki 0'dan küçük değerler 0'a set edilir. 1'den büyük değerler 255'e set edilir. Diğer giriş değerleri ise 255 ile çarpılarak dönüşüm sağlanır.

Dönüşüm örneği

```
>> A=randn(252);  
>> B=mat2gray(A);  
>> subplot(1,3,1), imshow(A), ...  
    subplot(1,3,2), imshow(A, []), ...  
    subplot(1,3,3), imshow(B)
```



Uint8 sınıfından *double* sınıfına dönüşüm

Bu işlem, giriş dizi elemanlarının herbirini 255'e bölme işlemidir. `im2double` fonksiyonu; giriş datasını `double` sınıf dataya dönüştürür. Eğer giriş dizi datası `unit8`, `unit16`, `logical` sınıfından ise `im2double` fonksiyonu `double` sınıfına dönüştürür. Eleman değerleri 0 ve 1 arasında değişir.

```
>> h=uint8([25 50; 128 200]);  
>> g=im2double(h)
```

```
g =
```

```
    0.0980    0.1961  
    0.5020    0.7843
```

mat2gray fonksiyonu: double sınıfı bir dizinin yine double sınıfı fakat eleman değerlerinin 0, 1 arasına yerleştirilmesi istendiğinde kullanılır.

$$G=\text{mat2gray}(A,[A_{\min},A_{\max}])$$

burada A dönüştürülecek dizi, **Amin** değerinden küçük dizi elemanları 0(Siyah), **Amax** değerinden büyük değerler ise 1 (Beyaz) olarak dönüştürülür.

Eğer; **g=mat2gray(f)** yazılırsa, min ve max değerleri f dizisindeki en küçük ve en büyük değerler olarak algılanır.

(Bir matrisden gray skala bir image dizisi oluşturur.)

```
>> f=[1 2; 3 4];  
>> g=mat2gray(f)
```

```
g =
```

```
          0          0.3333  
0.6667      1.0000
```

im2bw(f,T)

im2bw fonksiyonu, bir gri seviye veya indexlenmiş veya RGB giriş görüntü datasını bir binary (İkili) görüntüye dönüştürür.

Buradaki f dönüştürülecek gri seviye (intensity) görüntüdür. T ise eleman değerlerinin 0 veya 1 olmasını sağlayan eşik değeridir. Çıkış image dizisi logical sınıfındandır.

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

```
>> g=mat2gray(f)
```

```
g =
```

0	0.3333
0.6667	1.0000

```
>> gb=im2bw(g, 0.6)
```

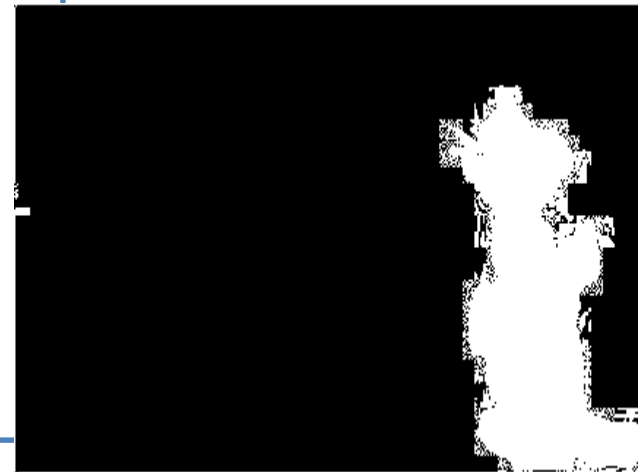
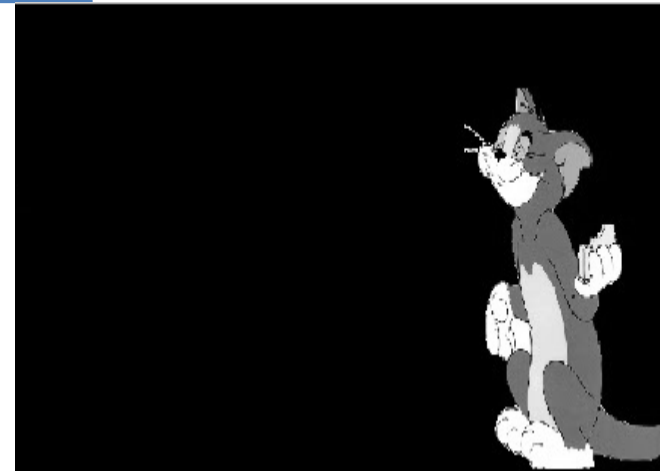
```
gb =
```

