

# Görüntü İşleme

**K.Sinan YILDIRIM**

**Cenk İNCE**

**Tahir Emre KALAYCI**

Ege Üniversitesi

Bilgisayar Mühendisliği Bölümü

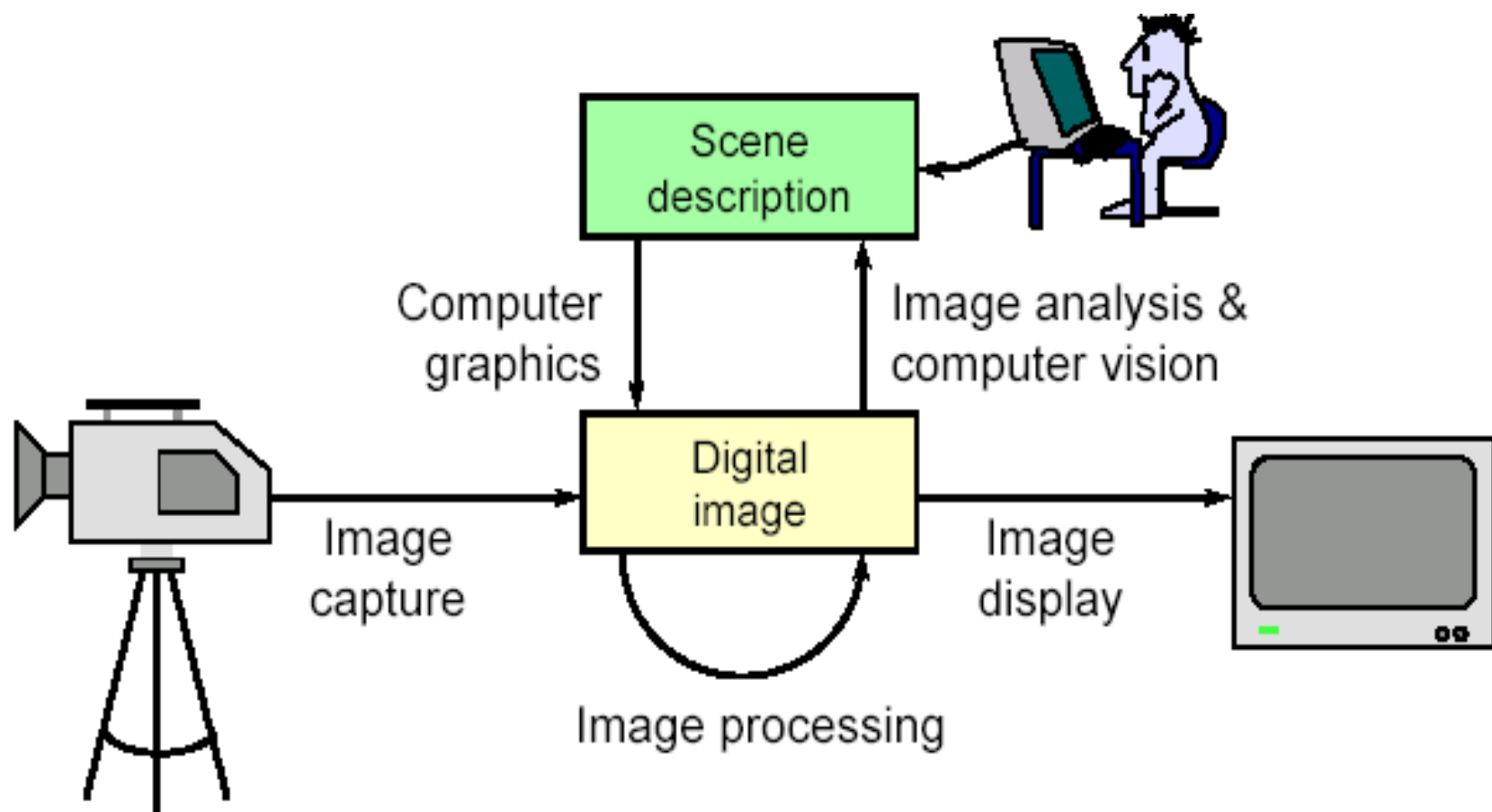
2003 ©

# İçerik

- **Görüntü İşleme Nedir?**
- **Görüntü Tanımlamaları**
- **Görüntü Operasyonları**

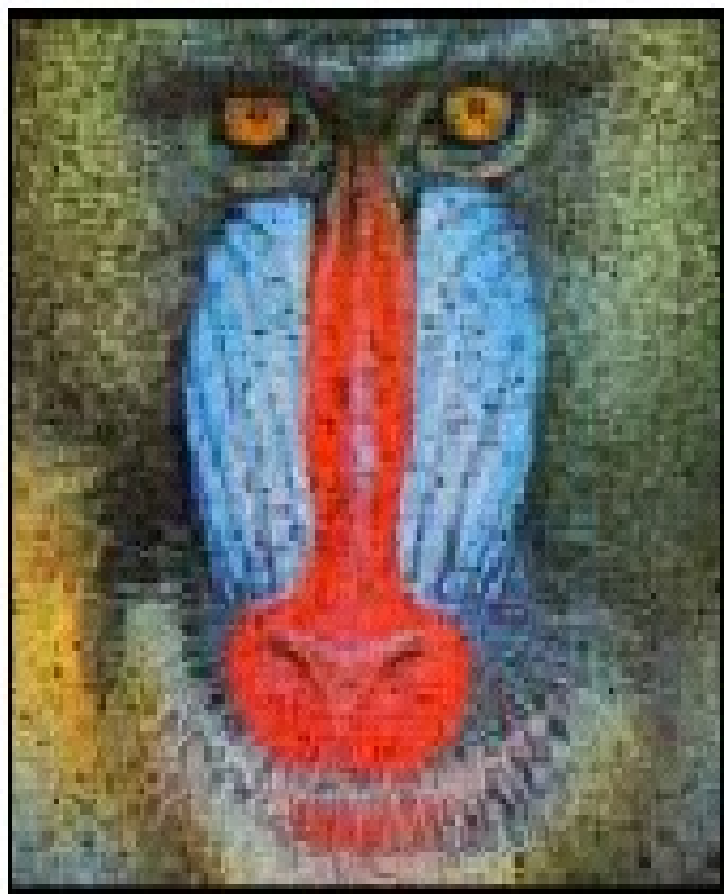
# Görüntü İşleme Nedir?

- ◆ Dijital bir resim haline getirilmiş olan gerçek yaşamdaki görüntülerin, bir girdi resim olarak işlenerek, o resmin özelliklerinin ve görüntüsünün değiştirilmesi sonucunda yeni bir resmin oluşturulmasıdır.



# Görüntü İşlemenin Yararları- 1

- ◆ Resimler genellikle analog ortamlardan dijital ortamlara geçirildiği için bozukluk (noise) içerir. Görüntü işleme bu hataları düzeltmek için kullanılabilir.

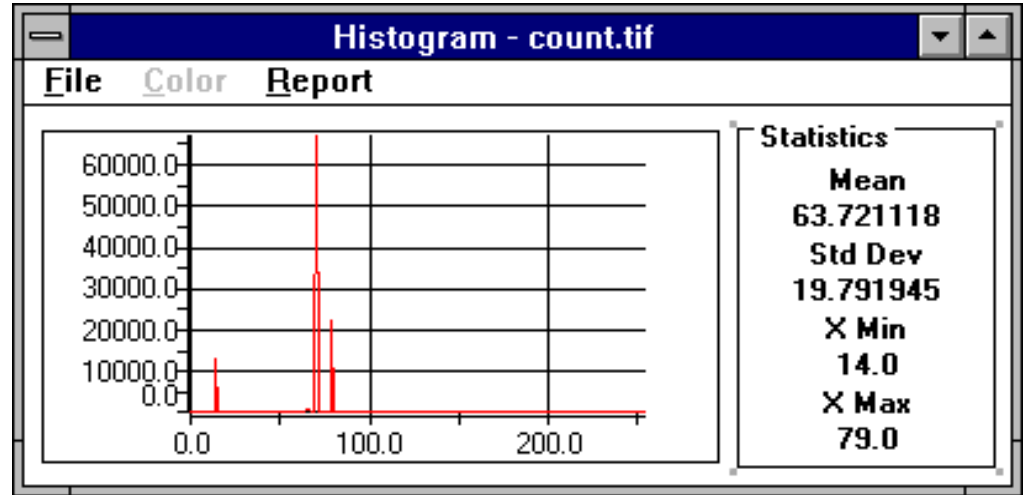
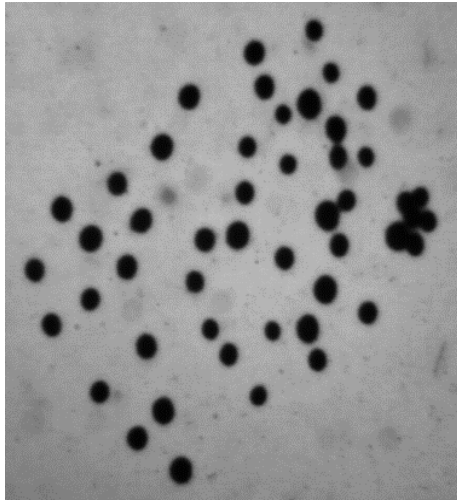


noise removal



# Görüntü İşlemenin Yararları- 2

- ◆ Resimler faydalı işler için işlenir.



-Bir görüntüdeki kan sayısını saymak.

