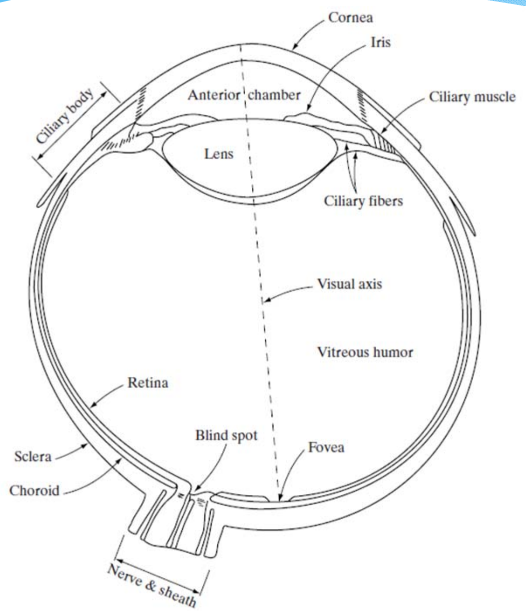
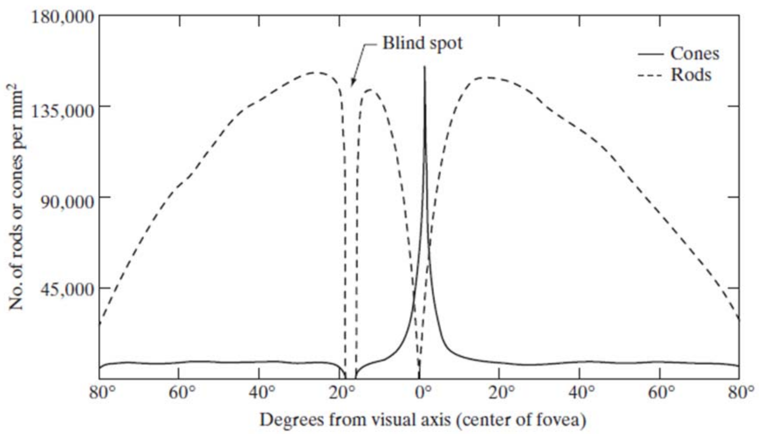
# 디지털 영상 기초

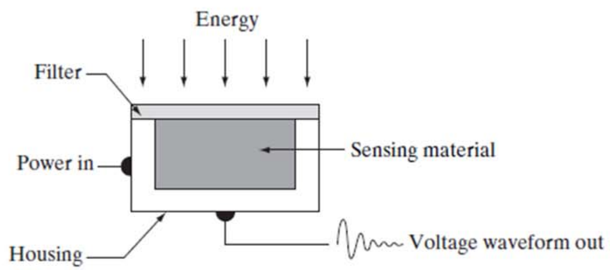
# 시각적 인지의 요소

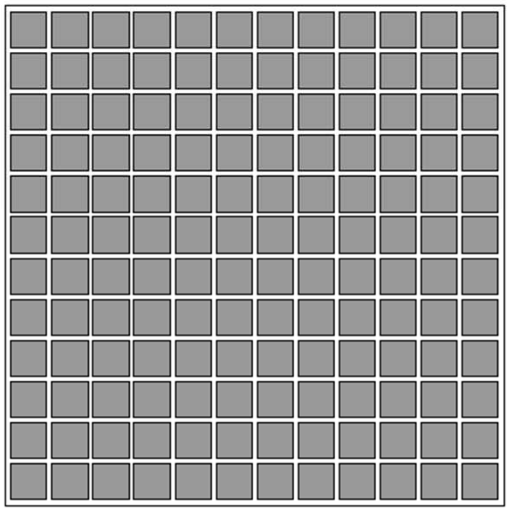


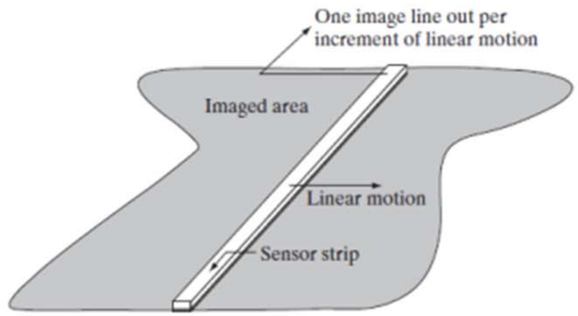




- 중심오목(Fovea)
  - ✓ 망막(Retina)의 중앙부
  - ✓ 1.5mmX1.5mm인 정사각형 센서 배열로 다룸
- 추상세포(Cones)
  - 컬러와 디테일을 인지
  - 중심오목에 주로 위치
  - 명소시(Photopic or bright-light Vision)
- 간상세포(Rods)
  - 시야의 전체적인 그림 제공
  - 낮은 레벨의 조명에 민감
  - 암소시(Scotopic or Dim-light Vision)

# 영상감지 및 획득


Energy
Filter
Power in
Sensing material
Housing
Voltage waveform out




One image line out per
increment of linear motion
Imaged area
Linear motion
Sensor strip