0	0
1	1

imbw2 fonksiyonu kullanmadan gri resmi binary resme dönüştürme

Original image

```
function mybinary
A=imread('tom.png');
figure,imshow(A);
title('Original image');
B=zeros(size(A,1),size(A,2));
for l=1:size(A,1)
    for m=1:size(A,2)
        if (sum(A(l,m,:))>0)
            B(l,m)=1;
        end
    end
end
figure(2)
imshow(B)
```



YIQ → RGB Dönüşümü

•MATLAB code:

%YIQ to RGB

%R=Y+0.956*I+0.621*Q

%G=Y-0.272*I-0.647*Q

%B=Y-1.106*I+1.703*Q

RGB=uint8(zeros(size(YIQ)));

for i=1:size(YIQ,1)

for j=1:size(YIQ,2)

RGB(i,j,1)=YIQ(i,j,1)+0.956*YIQ(i,j,2)+0.621*YIQ(i,j,3);

RGB(i,j,2)=YIQ(i,j,1)-0.272*YIQ(i,j,2)-0.647*YIQ(i,j,3);

RGB(i,j,3)=YIQ(i,j,1)-1.106*YIQ(i,j,2)+1.703*YIQ(i,j,3);

end

end

figure,imshow(RGB);

title('RGB Image');

RGB → YIQ

YIQ NTSC (National Television System Committee) formatının temelini oluşturur.

$$Y = 0.30R + 0.59G + 0.11B$$

$$I = 0.60R - 0.28G - 0.32B$$

$$Q = 0.21R - 0.52G + 0.31B$$

```
Im=imread('peppers.png');
figure,imshow(Im);
title('Original Image')
%y=0.2989 * R + 0.5870 * G + 0.1140 * B
%i=0.60*R - 0.28*G-0.32*B
%Q=0.21*R -0.52*G+0.31*B
YIQ=uint8(zeros(size(Im)));
for i=1:size(Im,1)
    for j=1:size(Im,2)
        YIQ(i,j,1)=0.2989*Im(i,j,1)+0.5870*Im(i,j,2)+0.1140*Im(i,j,3);
        YIQ(i,j,2)=0.596*Im(i,j,1)-0.274*Im(i,j,2)-0.322*Im(i,j,3);
        YIQ(i,j,3)=0.211*Im(i,j,1)-0.523*Im(i,j,2)+0.312*Im(i,j,3);
    end
end
figure,imshow(YIQ);
title('YIQ Image');
```

RGB→HSI Dönüşümü

1. RGB görüntüyü oku
2. RGB görüntüsünü [0,1] aralığında göster.
3. HSI bileşenlerini bul

$$\theta = \cos^{-1} \left[\frac{\frac{1}{2}[(R-G) + (R-B)]}{\left[(R-G)^2 + (R-B)(G-B) \right]^{1/2}} \right]$$
$$H(\text{Hue}) = \begin{cases} \theta & \text{If } B \leq G \\ 360 - \theta & \text{If } B > G \end{cases}$$

$$S(\text{Saturation}) = 1 - \frac{3}{(R + G + B)} [\min(R, G, B)]$$

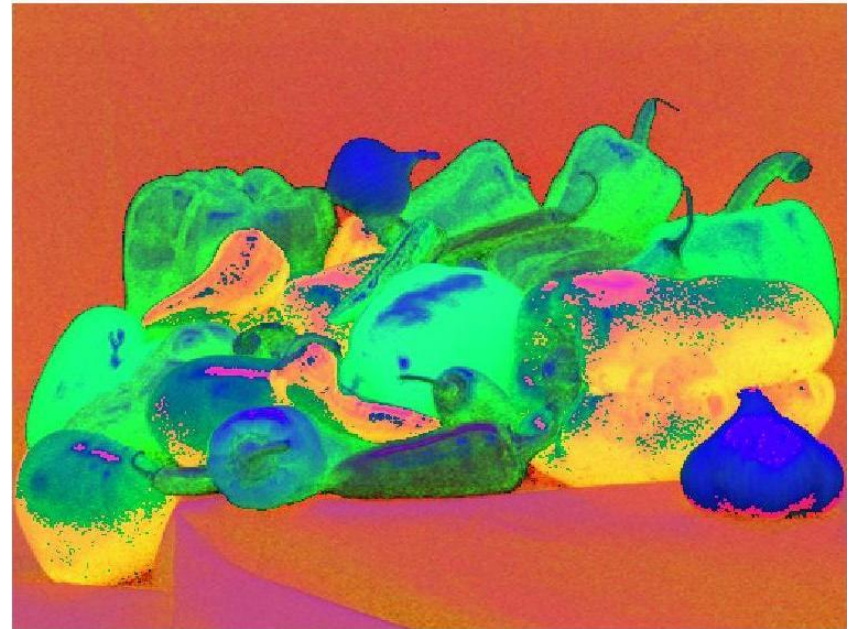
$$I(\text{Intensity}) = \frac{1}{3} (R + G + B)$$