-Zarfların üzerinden Zip-Kodun okunması

# Görüntü işlemek için neler yapılır?



- ◆ Gürültü” Temizlemek
- ◆ Parlaklığı Ayarlamak
- ◆ Koyuluk Ayarlamak
- ◆ Görüntü keskinleştirmek ve bulanıklaştırmak
- ◆ Doğru renk ayarlamak

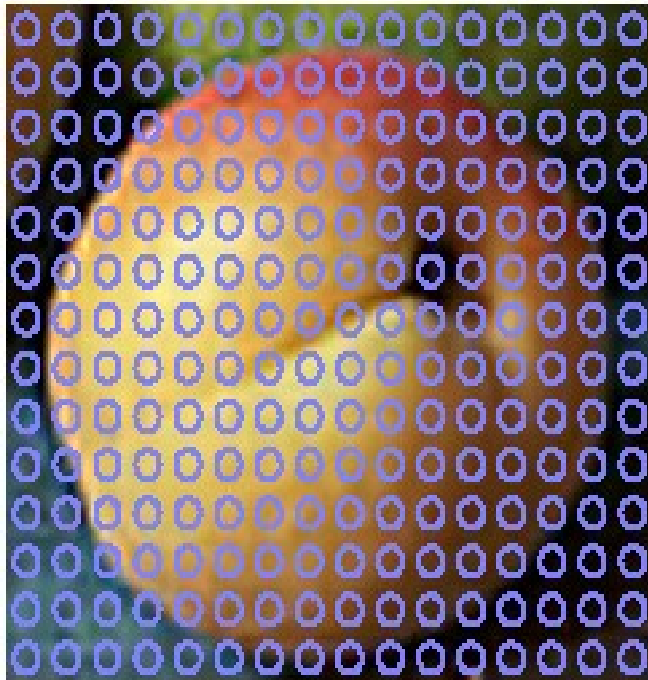


# Görüntü Nedir ?

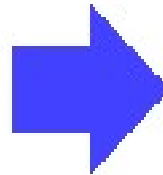
- ◆ Gerçek yaşamda, bir görüntü veya resim basit 2 değişkenin bir fonksiyonu olarak tanımlanır.
- ◆  $a(x,y)$  gibi bir fonksiyonla ifade edilen bir resimde  $a$  bir şiddet birimi( örneğin parlaklık) ve  $x$  ve  $y$  değişkenleri ise resmin gerçek koordinatlarıdır.

# Dijital Görüntü Tanımlamaları

- ◆ Dijital görüntü sayısal değerlerden oluşur.
- ◆ 1 ve 0'lardan oluşan sayısal görüntü yapımız  $a[m,n]$ , 2 boyutlu dünyadan elde edilen  $a(x,y)$  fonksiyonundan örnekleme tekniği kullanılarak oluşturulur.



A real image

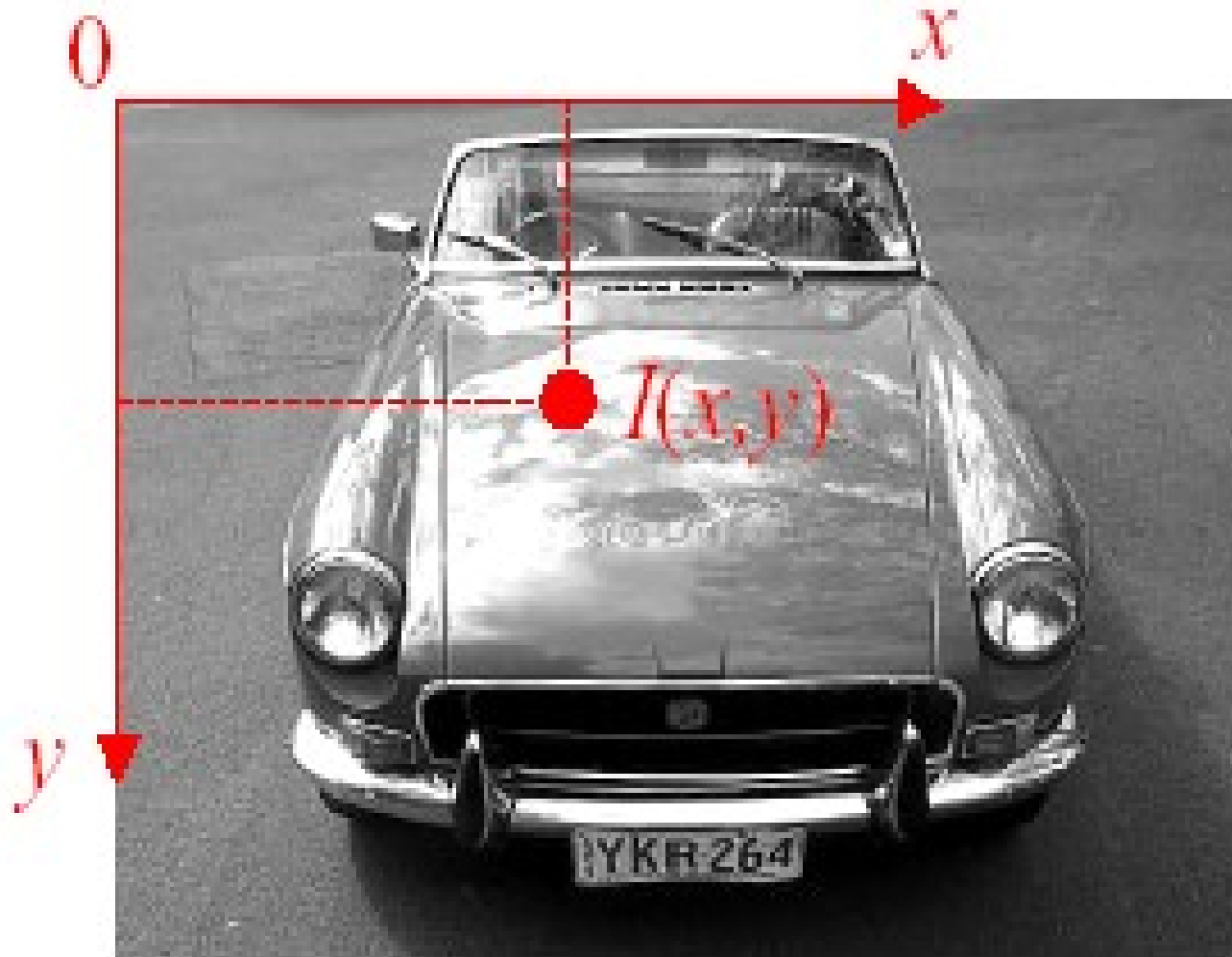


103	59	12	80	56	12	34	30	1	78	79	21	145	156	52	136	143	85	115	129	41	128	143	50	85				
106	11	74	96	14	85	97	23	66	74	23	73	82	29	67	76	21	40	48	7	33	39	9	94	54	19			
42	27	6	19	10	3	59	60	28	102	107	41	208	88	63	204	75	54	197	82	63	179	63	46	158	62			
48	146	49	40	52	65	21	60	68	11	40	51	17	35	37	0	28	29	0	83	50	15	2	0	1	13	14		
8	243	173	161	231	140	89	239	142	89	230	143	90	210	126	79	184	88	48	152	69	35	123	51					
27	104	41	23	55	45	9	36	27	0	28	28	2	29	28	7	40	28	18	13	13	1	224	167	112	240			
174	80	227	174	78	227	176	87	233	177	94	213	149	78	196	123	57	141	72	31	108	53	22	121					
62	22	126	50	24	101	49	35	16	21	1	12	5	0	14	16	11	3	0	0	237	176	83	244	206	123			
241	236	144	238	222	147	221	190	108	215	170	77	190	135	52	136	93	38	76	35	7	113	56	26					
156	83	38	107	52	21	31	14	7	9	8	0	20	14	12	255	214	112	242	215	108	246	227	133	239				
232	152	229	209	123	232	193	98	208	182	84	179	133	47	142	90	32	29	19	27	89	53	21	171					
116	49	114	64	29	75	49	24	10	9	5	11	16	9	237	190	82	249	221	122	241	225	129	240	219				
126	240	199	93	218	173	69	188	135	33	219	186	79	189	184	93	136	104	85	112	69	37	191	153					
80	122	74	28	80	51	19	19	37	47	16	37	32	223	177	83	235	208	105	243	218	125	238	206					
103	221	188	83	228	204	96	224	220	123	210	194	109	192	159	62	150	98	40	116	73	28	146	104					
48	109	59	24	75	48	18	27	33	33	47	100	118	216	177	96	223	189	91	239	209	111	236	213					
117	217	200	108	218	200	100	218	206	104	207	175	76	177	131	54	142	88	41	108	85	22	103						
59	22	93	53	18	76	50	17	9	10	2	54	76	74	108	111	102	218	194	106	228	203	102	228	200				
100	212	180	79	220	162	85	198	156	82	180	138	54	155	106	37	132	82	33	95	51	14	87	48					
15	81	46	14	16	15	0	11	6	0	64	90	91	54	80	93	220	186	97	212	190	105	214	177	86	208			
165	71	196	150	64	175	127	42	170	117	49	139	89	30	102	53	12	84	43	13	79	46	15	72	42				
14	10	13	4	12	8	0	89	104	110	58	96	109	130	128	115	198	154	82	198	148	68	183	138	70				
174	125	58	169	120	54	146	97	41	118	67	24	90	52	16	75	46	16	58	42	19	13	7	9	10	5			
0	18	11	3	66	111	116	70	100	102	78	103	99	57	71	82	162	111	68	141	96	37	152	102	51				
130	80	31	110	63	21	83	44	11	69	42	12	28	8	0	7	5	10	18	4	0	17	10	2	30	20	10		
58	88	96	53	88	94	59	91	102	69	99	110	54	80	79	23	69	85	31	34	25	53	41	25	21	2			
0	8	0	0	17	10	4	11	0	0	34	21	13	47	35	23	38	26	14	47	35	23							