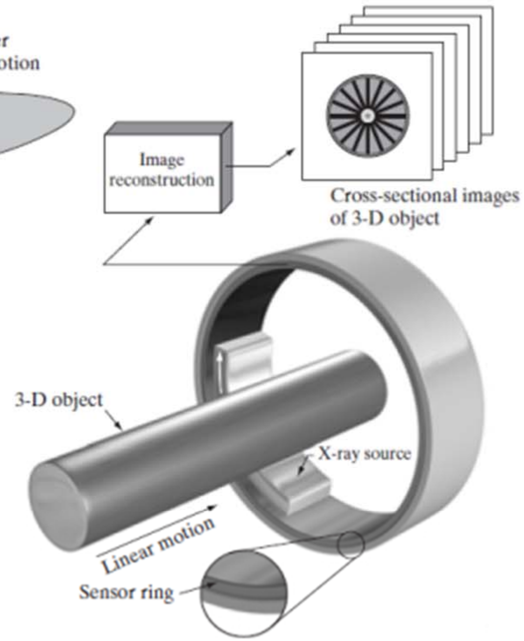
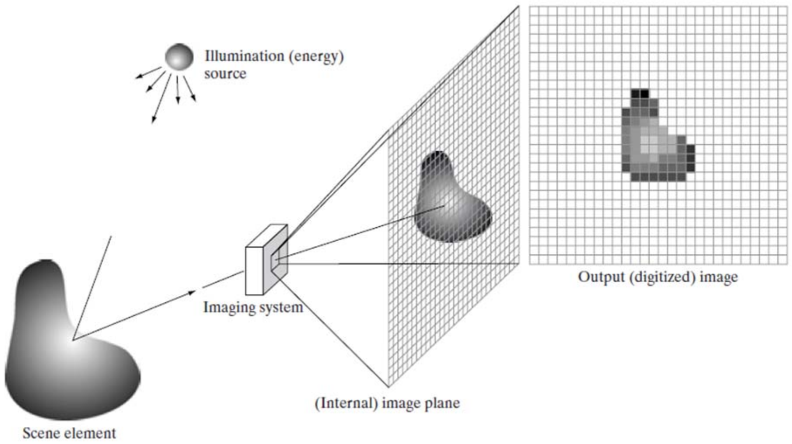


Image
reconstruction
Cross-sectional images
of 3-D object
3-D object
X-ray source
Linear motion
Sensor ring