RGB \rightarrow HSI Dönüşümü

RGB Image



HSI Image



F dizisinin elemanlarının 2'den büyüklerinde elemanı lojik 1 yap.

```
>> gb=f>2
```

```
gb =
```

0	0
1	1

```
>> gbv=islogical(gb)
```

```
gbv =
```

```
1
```

Matlab fonksiyonlarının içi içe kullanılması

- $F=[1 \ 2;3 \ 4]$ double image için;
- $G_b = \text{im2double}(\text{im2bw}(\text{mat2gray}(f),0.6))$

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

```
f =
```

```
    1    2  
    3    4
```

```
>> G_b = im2double(im2bw(mat2gray(f),0.6))
```

```
G_b =
```

```
    0    0  
    1    1
```

```
>> c=mat2gray(f)
```

```
c =
```

```
    0    0.3333  
0.6667    1.000
```

Dizi İndeksleme(Array indexing) Dizi Oluşturma

- Matlab birçok indeksleme formatını destekler. İndeksleme birçok bakımdan dizilerle işlem yapmayı kolaylaştırır.
- Burada tek ve iki boyutlu dizilerde indeksleme üzerinde durulacaktır.
- Vektör İndeksleme
- Matrix Indexing
- Dizi boyutlarının seçimi

Vektör indeksleme

- 1xN boyutlu bir dizidir. Satır vektör olarak bilinir. Tek boyutlu dizin oluşturmadır. V(1) vektörün, 1. elemanının, v(2) ikinci elemanın değerini gösterir.
- Matlab da vektör oluşturmak için elemanları [] içine yazmak gerekir.

- `>> v=[1 3 5 7 9]`

v =

1 3 5 7 9

- `>> v(2)` (vektörün 2. elemanın değerini göster.)

ans =

3

- `>> w=v.'` (satır vektörünü sütun vektörüne dönüştürmek)

w =

1

3

5

7

9

Vektördeki bir eleman bloğunu görmek için

- `>> v(1:3)` (Vektörün ilk üç elemanını verir. : işareti kolon işaretidir. çok önemli kolaylıklar sağlar)

ans =

1 3 5

- `>> v(2:4)` (v vektörünün 2,3,4. elemanlarını verir)

ans =

3 5 7

- `>> v(3:end)`

ans =

5 7 9

- `>> v=[1 3 5 7 9]`
`v =`
`1 3 5 7 9`
- `>> v(:)` (Bir sütün vektörü oluşturur. : İŞARETİ ÖNEMLİ İŞ YAPAR)
`ans =`
`1`
`3`
`5`
`7`
`9`
- `>> v(1:2:end)` (1.elemandan başlayıp 2şer 2şer son elemana kadar değerleri verir)
`ans =`
`1 5 9`
- `>> v(end:-2:1)` (Son elemandan başlayıp geriye doğru 2Şer sayarak değerleri bulur)
`ans =`
`9 5 1`
- `>> C=[1:3;3:5;10:12;20:22]` (3X3'lük Matris oluşturur. 1.satır 1den başlayıp 3'e kadar birer artan eleman değerleridir.)