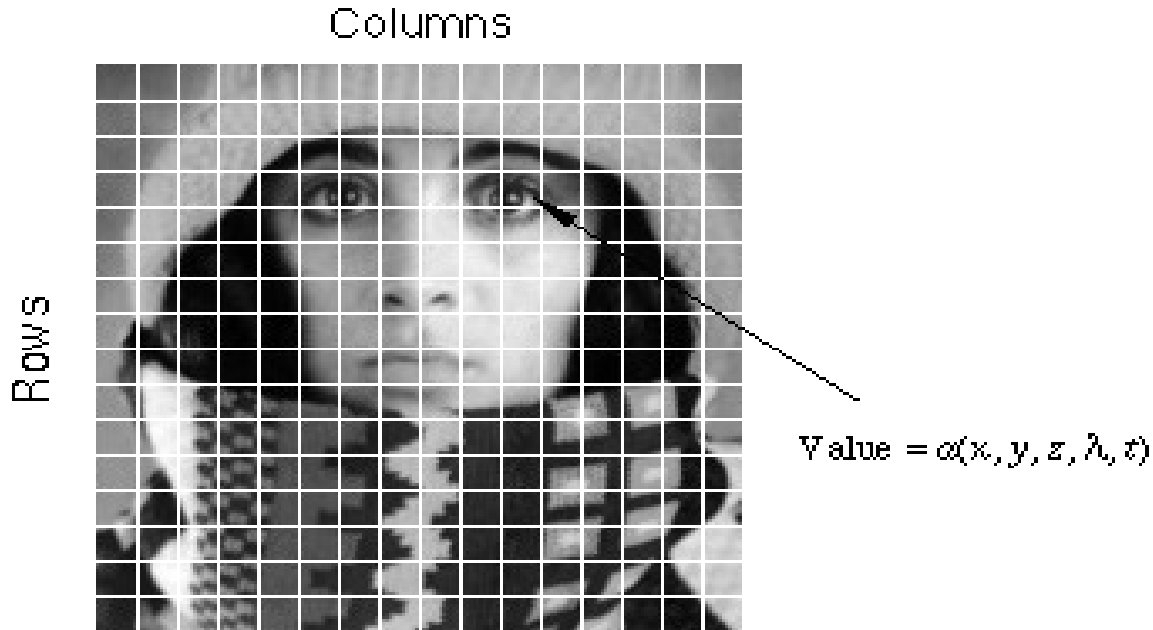
A digital image

# Dijital Görüntü Tanımlamaları

- ◆ Sayısal görüntümüz  $M$  ve  $N$  sayılarında satır ve sütünlardan oluşur ve satır ve sütünların kesiştiği her bölgeye pixel denir.
- ◆ O piksel'deki değer ise derinlik ( $z$ ) , renk( $\lambda$ ) ve zamanın( $t$ ) bir fonksiyonudur.



# Görüntü Tanımlama Örneği



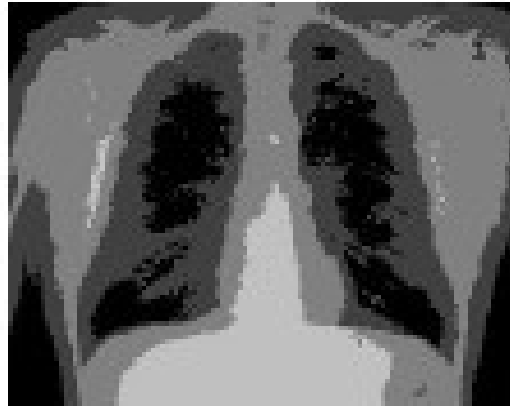
- ◆ Yukarıdaki resimde  $N=16$  ve  $M=16$  'dır. Her pixele atanan değer ise o pikselin parlaklık değerinin en yakın tamsayıya yuvarlatılmış halidir.  $[m=10, n=3]$  'teki değer  $L=110$ 'dur.

# Niteliklendirme

- ◆ Görüntünün piksel değerlerinin belirli aralıklarda olması, meydana gelen görüntünün niteliğini değiştirir.
- ◆ Örneğin 0 beyazı ve  $n-1$  de siyahı temsil ederse ve bu değerler arası gri tonlarını ifade eder.
- ◆ Burada  $n = 2^b$  olmak üzere,  $b$  değeri görüntünün 1 pikselini ifade etmek için gereken bit sayısıdır.
  - Örneğin  $b=8$  ise 256 adet gri tonu bulunmaktadır.
- ◆  $B=1$  ise resim sadece 0 ve 1 'lerden oluşur ve buna İkili resim(Binary Image) denir.



$n=2$



$n=4$



$n=8$



$n=16$



$n=64$

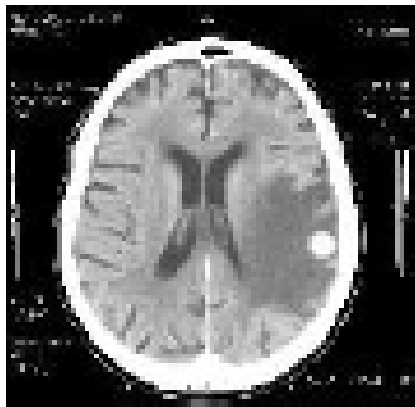


$n=256$

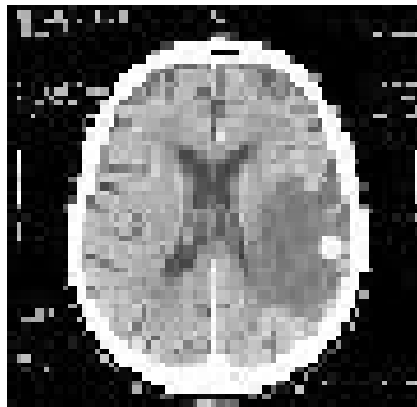


# Uzaysal Çözünürlük

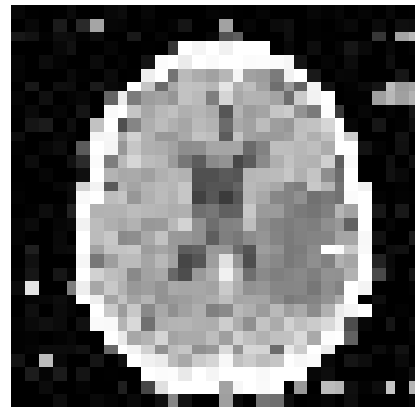
- ◆ Bir resmin uzaysal çözünürlüğü (Spatial Resolution) , o resmin 1 pikselinin fiziksel büyüklüğüne eşittir.
- ◆ Kısaca, bir resmin detaylanabilir en küçük parçasıdır.



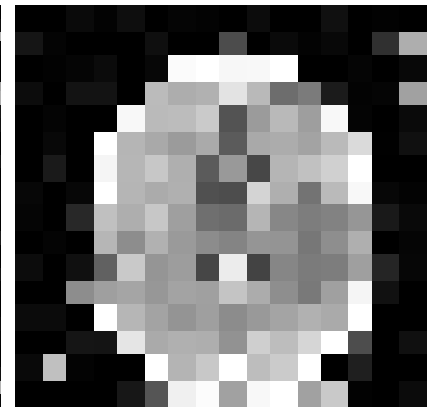
128x128



64x64



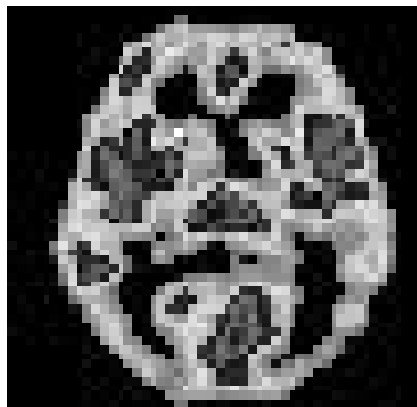
32x32



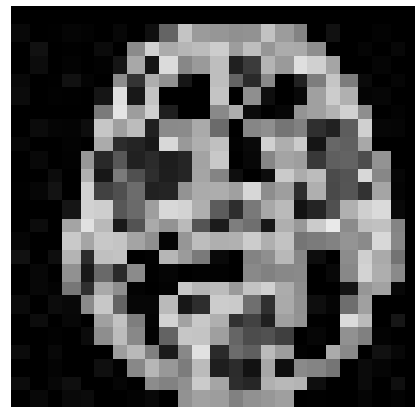
16x16



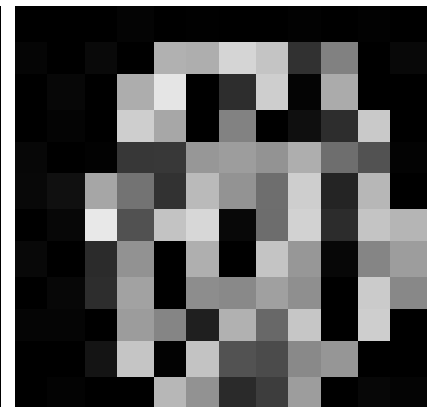
100x100



50x50



24x24



12x12

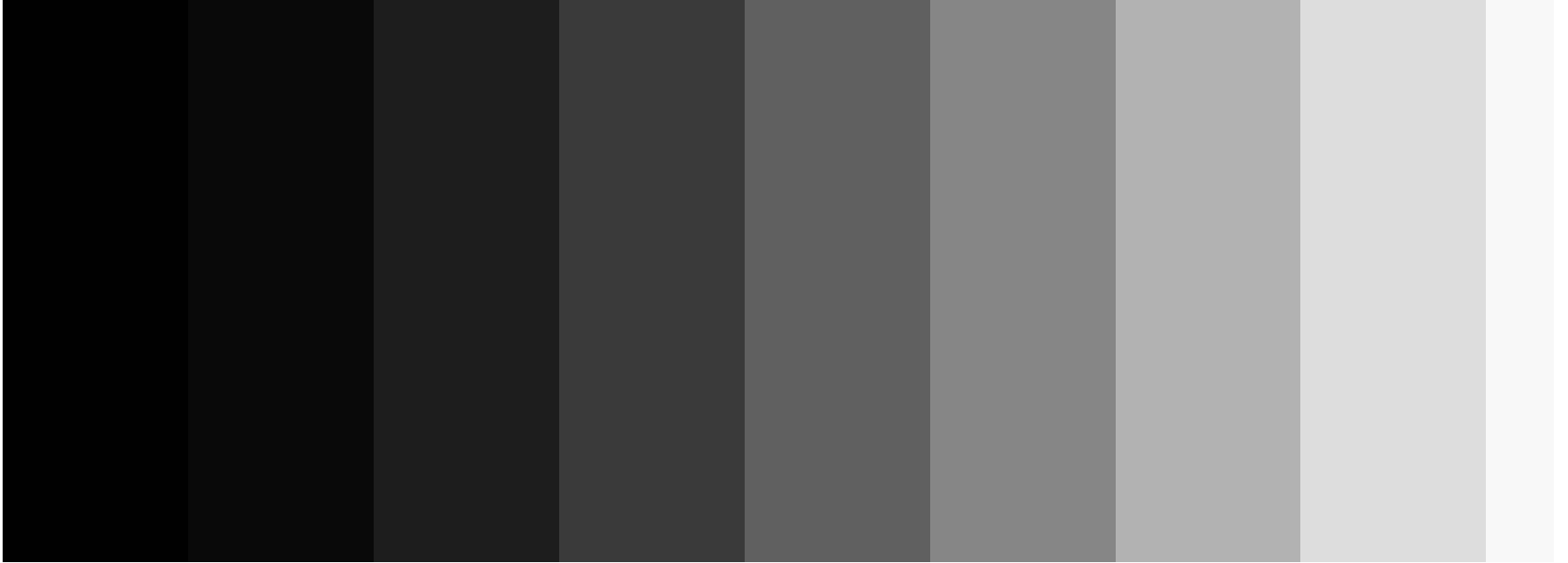
# Koyuluk (Contrast)

- ◆ Göz lokal etkilere göre görüntüyü farklı algılar.
- ◆ Kontrast görüntü oluşumunda ve gözün algılamasında önemli bir etkidir.