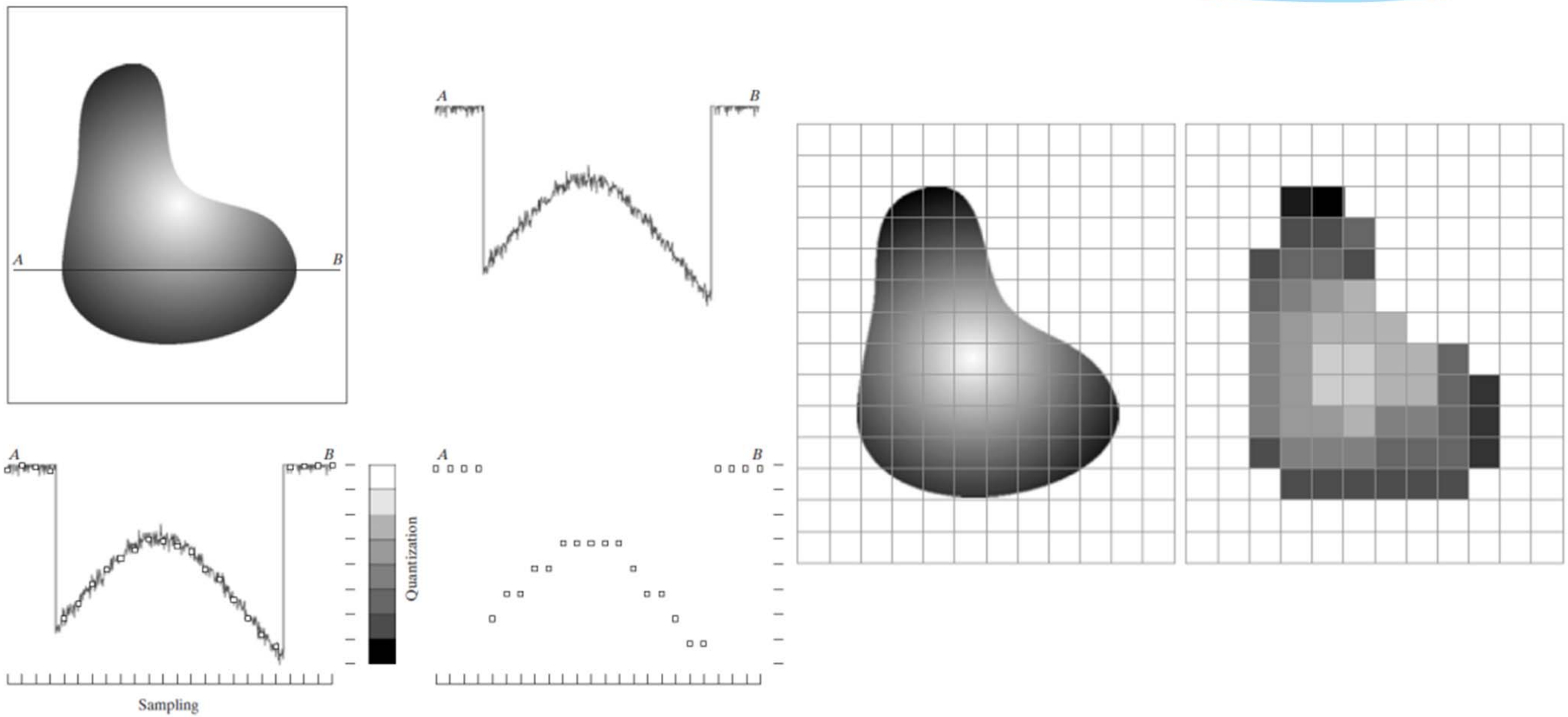
# 간단한 영상 형성 모델



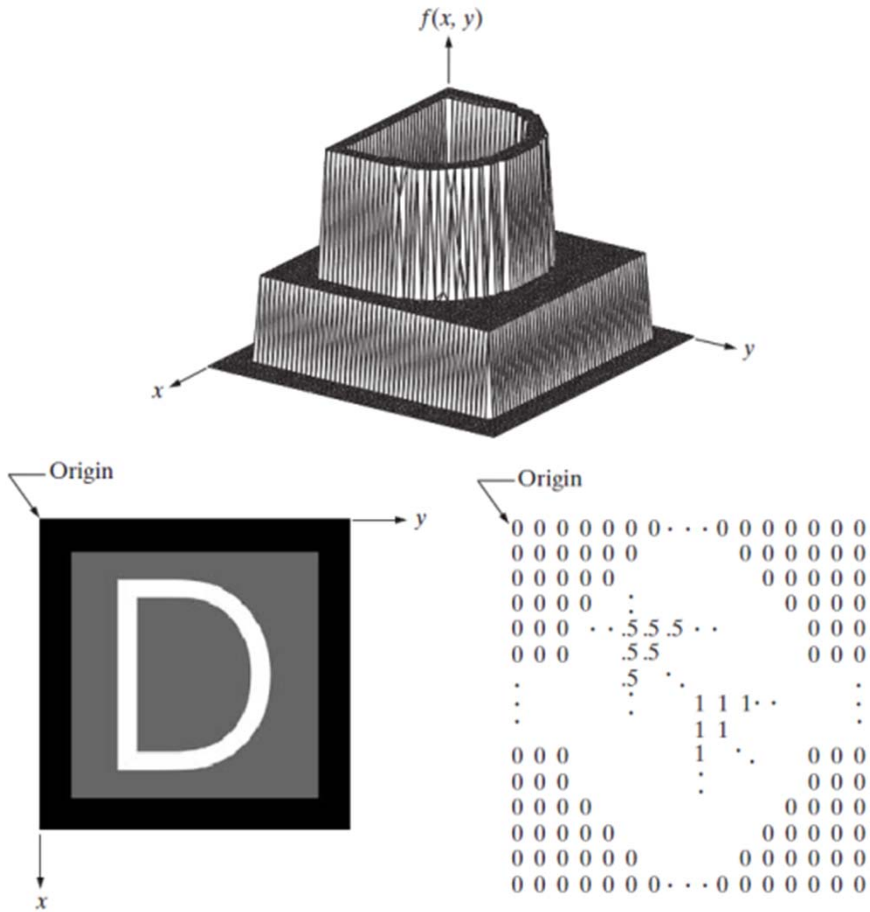

$$\ell = f(x_0, y_0)$$
$$\mathcal{L}_{min} \le \ell \le \mathcal{L}_{max}$$
$$[\mathcal{L}_{min}, \mathcal{L}_{max}] \ : Gray\ Scale$$
$$[0, \mathcal{L} - 1]$$

* 영상을 $f(x, y)$ 형태의 2-D함수로 표기
* 진폭은 광원에 의해 결정되는 스칼라량
* $0 < f(x, y) < \infty$
* $f(x, y) = i(x, y) r(x, y)$
* $0 < i(x, y) < \infty$; 조명성분(*illumination*)
* $0 < r(x, y) < \infty$; 반사성분(*reflectance*) *or* 투과성분(*transmissivity*)

# 셈플링과 양자화



# 디지털 영상 표현




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

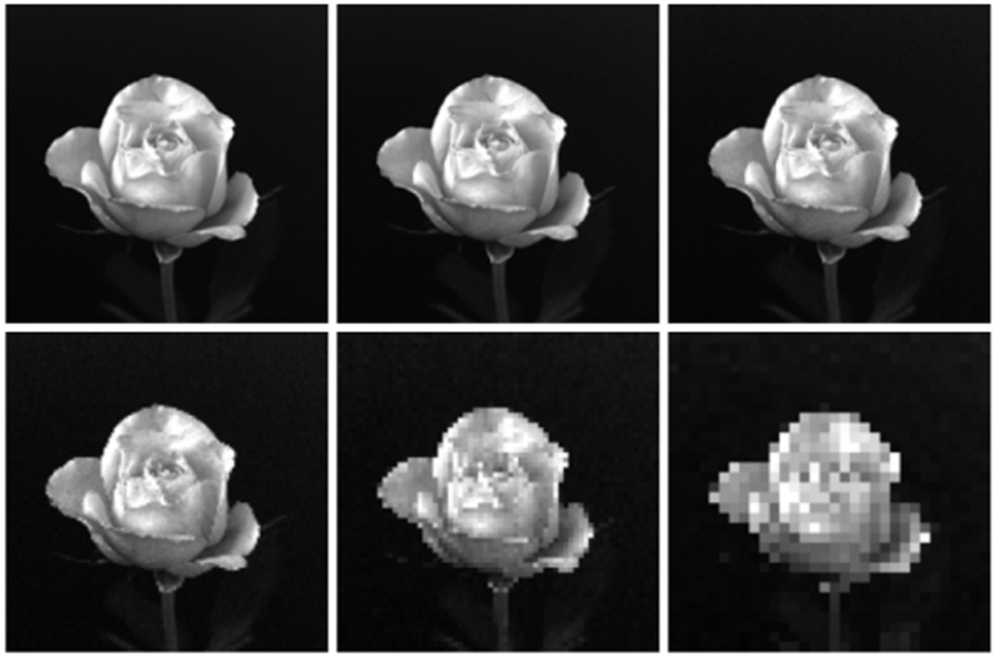
$$\mathbf{A} = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix}$$