`C =`

```

1   2   3
3   4   5
10  11  12
20  21  22
```

- `linspace(a, b, n)`, a başlangıç ve b son değerler arasında lineer olarak n elemanlı bir vektör oluşturur.

```
>> x = linspace(1,5,3)
```

```
x =
```

```
1 3 5
```

```
>> v=[1 3 5 7 9]
```

```
v =
```

```
1 3 5 7 9
```

```
>> v(x) (v vektörünün 1. 3. 5. elemanlarının seçimi için başka bir vektör (x vektörü gibi)  
kullanabiliriz.)
```

```
ans =
```

```
1 5 9
```

Matris İndeksleme

- Matrisler klasik olarak satır ve sütunların sırası şeklinde MATLAB'da gösterilir. Oluşturma formu;

```
>> A=[1 2 3; 4 5 6; 7 8 9]
```

```
A =
```

1	2	3
4	5	6
7	8	9

```
>> A(2,3)
```

```
ans =
```

```
6
```

Matris işlemleri

A Matrisinden,

Yeni bir kolon vektör dizisi elde etmek $C3=A(:,3)$, Buradaki 3, 3.kolonun dışarıya çıkartılması anlamındadır.

Yeni bir satır vektörü oluşturmak için $R2=A(2,:)$, Buradaki 2. 2.satırın çıkarılması anlamındadır.

Yeni bir matris dizisi oluşturmak için $T2=A(1:2,1:3)$, T2 matrisinin , A matrisinin ilk 2 satırından (1:2) ve ilk 3 sütunundan (1:3) oluştuğu anlaşılır.

```
A =  
    1    2    3  
    4    5    6  
    7    8    9
```

```
>> C3=A(:,3)
```

```
C3 =
```

```
    3
```

```
    6
```

```
    9
```

```
>> R2=A(2,:)
```

```
R2 =
```

```
    4
```

```
    5
```

```
    6
```

```
>> T2=A(1:2,1:3)
```

```
T2 =
```

```
    1
```

```
    2
```

```
    3
```

```
    4
```

```
    5
```

```
    6
```

A =

1	2	3
4	5	6
7	8	9

A matrisinden, son sütunu 0 olan yeni bir aynı boyutlu B matrisi yaratmak.

```
>> B=A;  
>> B(:,3)=0
```

B =

1	2	0
4	5	0
7	8	0

Matris operatörleriyle ilgili değişik uygulamalar aşağıdadır.

A(end,end) A matrisinin en son elemanını verir

A(end, end-2) A matrisinin enson satırındaki , end-2 sütunundaki elemanı verir.

E=A([1 3],[2 3]), A matrisindeki a(1,2),a(1,3), a(3,2),a(3,3) elemanlarını dışarı alır.

```
>> A(end,end)
```

```
ans =  
9
```

```
>> A(end,end-2)
```

```
ans =  
7
```

```
>> A(2:end,end:-2:1)
```

```
ans =  
6 4  
9 7
```

```
>> E=A([1 3],[2 3])
```

```
E =  
2 3  
8 9
```

Birçok karmaşık şemalar matris adreslemeleri kullanılarak rahatça gerçekleştirilebilir. A(D) formatı matris indeksleme için kullanılan faydalı bir adreslemedir. Burada D lojik bir dizidir.

```
>> D=logical([1 0 0; 0 0 1; 0 0 0])
```

```
D =
```

1	0	0
0	0	1
0	0	0

```
>> A(D)
```

```
ans =
```

1
6

: operatörü ile bir matris, sütun vektörü halinde ifade edilebilir.

```
T2 =
```

1	2	3
4	5	6

```
>> v=T2(:)
```

```
v =
```

1
4
2
5
3
6

Bir matrisin elemanları toplamının bulunmasında da : kolon operatörü kullanılabilir.

```
A =
```

1	2	3
4	5	6
7	8	9

```
>> s=sum(A(:))
```

```
s =
```

45

```
>> s1=sum(A)
```

```
s1 =
```

12	15	18
----	----	----

```
>> s2=sum(sum(A))
```

```
s2 =
```

45

Örnek:512x512 gri seviye bir f görüntü verisi uint8 sınıfındadır.

```
>> f=imread('rose_512.tif')  
>> imshow(f)
```



```
>> fp=f(end : -1:1, :)  
>> imshow(fp)
```



```
>> f=imread('rose_512.tif')  
>> imshow(f)
```



```
>> fp=f(:, end : -1:1)  
>> imshow(fp)
```



```
>> f=imread('rose_512.tif')  
>> imshow(f)
```



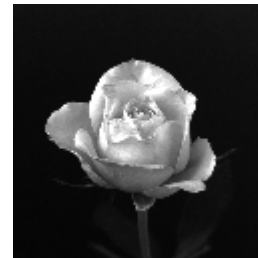
```
>> k = f(125:375, 125:375);  
>> imshow(k)
```



```
>> f=imread('rose_512.tif')  
>> imshow(f)
```



```
>> z = f(1:4:end,1:4:end);  
>> imshow(z)
```



Dizi Boyutu Seçme (Selecting Array Dimensions)

`operation(A, dim)`

- Bir dizinin satır veya sütun sayısını elde etmek için kullanılır. $A(M \times N)$ 'lik bir dizi olsun.
- `>> k=size(A,1);` A'nın ilk boyutu(dikey-vertical) boyunca size'ını verir. Başka bir deyişle bu komut row (dizi-satır)sayısını verir. `K=size(A,2)` ise satır boyunca size'ını verir.