Aşağıda merkezdeki kare aynı olmasına rağmen göz onu farklı kareler gibi algılamaktadır.

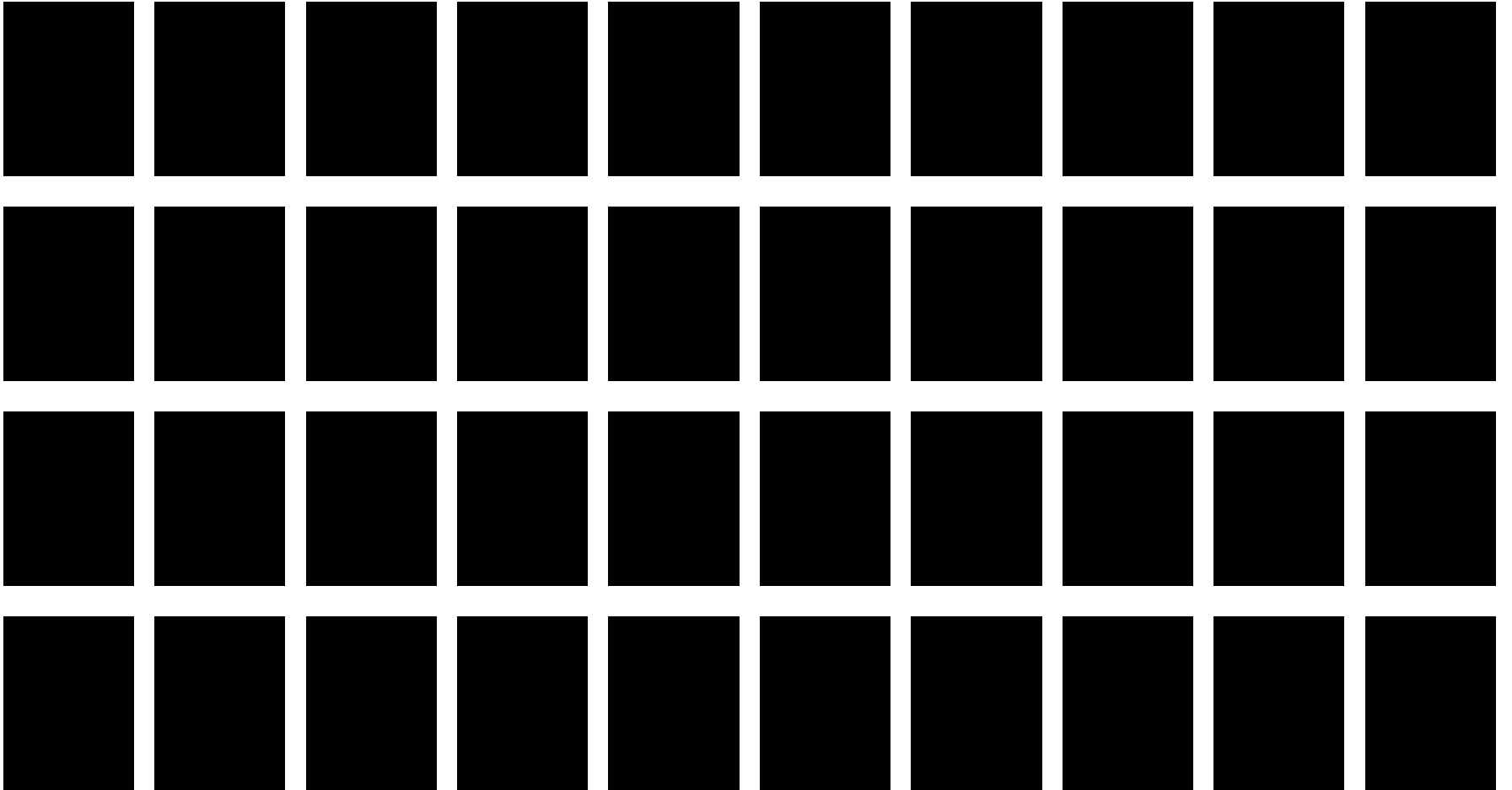


Aşağıda her bir dikdörtgensel bölge aynı renkte olmasına rağmen göz farklı renklerde gibi algılamaktadır.



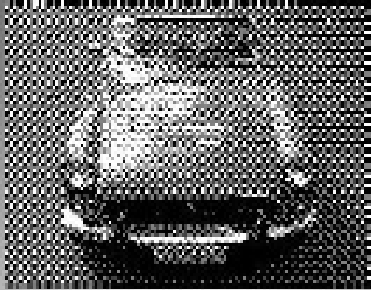
Each of the nine rectangles is a constant colour

Aşağıda göz aslında olmayan hayalet kareler  
görmektedir. (Ghost Squares)



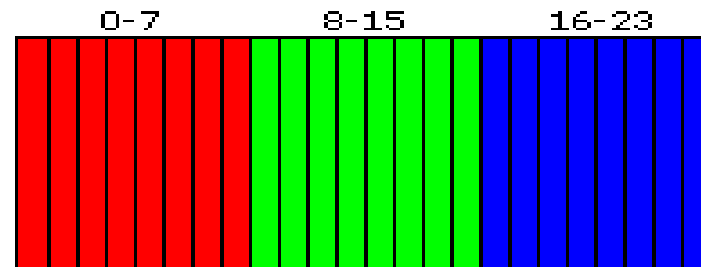
# İndekslenmiş Görüntüler

- ◆ Resimleri daha iyi ve verimli saklamak için diğerk bir yöntem ise indekslemedir.
- ◆ İndekslenmiş resimlerin bir renk paleti vardır.
- ◆ Bu palet 3 tane kolonu olan ve her satırın o bir rengin RGB değerini içerdđđi bir matristir.
- ◆ Dolayısıyla her piksel , bu tabloya bir indeks değeri içerir.



•Değişik renk paletleri kullanılarak aynı resim değişik şekillerde ifade edilir.

•Renkli resimlerde ayrı ayrı RGB değerlerinin tutulması yerine palet tutulması , bu bakımdan daha avantajlıdır.



**Bit Layout for 24-Bit Colour**



# RGB -> Gri tonları

- ◆ RGB değerleri, gri tonlarına şu formülle dönüştürülür:

$$I_{grey}(\mathbf{p}) = \frac{I_R(\mathbf{p}) + I_G(\mathbf{p}) + I_B(\mathbf{p})}{3}$$

# Gri Tonları -> İkili Resimler

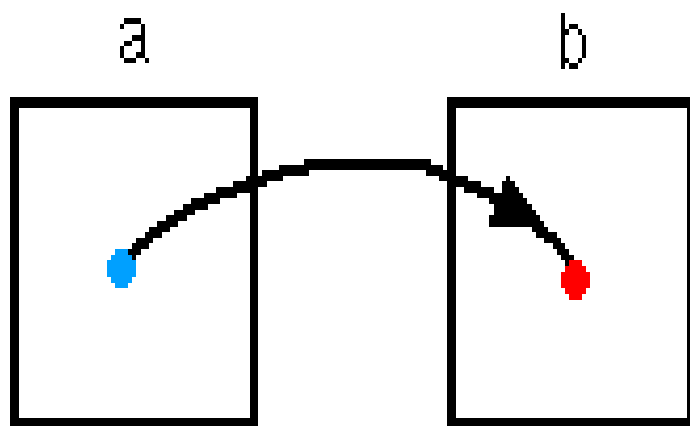
- ◆ Gri tonları ise ikili resimlere şu formülle dönüştürülür:

$$\mathbf{I}_{bin}(\mathbf{p}) = \begin{cases} 1 & \text{if } \mathbf{I}_{grey}(\mathbf{p}) \geq d \\ 0 & \text{otherwise} \end{cases}$$

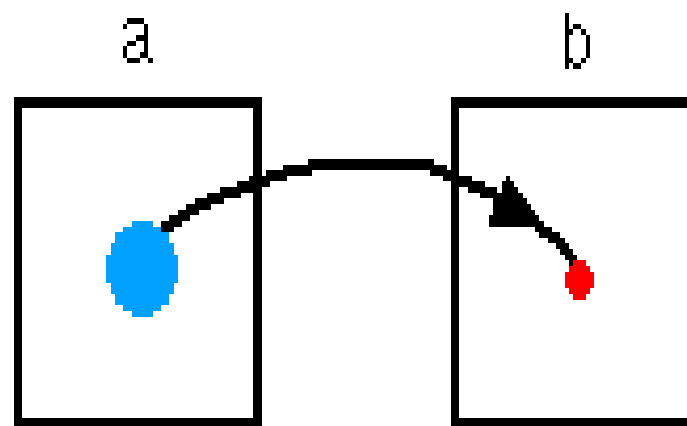
- ◆ Burada  $d$  belirli bir eşik değeridir ve bu değer, çevirim için ana noktadır. (**Threshold**)
- ◆ Eşik noktalarının kullanılması hesaplama işlemini kolaylaştırırken bilgiyi yeterli kullanmaması ve genellikle elle girilen bir değer oluşturması bir dezavantajdır.