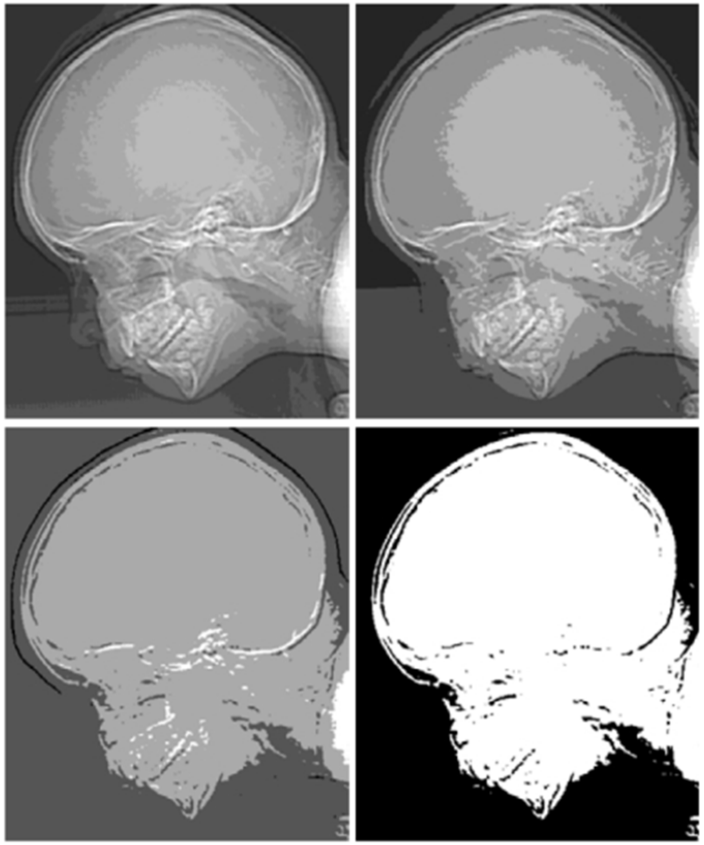
$$\mathcal{L} = 2^k$$

$$b = M \times \mathrm{N} \times \mathrm{k}$$

# 공간 밝기 해상도

공간 해상도(Spatial Resolution) : 디지털 영상이 몇 개의 화소(pixel)로 표현되었는가?
밝기 해상도(Intensity Resolution) : 디지털 영상이 몇 개의 명암 단계를 가지고 있는가?