Bazı önemli standart Diziler (Bazı basit imag'ler üretmek için kullanılırlar)

<code>zeros (M,N)</code>	generates an $M \times N$ matrix of 0s of class double.
<code>ones (M,N)</code>	generates an $M \times N$ matrix of 1s of class double.
<code>true (M,N)</code>	generates an $M \times N$ logical matrix of 1s.
<code>false (M,N)</code>	generates an $M \times N$ logical matrix of 0s.
<code>magic (M)</code>	generates an $M \times M$ "magic square".
<code>rand (M,N)</code>	generates an $M \times N$ matrix whose entries are uniformly distributed random numbers in the interval $[0,1]$.
<code>randn (M,N)</code>	generates an $M \times N$ matrix whose numbers are normally distributed random numbers with mean 0 and variance 1.

Önemli standart diziler (devam)

```
>> A=5*ones(3)
```

```
A =
```

5	5	5
5	5	5
5	5	5

```
>> magic(3)
```

```
ans =
```

8	1	6
3	5	7
4	9	2

```
>> B=rand(2,4)
```

```
B =
```

0.9501	0.6068	0.8913	0.4565
0.2311	0.4860	0.7621	0.0185

M-Function Programlama

- Bir M-Function'da aşağıda belirtilenlerden oluşur.
- M-Files
- Operators
- Flow Control
- Code Optimization
- Interactive I/O
- Cell Arrays and Structures

M-Files

- MATLAB M-File'ları sadece MATLAB'IN uygulanabilir deyimlerini yürütebilmek için,
- Veya bir function şeklinde; kabul edebilir bir veya daha fazla çıktılar üretebilir komutlar şeklinde olabilirler. Bunun için kullanıcı tarafından tanımlanabilirler.
- M-File'lar bir text editörü ile yaratılabilir ve filename.m formatında saklanabilirler.

Bir function M-File'in bileşenleri

- The function definition line (Fonksiyonun tanım satırı)
- The H1 line
- Help text (Yardım text'i)
- The function body (Fonksiyon gövdesi)
- Comments (Açıklamalar)

M-File

```
function [G,x] = planerot(x)
%PLANEROT Givens plane rotation.
% [G,Y] = PLANEROT(X), where X is a 2-component column vector,
% returns a 2-by-2 orthogonal matrix G so that Y=G*X has Y(2)=0.
%
% Class support for input X:
%     float: double, single

% Copyright 1984-2004 The MathWorks, Inc.
% $Revision: 5.10.4.1 $ $Date: 2004/04/10 23:30:05 $

if x(2) ~= 0
    r = norm(x);
    G = [x'; -x(2) x(1)]/r;
    x = [r; 0];
else
    G = eye(2,class(x));
end
```

Operators

- Arithmetic Operators : Sayısal hesaplamalar için.
- Relational Operators: nicelik/miktar karşılaştırma operatörüdür.
- Logical Operators and Functions: AND,OR,NOT gibi lojik işlemleri başarır.

Aritmetik Operatörler

Operator	Name	MATLAB Function	Comments and Examples
+	Array and matrix addition	<code>plus (A,B)</code>	$a+b$, $A+B$, or $a+A$.
-	Array and matrix subtraction	<code>minus (A,B)</code>	$a-b$, $A-B$, $A-a$.
<code>.*</code>	Array multiplication	<code>times (A,B)</code>	$C=A.*B$, $C(I,J)=A(I,J)*B(I,J)$.
<code>*</code>	Matrix multiplication	<code>mtimes (A,B)</code>	$A*B$, standard matrix multiplication, or $a*A$, multiplication of a scalar times all elements of A .
<code>./</code>	Array right division	<code>rdivide (A,B)</code>	$C=A./B$, $C(I,J)=A(I,J)/B(I,J)$.
<code>.\</code>	Array left division	<code>ldivide (A,B)</code>	$C=A.\B$, $C(I,J)=B(I,J)/A(I,J)$.
<code>/</code>	Matrix right division	<code>mrdivide (A,B)</code>	A/B is roughly the same as $A*inv(B)$, depending on computational accuracy.
<code>\</code>	Matrix left division	<code>mldivide (A,B)</code>	$A\B$ is roughly the same as $inv(A)*B$, depending on computational accuracy.