# Görüntü Operasyonları

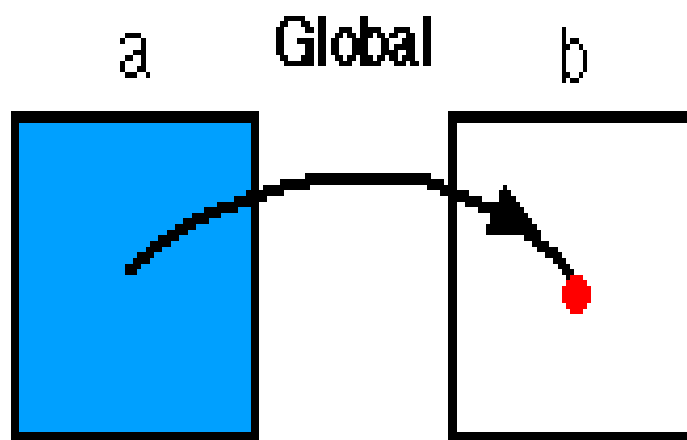
- ◆ Görüntü operasyonları ,  $a[m,n]$  gibi bir girdi görüntüsünü,  $b[m,n]$  gibi bir çıktı görüntüsüne çevirir.
- ◆ Operasyonlar,
  - Nokta
  - Yerel
  - Globalolmak üzere 3 sınıfta toplanırlar.



Point



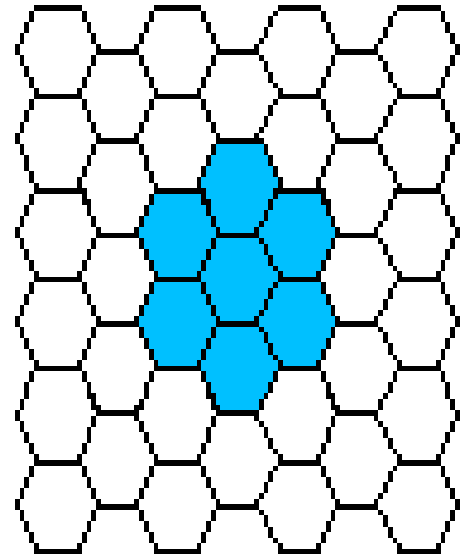
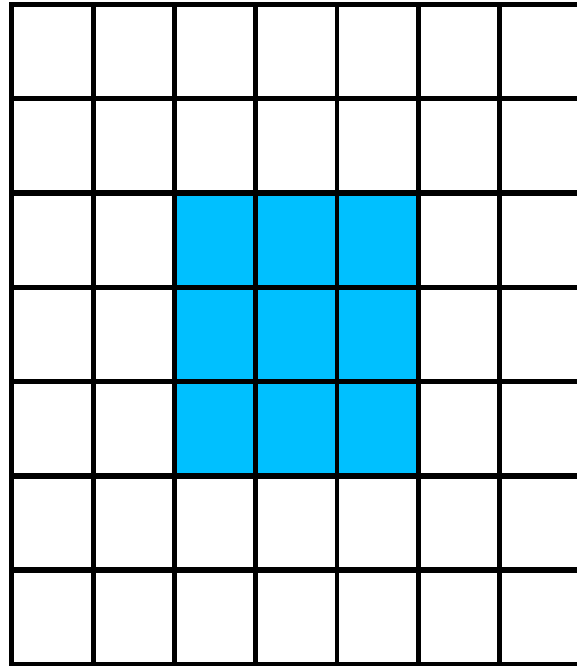
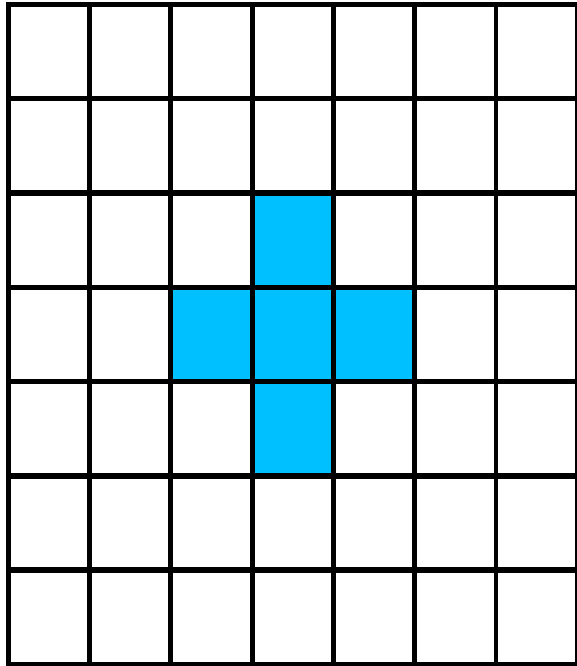
Local



● =  $[m=m_0, n=n_0]$

# Komşuluk İlişkileri

- ◆ Sayısal görüntü işlemede komşuluk ilişkileri çok önemlidir.
- ◆ Komşuluk ilişkileri ise ancak görüntü örnekleme yapılarak elde edilebilir. Temel örnekleme yöntemleri şunlardır:
  - *Dikdörtgensel örneklemede* resmin üzerinde dikdörtgensel bir ızgara olduğu düşünülür.
  - *Altıgensel örneklemede* ise resmin altıgenlerden oluşmuş parçalar içerdiği düşünülür.



# Fitreleme

- ◆ Fitreleme resmin üzerinde bir filtre varmış gibi düşünüp her piksel değerinin yeniden hesaplanmasıdır.
- ◆ Filtreler sayesinde girdi resminden yeni resim değişik efektler verilerek elde edilir. Fitreleme işlemi şu formülle elde edilebilir:

$$f'(x, y) = \sum_{i=-\infty}^{+\infty} \sum_{j=-\infty}^{+\infty} h(i, j) \times f(x - i, y - j)$$

Burada  $h$  fonksiyonu filtredir.

# Örnek Filtreler –1

Basic 3x3 blurring filter

$$\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix} = \frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Gaussian 3x3 blurring filter

$$\frac{1}{16} \times \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Gaussian 5x5 blurring filter

$$\frac{1}{112} \times \begin{bmatrix} 1 & 2 & 4 & 2 & 1 \\ 2 & 6 & 9 & 6 & 2 \\ 4 & 9 & 16 & 9 & 4 \\ 2 & 6 & 9 & 6 & 2 \\ 1 & 2 & 4 & 2 & 1 \end{bmatrix}$$



# Örnek Filtreler –2

**Horizontal**

1	1	1
0	0	0
-1	-1	-1

**Vertical**

1	0	-1
1	0	-1
1	0	-1

**Diagonal**

1	1	0
1	0	-1
0	-1	-1

Prewitt filters

1	0
0	-1

0	1
-1	0

Roberts filters

1	2	1
0	0	0
-1	-2	-1

1	0	-1
2	0	-2
1	0	-1

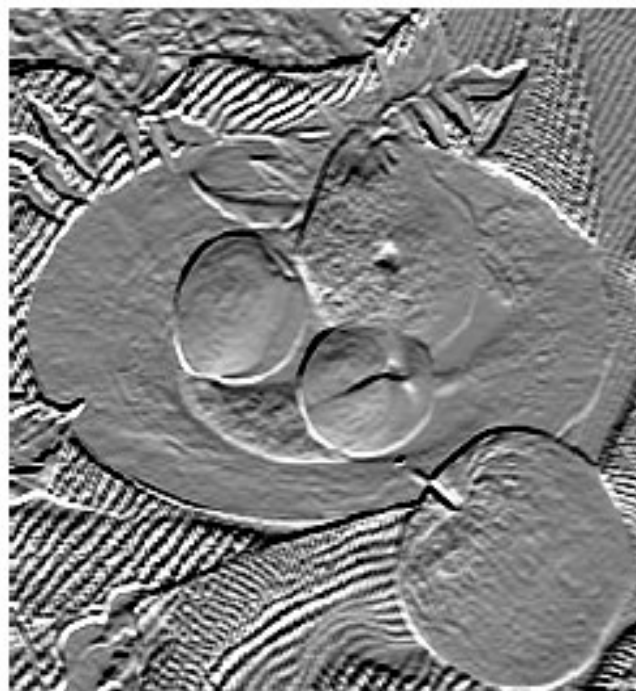
2	1	0
1	0	-1
0	-1	-2

Sobel filters





original image



after use of a  $3 \times 3$  Prewitt  
horizontal edge detection filter

*mid-grey = no edge, black or white = strong edge*

# Medyan Filtreleme

- ◆ Medyan filtreleme, lineer bir işlem değildir.
- ◆ Merkezi bizim pikselimiz olan bir bölgenin çevresindeki piksellerin medyanlarının alınmasına dayanır.

$\mathbf{I}_{med}(\mathbf{p})$  = median of  $m \times n$  region centred at  $\mathbf{I}(\mathbf{p})$ .

# Örnek- Medyan Filtreleme

e.g. 3x3 median filter

10	15	17	21	24	27
12	16	20	25	99	37
15	22	23	25	38	42
18	37	36	39	40	44
34	2	40	41	43	47

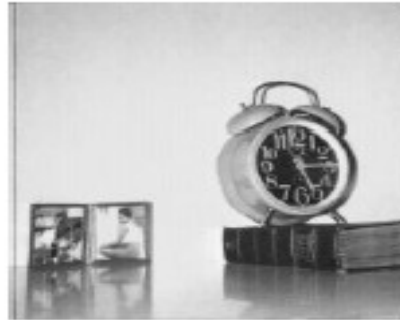
(16,20,22,23,  
25,  
25,36,37,39)

	16	21	24	27	
	20	25	36	39	
	23	36	39	41	

sort into order and take median

# Medyan Filtreleme Yararları

- ◆ Gürültü eklenmiş bir resim eski haline bu metoda döndürülebilir.