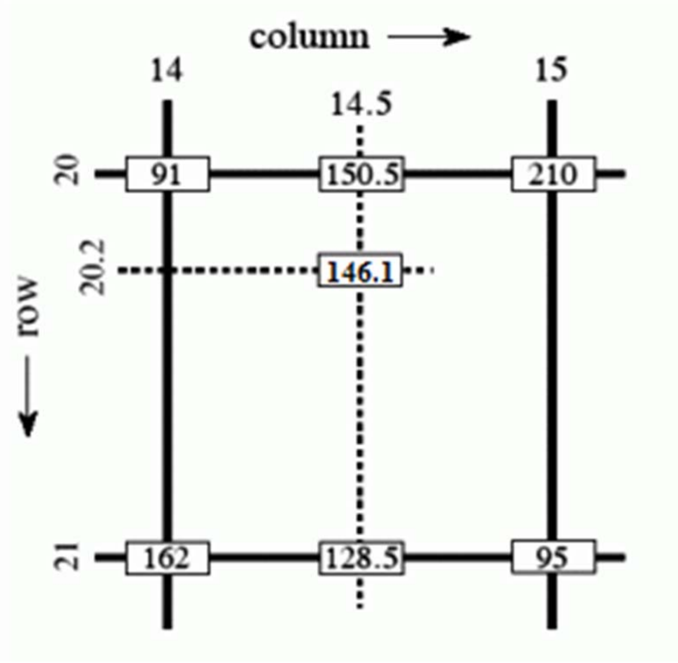




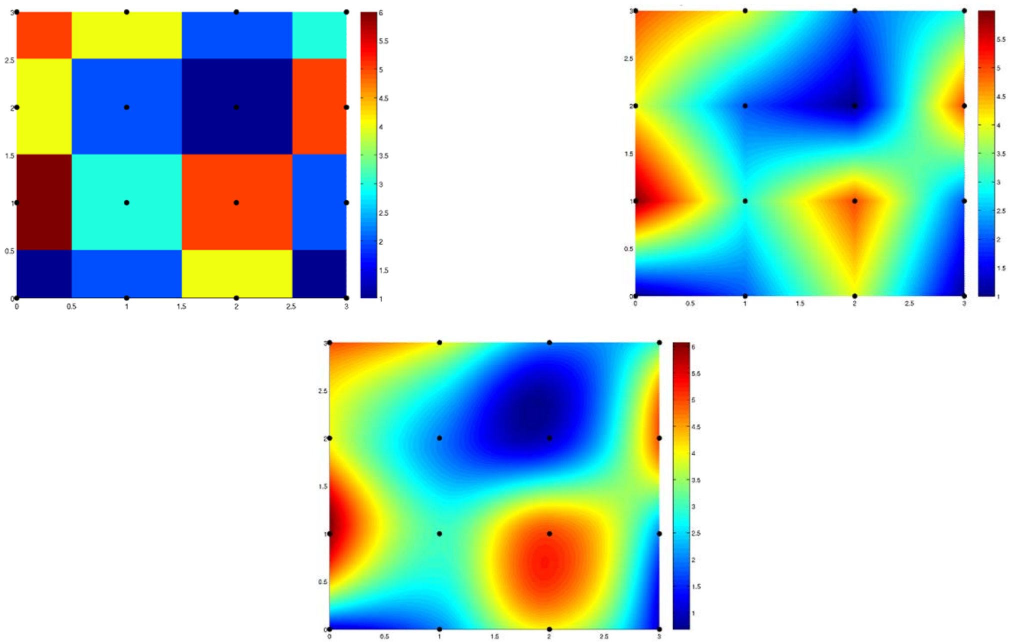
# 영상 보간법

보간법(*interpolation*) : using known data to estimate values at unknown location

- Bilinear Interpolation
- $v(x, y) = ax + by + cxy + d$




- Bicubic Interpolation
- $v(x, y) = \sum_{i=0}^{3} \sum_{j=0}^{3} a_{ij} x^i y^j$



# 인접성, 연결성, 영역, 경계

- $N_4(p) = \{(x+1,y),(x-1,y),(x,y+1),(x,y-1)\}$
- $N_D(p) = \{(x+1,y+1),(x+1,y-1),(x,-1\ y+1),(x-1,y-1)\}$

➢ *4-adjacency*. Two pixels $p$ and $q$ with values from *V* are 4-adjacent if q is in the set $N_4(p)$.

➢ *8-adjacency*. Two pixels $p$ and $q$ with values from *V* are 8-adjacent if q is in the set $N_8(p)$.

➢ *m-adjacency* (mixed adjacency).Two pixels $p$ and $q$ with values from *V* are m-adjacent if

1. $q$ is in $N_4(p)$, or
2. $q$ is in $N_D(p)$ *and* the set $N_4(p) \cap N_4(q)$ has no pixels whose values are from V.

a. $D(p,q) \geq 0\ (D(p,q) = 0\ iff\ p = q)$

b. $D(p,q) = D(q,p)$

c. $D(p,z) \leq D(p,q) + D(p,z)$

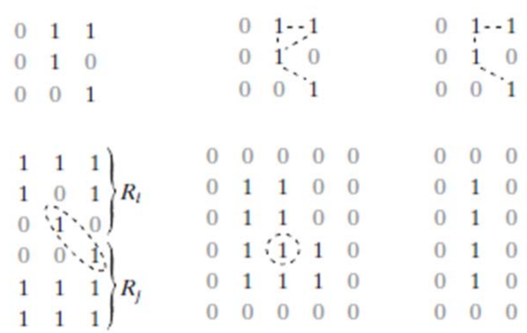
$$\begin{array}{ccccc} & & 2 & & \\ & 2 & 1 & 2 & \\ 2 & 1 & 0 & 1 & 2 \\ & 2 & 1 & 2 & \\ & & 2 & & \end{array}$$

a. $D_e(p,q) = [(x-s)^2 + (y-t)^2]^{\frac{1}{2}}$

b. $D_4(p,q) = |x-s| + |y-t|$ ; City-block

c. $D_8(p,q) = \max(|x-s|,|y-t|)$ ; Chessboard

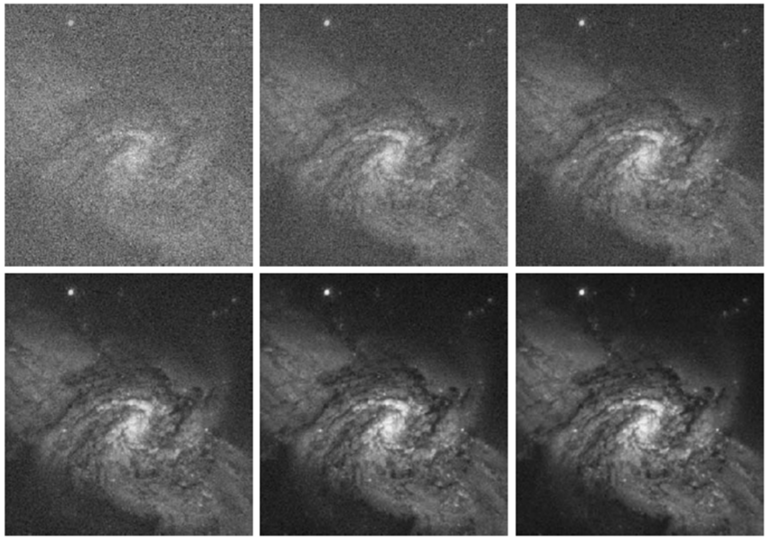
$$\begin{array}{ccccc} 2 & 2 & 2 & 2 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 \\ 2 & 1 & 1 & 1 & 2 \\ 2 & 2 & 2 & 2 & 2 \end{array}$$