Aritmetik Operatörler-2

Operator	Name	MATLAB Function	Comments and Examples
.^	Array power	power (A,B)	If $C=A.^B$, then $C(I,J)=A(I,J)^B(I,J)$.
^	Matrix power	mpower (A,B)	Square matrix to the scalar power, or scalar to the square matrix power.
.'	Vector and matrix transpose	transpose (A)	$A.'$. Standard vector and matrix transpose.
'	Vector and matrix complex conjugate transpose	ctranspose (A)	A' . Standard vector and matrix conjugate transpose.
+	Unary plus	uplus (A)	$+A$ is the same as $0+A$.
-	Unary minus	uminus (A)	$-A$ is the same as $0-A$ or $-1*A$.
:	Colon		Discussed earlier.

Image Aritmetik Fonksiyonları

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

İmage Aritmetik Fonksiyonları

Toplama:

$$C(i,j,1)=\min(A(i,j,1)+B(i,j,1),255)$$

- Denklemden (i, j) piksel pozisyonunu gösterir.
- Görüntü toplama bir görüntüden diğerine bileşenleri eklemek için kullanılabilir.
- **Çıkarma**
- $C=A-B;$
- $C(i, j,:) = \max(A(i, j,:) - B(i, j,:), 0).$
- Görüntü çıkarma değişim tespiti için geniş bir şekilde kullanılır.



Subtracted Image



Original Image

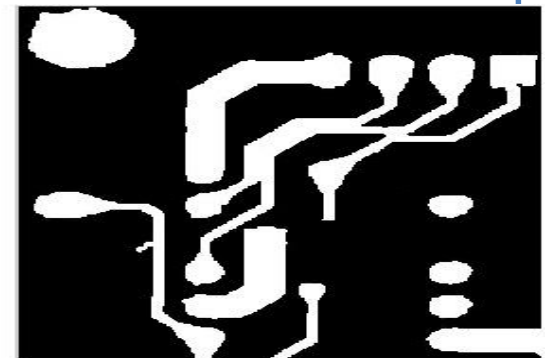


Image with defects

İmage Aritmetik Fonksiyonları

Çarpma:

- Görüntü çarpma görüntüdeki pikselleri bir sabit ile çarparak ortalama gri seviyeyi arttırmak için kullanılır.
- Ayrıca maskeleme işlemleri için de kullanılır. $C=A.*B$;

Bölme:

- $C=A./B$;

AND ve OR

- Görüntünün kalanından ilgili alanı izole etmek için mantıksal AND veya OR kullanılır. Maske L ve A görüntüsü için ilgilenen alanı elde etmek için $D=\text{and}(L,A)$ işlemi uygulanır.

Örnek:

Background Image



Tom



Jerry



Image Addition



Örnek:

```
background=imread('back.jpg');  
A=imread('tommy1.bmp');  
B=imread('jerry1.bmp');  
object=A+B;  
background=imresize(background,[size(object,1) size(object,2)]);  
Im3=uint8(zeros(size(object)));  
whitelmg=uint8(ones(size(object)));  
mask=whitelmg./object;  
im3=uint8(mask&background);  
figure,imshow(mask);  
finalImg=(background.*im3)+object;  
figure,imshow(finalImg)
```

Örnek:

Image Mask



Final Image



Görüntü üzerinde bazı işlemler: devirme

$$B = A'$$

$$B(j, i) = A(i, j)$$

$$(i = 0, \dots, N-1, j = 0, \dots, M-1)$$



Görüntü üzerinde bazı işlemler: düşeyde çevirme

$$B(i, M - j - 1) = A(i, j) \\ (i = 0, \dots, N - 1, j = 0, \dots, M - 1)$$



Görüntü üzerinde bazı işlemler: kırpma

$$B(i, j) = A(n_1 + i, n_2 + j)$$
$$(i = 0, \dots, m_1 - 1, j = 0, \dots, m_2 - 1)$$