Original image

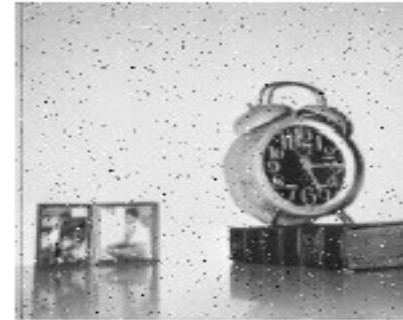
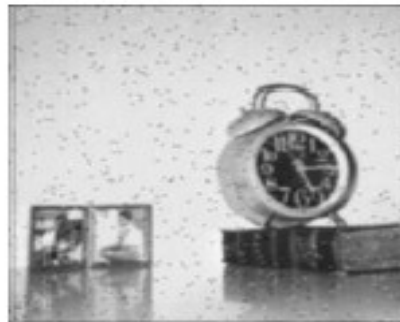


Image plus salt  
& pepper noise



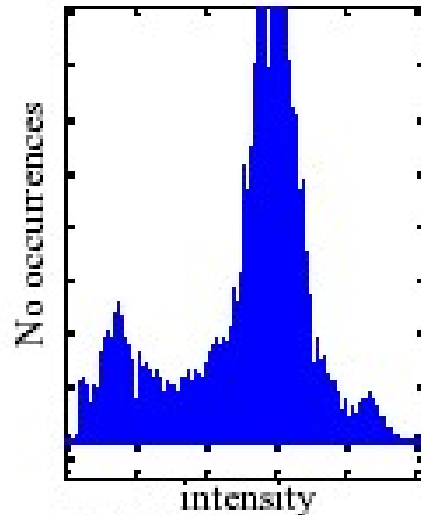
Local averaging,  
Across  $3 \times 3$  regions



Median filtering  
with  $3 \times 3$  kernel

# Histogram

- ◆ Görüntünün içerdığı piksel değerlerinin ağırlığını belirten grafiksel bir gösterimdir.



Original Histogram



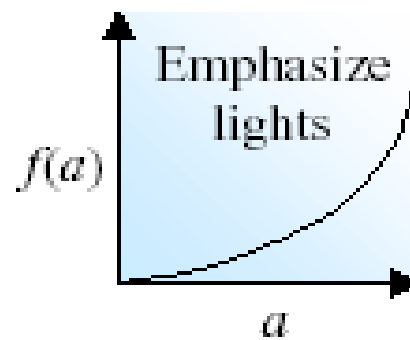
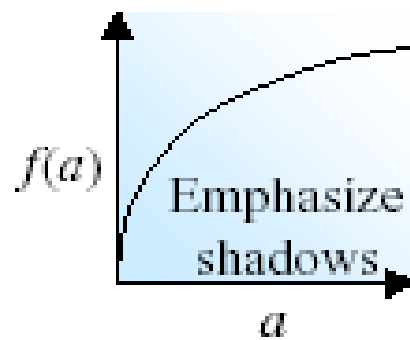
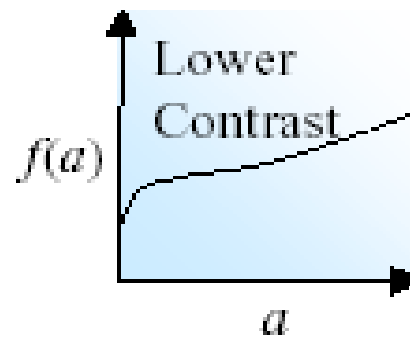
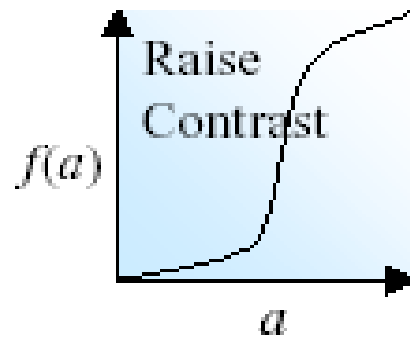
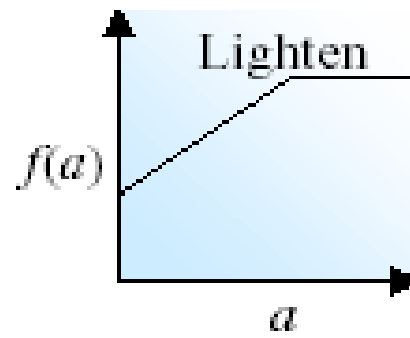
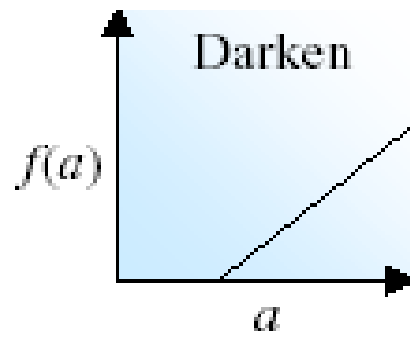
Original Image

# Homojen Nokta İşlemleri

- ◆ En basit görüntü işlemleri , o noktanın belirli bir fonksiyondan geçirilerek yeni nokta değerlerinin bulunmasına dayanır.

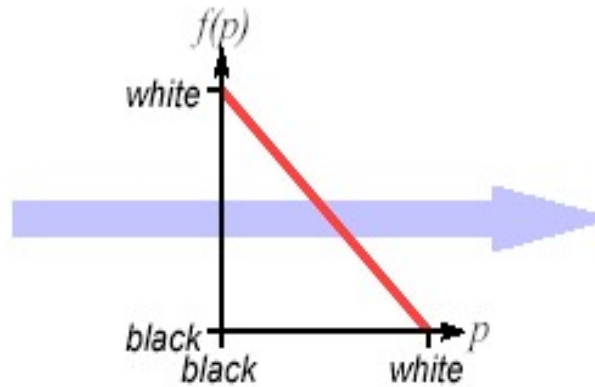
$$f: \mathbf{I}(\mathbf{p}) \rightarrow \mathbf{I}_{new}(\mathbf{p})$$





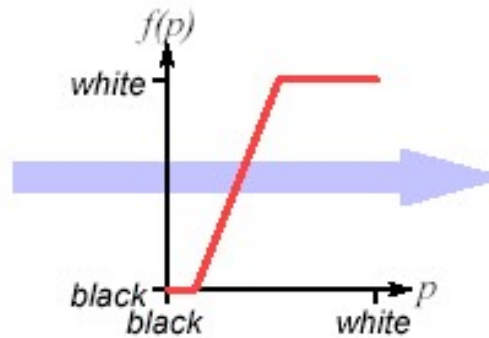
# Örnek Nokta İşlemi –1

- ◆ Bir resmin renklerini çevirmek:

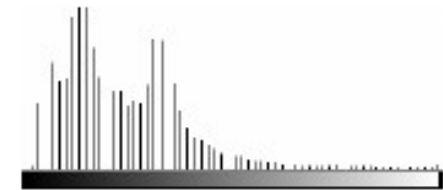


# Örnek Nokta İşlemi –2

- ◆ Bir resmin kontrastını ayarlamak:



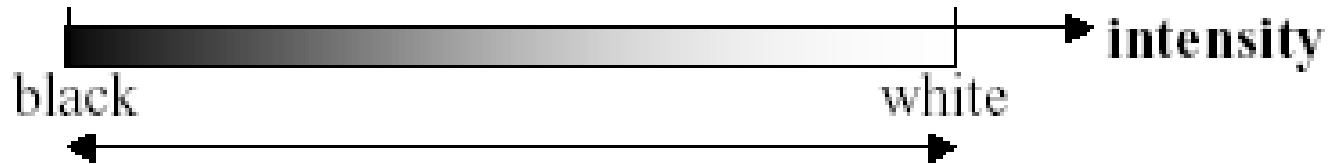
dark histogram



improved histogram

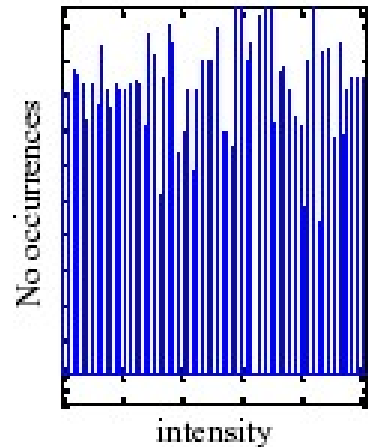
# Histogram Equalisation

- ◆ İnsan gözü , aşağıdaki gri düzeyleri geçişleri arasından yaklaşık 100 tane bölgeyi birbirinden ayırt edebilir



# Görüntünün Netleştirilmesi-1

- ◆ Şekilde birçok değer içeren dağınık bir piksel dağılımı vardır.
- ◆ Daha net bir görüntü için histogramın daha ayırık ve düzgün bir yapıya gelmesi sağlanmalıdır.



Equalised Histogram



Equalised Image

# Görüntünün Netleştirilmesi-2

- ◆ Bu işleme histogram eşitlemesi (Histogram Equalisation) denir. Bu işlem

$$f: I_1(\mathbf{p}) \rightarrow I_{eq}(\mathbf{p}) \quad f(a) = \sum_{r=1}^a \frac{n_1(r)}{N}$$

formülü ile ifade edilebilir.

- ◆ Burada  $n_1(r)$  fonksiyonu yoğunluğu  $r$  olan piksel sayısıdır ve  $N$  ise toplam piksel sayısıdır.



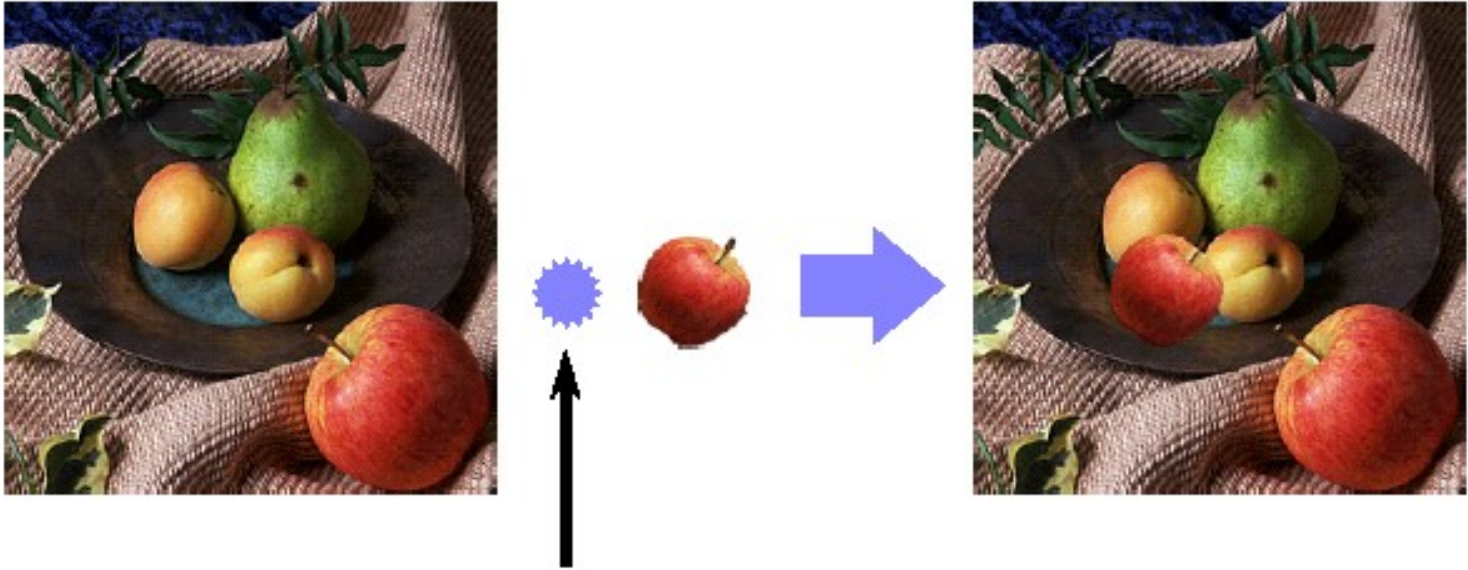
Original Image



Image after  
histogram equalisation

# İki Resmi Birleřtirmek

- ◆ İki resmi birleřtirmek için bir resimdeki seilen pikseller diđerine kopyalanır.