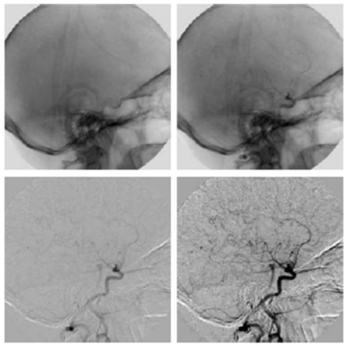


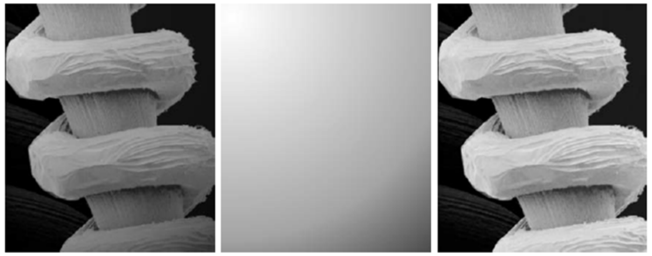
# 산술연산

$s(x,y) = f(x,y) + g(x,y)$ ; 노이즈 축소를 위한 영상의 덧셈(평균화)
$d(x,y) = f(x,y) - g(x,y)$ ; 차이 개선을 위한 영상의 뺄셈
$p(x,y) = f(x,y) \times g(x,y)$ ; *음영보정*, ROI
$v(x,y) = f(x,y) \div g(x,y)$

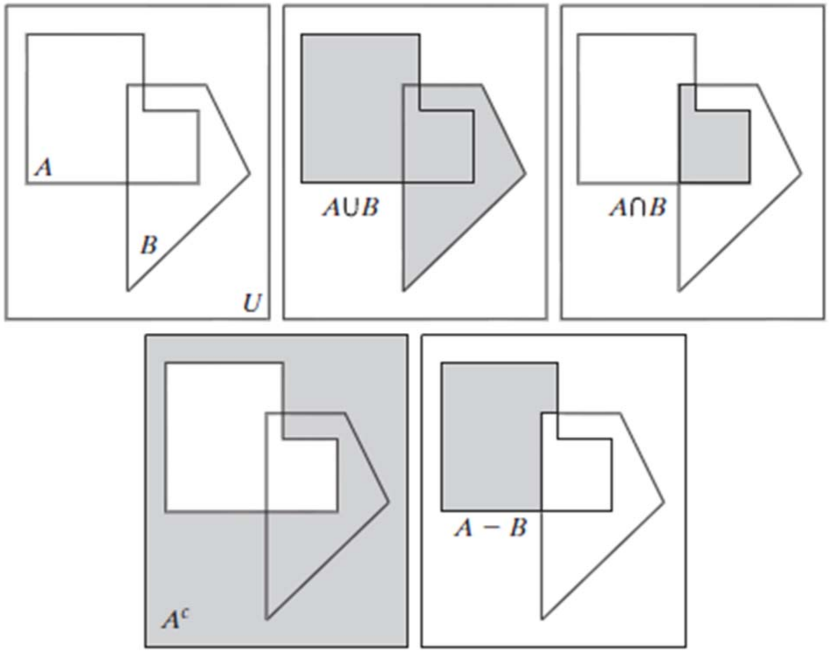
$f_m = f - \min(f)$ , $f_s = K[f_m / \max(f_m)]$ ; scaling