- $(n_1, n_2) \rightarrow$ Başlangıç noktası
- $(m_1, m_2) \rightarrow$ pencere boyutları



Görüntü üzerinde bazı işlemler: öteleme

$$B(i, j) = A(i - n_1 + 1, j - n_2 + 1)$$
$$(i = n_1, \dots, N, j = n_2, \dots, M)$$

$(n_1, n_2) \rightarrow$ Başlangıç noktası



Örnek:

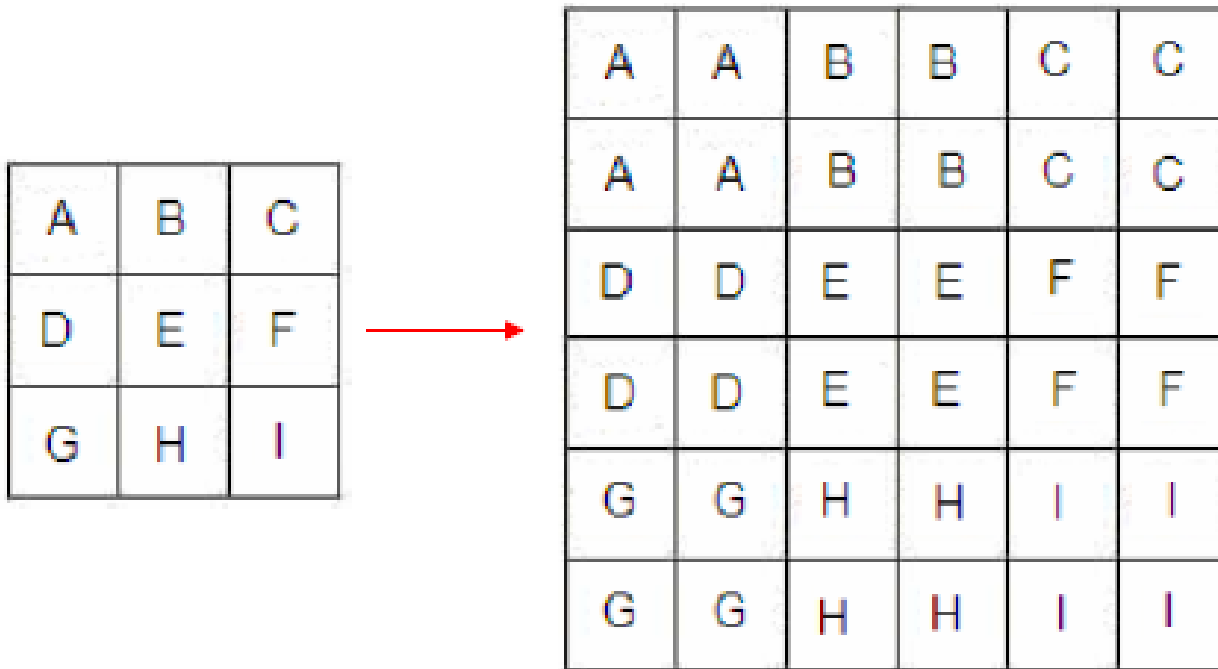
- Öteleme işlemi yapan bir Matlab işlevi yazalım:

```
function [B]=my_otele(A,n1,n2)  
    [w,h]=size(A);  
    B=zeros(w,h);  
    for i=n1:w  
        for j=n2:h  
            B(i,j)=A(i-n1+1,j-n2+1);  
        end  
    end
```

- Burada for döngüleri yerine tek bir satır yazarak aynı işlem yapılabilir.

Boyut deęiřtirme-yakınlařtırma


- Yakınlařtırma, düşük piksel boyutlu bir imgenin piksel boyutunun yazılımsal olarak arttırılmasıdır.
- Sayısal yakınlařtırma (digital zoom).