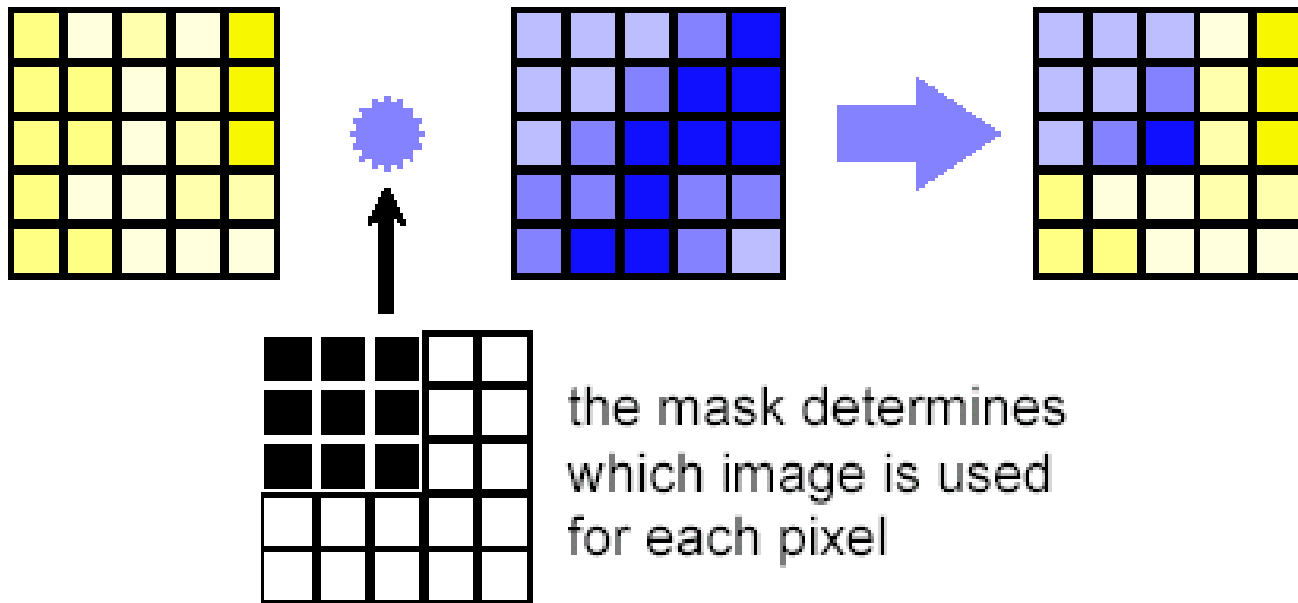
what does this operator do?



# Maskeleme -1

- ◆ Maskeleme , resmin istenen bölgesinin kesip çıkartılması ve elde edilmesidir.
- ◆ Şu formüle göre hesaplanır:

$$\mathbf{I}_{masked}(\mathbf{p}) = \mathbf{I}(\mathbf{p}) \cdot \mathbf{I}_{bin}(\mathbf{p})$$



Original image



Mask



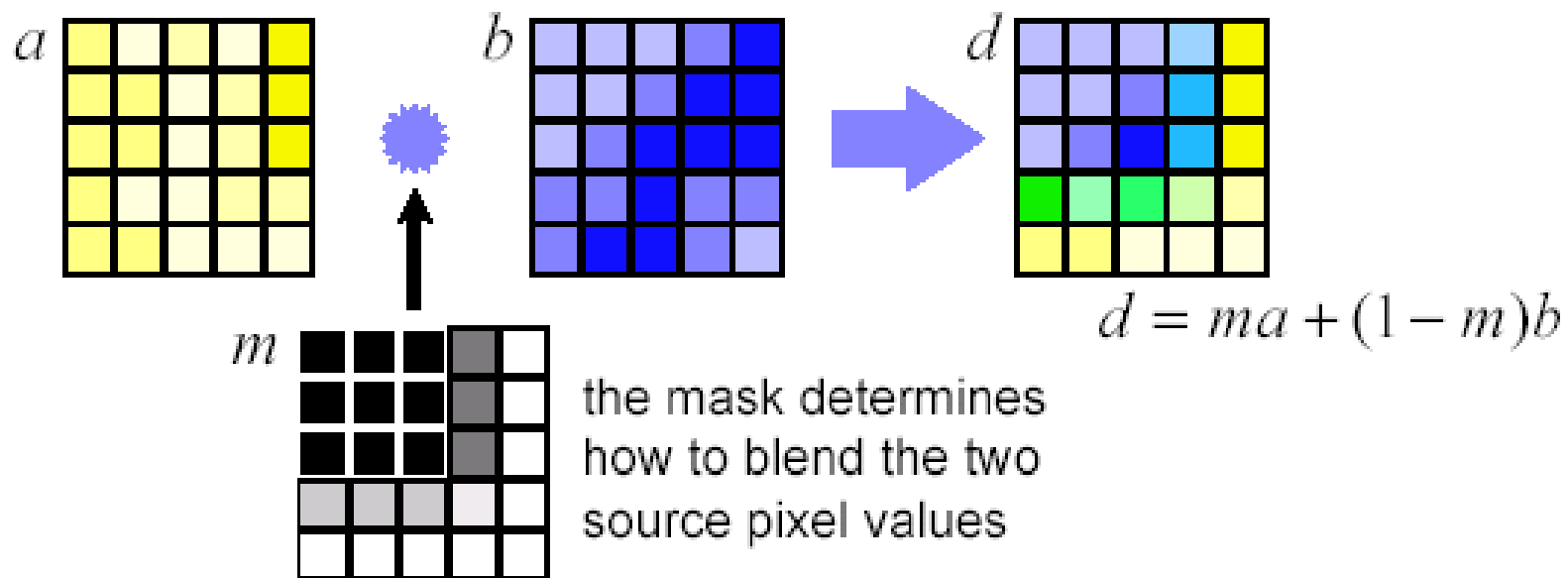
Product of image and mask



Background blurred

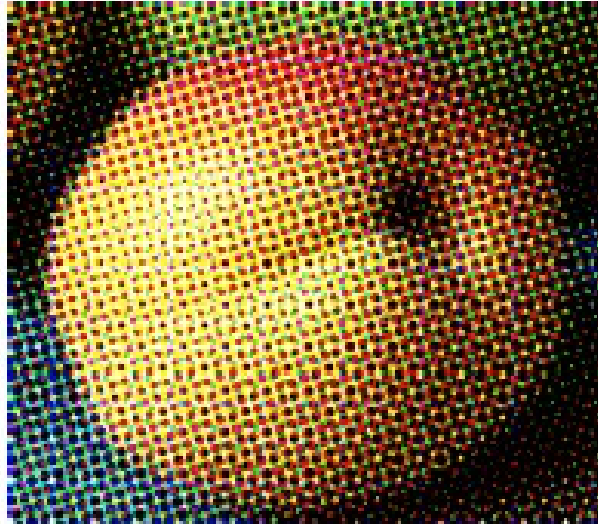
# Maskeleme –2

- ◆ Basit bir ikili maske yerine bazen alfa maskeleme tekniđi kullanılır.
- ◆ Bu teknikte her bir görüntünün piksellerinin hangi oranda karıştırılıp yeni piksel değeri hesaplanır.



# Halftoning & Dithering

- ◆ Gri tonlarında olan bir resmi ikili bir resme dönüştürmek için kullanılır.
- ◆ Ayrıca 4 adet resmin kullanıldığı print işleminde de kullanılır.



# Halftoning

- ◆ Her bir gri tonundaki resimin her bir pikseli ikili kare şeklinde piksellere dönüştürülebilir.
- ◆ Örneğin 5 adet yoğunluk değeri taşıyan pikseller, 2x2 boyutlarında pikseller ile ifade edilebilir.

## 1-to-4 pixel mapping



0-51

52-102

103-153

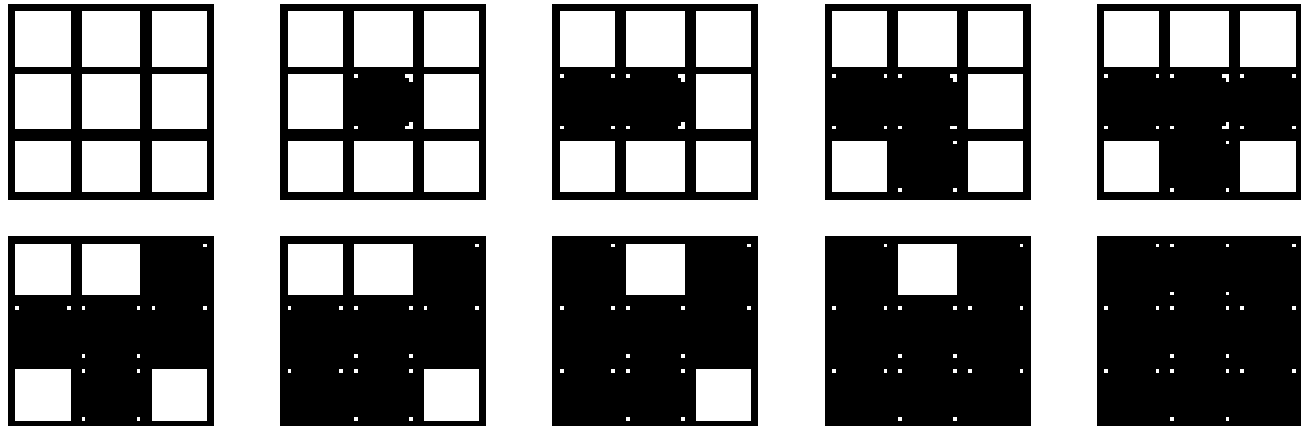
154-204

205-255

8-bit values that map to each of the five possibilities

# Halftoning Dither Matrisleri –1

- ◆ Diğer bir olasılık da 3x3 boyutunda desenler kullanmaktır.