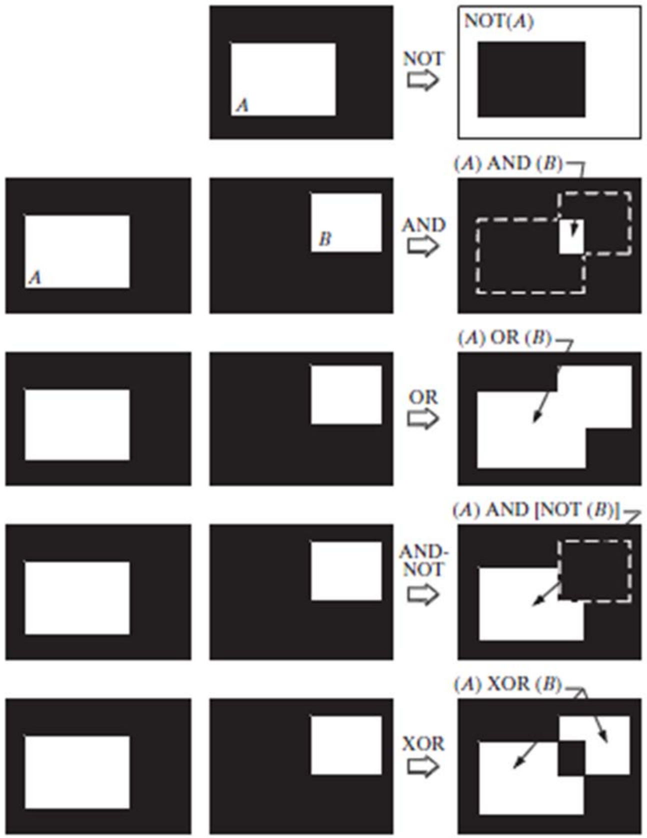




# 집합 연산과 논리 연산



A
B
U
A∪B
A∩B
A^c
A − B

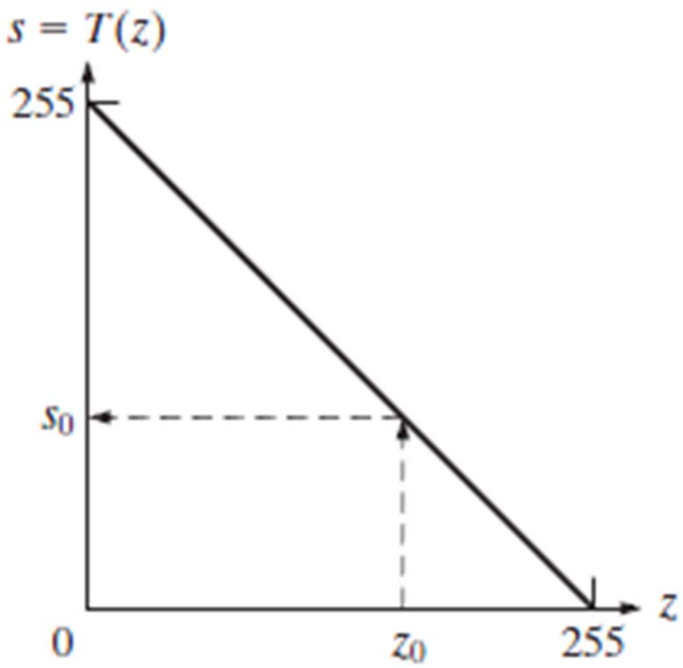



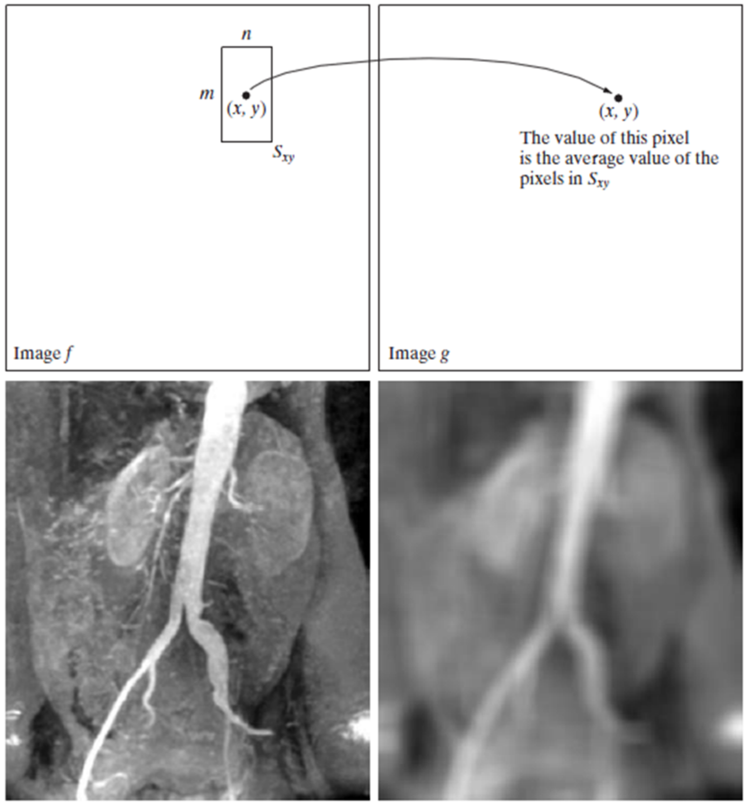
A
NOT
NOT(A)
(A) AND (B)
B
AND
(A) OR (B)
OR
(A) AND [NOT (B)]
AND-NOT
(A) XOR (B)
XOR

# 공간연산

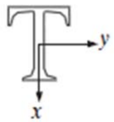
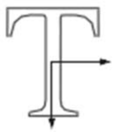
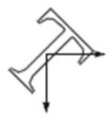
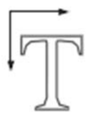
단일 화소 연산 : s = T(z)

이웃 연산(평균) : $g(x, y) = \frac{1}{mn}\sum_{(r,c)\in s_{xy}} f(r, c)$



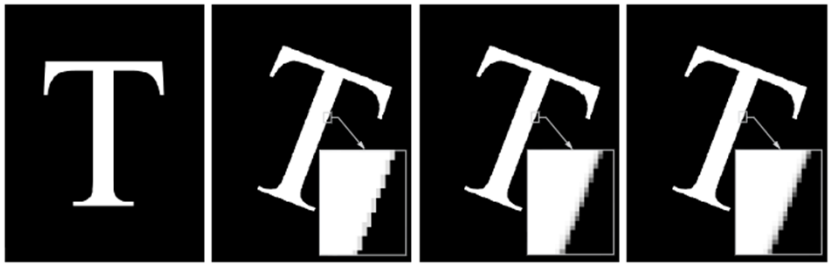
# 기하적 공간 변환

| Transformation Name | Affine Matrix, T | Coordinate Equations | Example |
|---|---|---|---|
| Identity | $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ | $x = v$<br>$y = w$ |  |
| Scaling | $\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$ | $x = c_x v$<br>$y = c_y w$ |  |
| Rotation | $\begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$ | $x = v\cos\theta - w\sin\theta$<br>$y = v\cos\theta + w\sin\theta$ |  |
| Translation | $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ t_x & t_y & 1 \end{bmatrix}$ | $x = v + t_x$<br>$y = w + t_y$ |  |
| Shear (vertical) | $\begin{bmatrix} 1 & 0 & 0 \\ s_v & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ | $x = v + s_v w$<br>$y = w$ |  |
| Shear (horizontal) | $\begin{bmatrix} 1 & s_h & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ | $x = v$<br>$y = s_h v + w$ |  |