Boyut değiştirme-yakınlaştırma

- Boyut büyültmede daha yumuşak geçişler için:

A	B	C
D	E	F
G	H	I



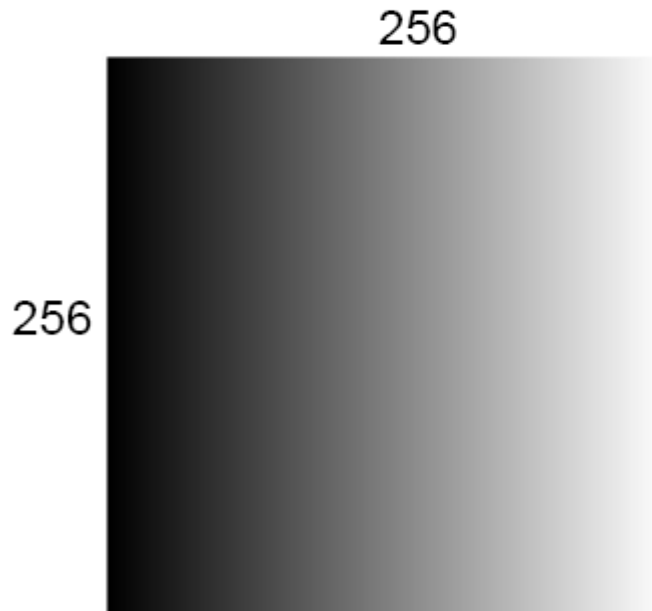
A	$\frac{A+B}{2}$	B	$\frac{B+C}{2}$	C
$\frac{A+D}{2}$	$\frac{A+B+D+E}{4}$	$\frac{B+E}{2}$	$\frac{B+C+E+F}{4}$	$\frac{C+F}{2}$
D	$\frac{D+E}{2}$	E	$\frac{E+F}{2}$	F
$\frac{D+G}{2}$	$\frac{D+E+G+H}{4}$	$\frac{E+H}{2}$	$\frac{E+F+H+I}{4}$	$\frac{F+I}{2}$
G	$\frac{G+H}{2}$	H	$\frac{H+I}{2}$	I

Boyut Değiştirme

- Matlab ile boyut değiştirme için “imresize” adındaki işlev kullanılabilmektedir.
`Is=imresize(I,oran,yöntem);`
- oran : giriş imgesinin boyutunun değişme oranını verir. $\text{oran} > 1$ (büyütme), $\text{oran} < 1$ (küçültme).
- yöntem : boyut değiştirmede kullanılacak aradeğerleme yöntemi.
- Örnek; **`Is=imresize(I,0.97, 'bicubic');`**

İmge oluşturma

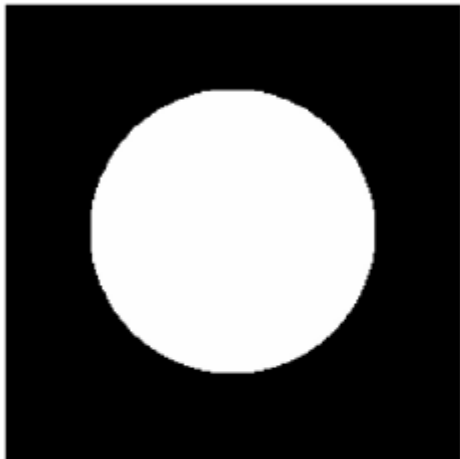
$$\mathbf{A} = \left[\begin{array}{ccccc} 0 & 1 & 2 & \dots & 255 \\ 0 & 1 & 2 & \dots & 255 \\ \vdots & & & & \\ 0 & 1 & 2 & \dots & 255 \end{array} \right] \left. \vphantom{\begin{array}{c} 0 \\ 0 \\ \vdots \\ 0 \end{array}} \right\} 256 \text{ rows}$$



```
for i = 1 : 256
    for j = 1 : 256
        A(i,j) = j - 1;
    end
end
```

İmge oluşturma

- (128,128) merkezli, yarıçapı 80 piksel beyaz bir daire



$$B(i,j) = \begin{cases} 255 & \text{if } \sqrt{(i-128)^2 + (j-128)^2} < 80 \\ 0 & \text{otherwise} \end{cases}$$

```
for i = 1 : 256
    for j = 1 : 256
        dist = ((i - 128)^2 + (j - 128)^2)^(.5);
        if (dist < 80)
            B(i,j) = 255;
        else
            B(i,j) = 0;
        end
    end
end
```

İmge Oluşturma



$$C = A \times B / 255$$

```
for i = 1 : 256  
    for j = 1 : 256  
        C(i,j) = A(i,j) * B(i,j)/255;  
    end  
end
```


Ödev:

Teslim Tarihi: 5 Kasım 2013

1. Görüntü üzerinde düşeyde çevirme, boyut değiştirme(imresize), kırpma, öteleme işlemlerini gerçekleştiren MATLAB kodlarını yazınız?
2. RGB→HSI dönüşümü için MATLAB kodunu yazınız?