# Halftoning Dither Matrisleri –2

- ◆ Bu desenler dither matrisi ile ifade edilebilir.

7	9	5
2	1	4
6	3	8

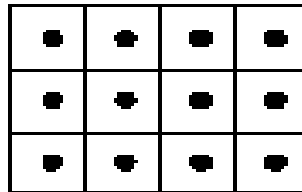
- Normal bir resim ve dither işleminden geçirilmiş resim aşağıda gösterilmiştir.



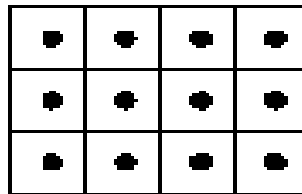
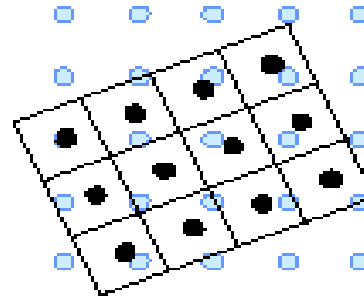


# Geometrik Dönüşümler

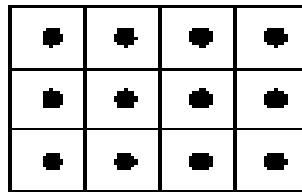
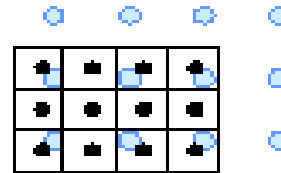
- ◆ Döndürme, oranlı büyütme ve x ve y eksenlerinde değişik oranlarda boyutlandırma işlemleri şu 2 aşama gerçekleştirilerek yapılır:
  - Eldeki piksel değerleri kullanılarak gerekli dönüşümler yapılır
  - Enterpolasyon kullanılarak dönüşüme uğramış resimin tamsayı piksel koordinatları bulunur.



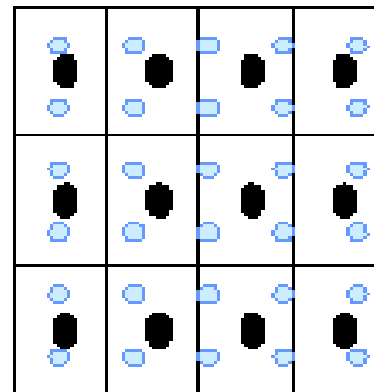
Rotation



Resizing  
(shrinking)



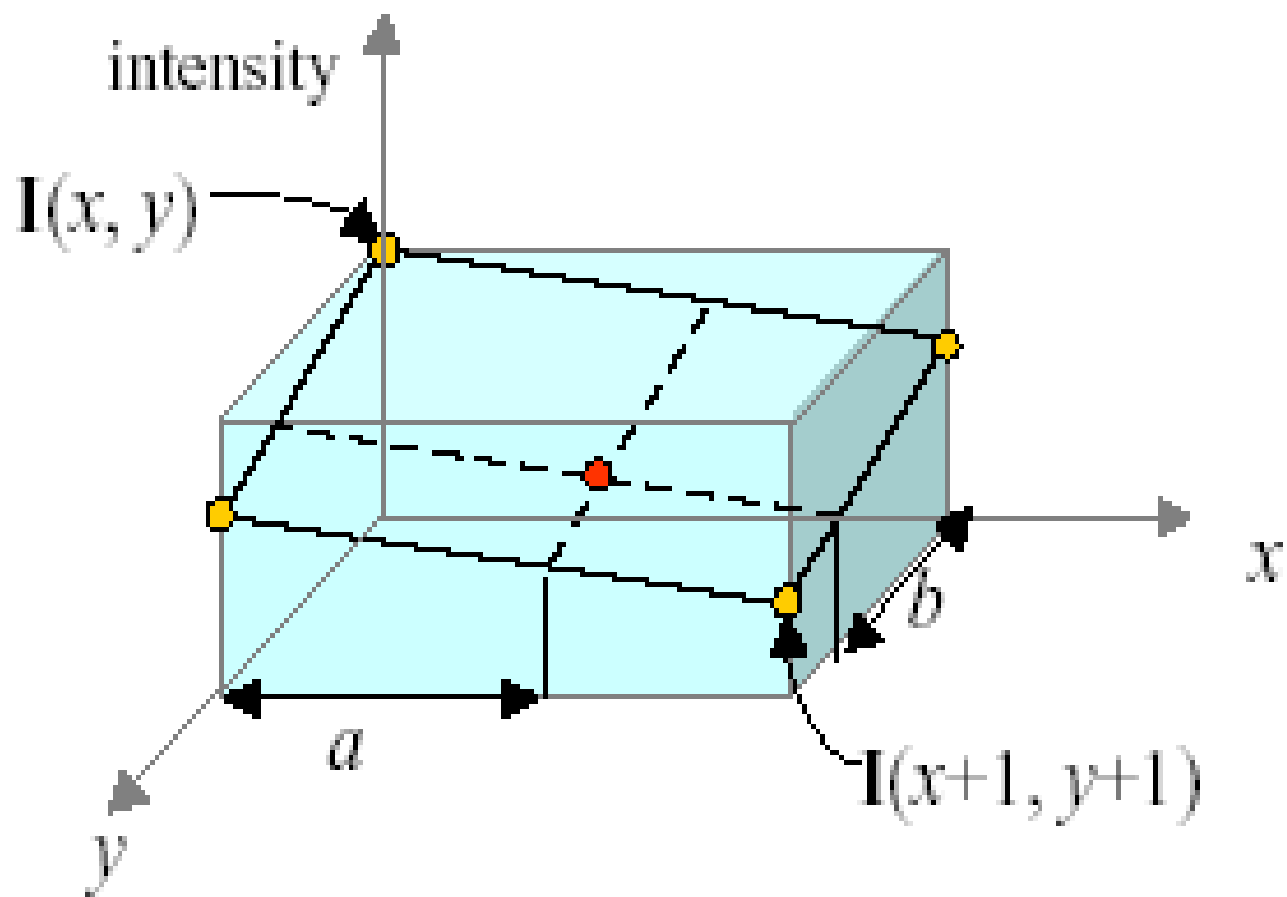
Resizing  
(enlarging each  
axis differently)



# Bilinear Enterpolasyon

- ◆  $I(x', y')$  noktasının yoğunluk değerinin,  $I(x, y)$ ,  $I(x+1, y)$ ,  $I(x, y+1)$ ,  $I(x+1, y+1)$  kullanılarak belirlenmesidir.

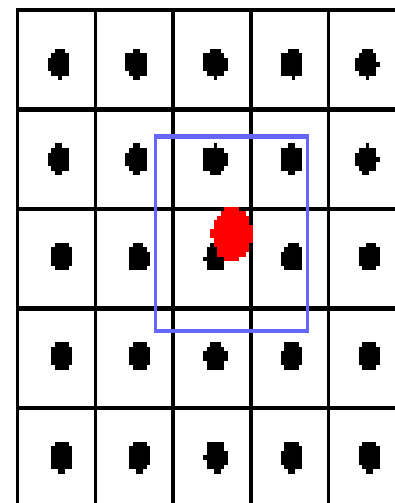
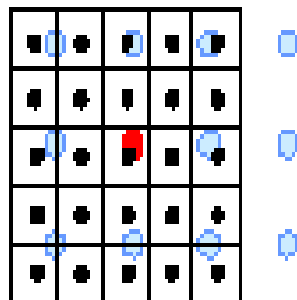
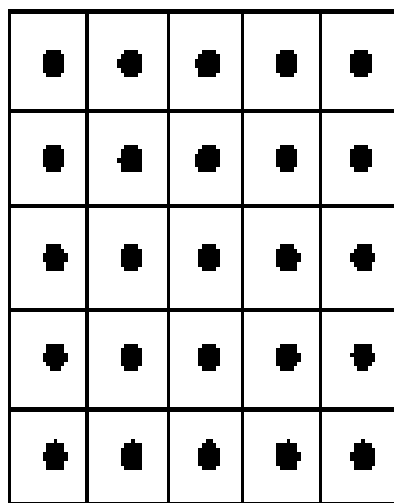
$$I(x', y') = \begin{bmatrix} 1-b & b \end{bmatrix} \begin{bmatrix} I(x, y) & I(x+1, y) \\ I(x, y+1) & I(x+1, y+1) \end{bmatrix} \begin{bmatrix} 1-a \\ a \end{bmatrix}$$



# Boyutlandırma

- ◆ Önce boyutlandırma matrisinden geçirilerek yeni piksel değeri elde edilir
- ◆ Enterpolasyon yardımı ile yeni tamsayı piksel elde edilir.
- ◆ Bu enterpolasyonda, bilinen komşu pikseller kullanılabilir.

$$\mathbf{P}_{scaled} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \mathbf{P}$$



# Döndürme

- ◆ Yeni piksel değeri döndürme matrisi yardımı ile elde edilir
- ◆ Yeni tamsayı değeri örneğin bilinear enterpolasyon kullanılarak elde edilir.

$$\mathbf{p}_{rot} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \mathbf{p}$$

$$\mathbf{I}_{rot}(\mathbf{p}) = \begin{bmatrix} 1-b & b \\ b & 1-a \end{bmatrix} \begin{bmatrix} \mathbf{I}(x,y) & \mathbf{I}(x+1,y) \\ \mathbf{I}(x,y+1) & \mathbf{I}(x+1,y+1) \end{bmatrix} \begin{bmatrix} 1-a \\ a \end{bmatrix}$$