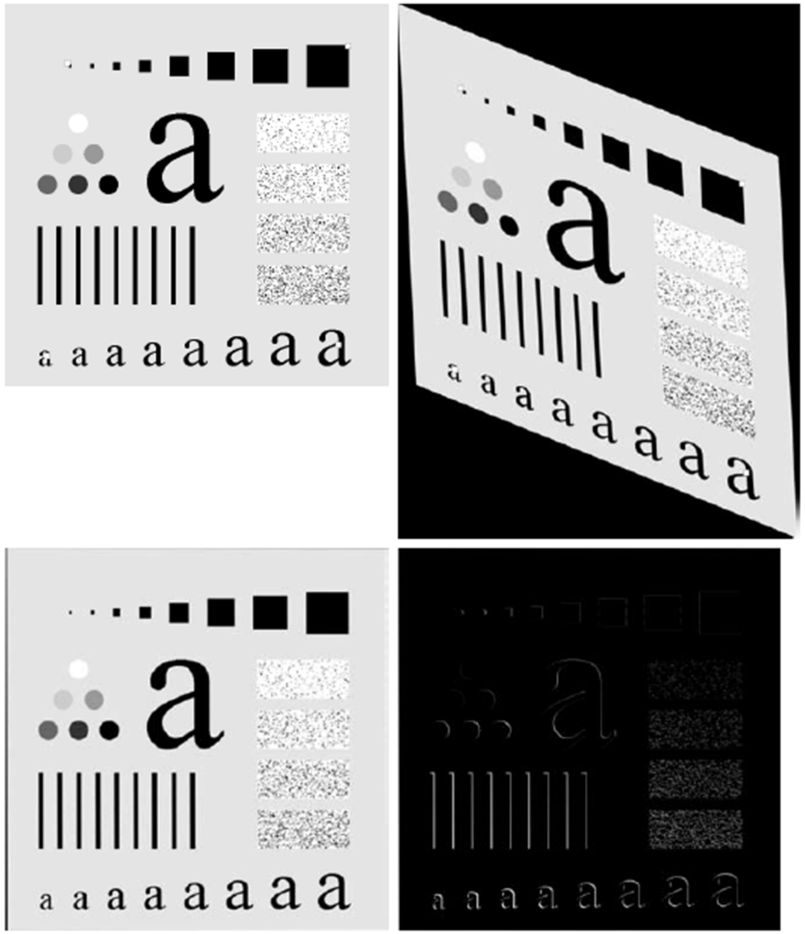
$$[x\ y\ 1] = [v\ w\ 1]T = [v\ w\ 1]\begin{bmatrix} t_{11} & t_{12} & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & 1 \end{bmatrix}$$

$(x, y) = T\{(v, w)\}$ ; 순방향 매핑

$(v, w) = T^{-1}(x, y)$ ; 역방향 매핑



# 영상 정합



$$x = c_1v + c_2w + c_3vw + c_4$$
$$y = c_5v + c_6w + c_7vw + c_8$$

# 백터 연산과 매트릭스 연산

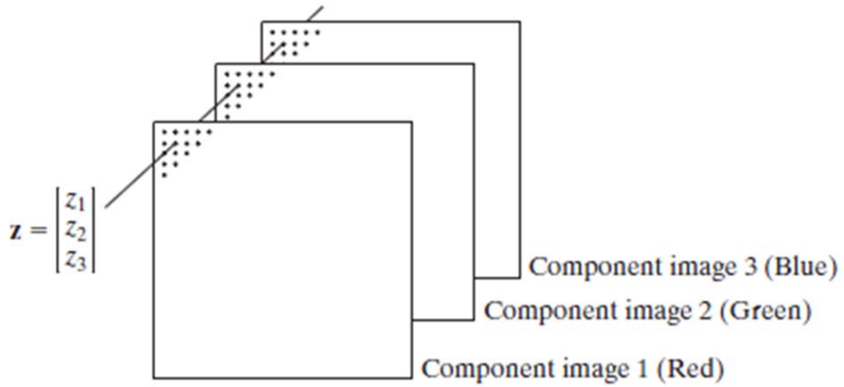
$$z = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \end{bmatrix}$$

$w = A(z - a)$ ; 화소백터의 선형변환

$g = Hf + n$ ; 광범위한 영상처리의 일반식

$$D(z, a) = \left[(z-a)^T(z-a)\right]^{\frac{1}{2}} = \left[(z_1 - a_1)^2 + (z_2 - a_2)^2 + \cdots + (z_n - a_n)\right]^{\frac{1}{2}} = \|z - a\|$$

2-D Eiclid 거리의 일반화, Vector Norm



# 영상변환

$$T(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)r(x,y,u,v)$$

$$f(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} T(x,y)s(x,y,u,v)$$

$r(x,y,u,v)$; *Forward Transform Kernel*

$s(x,y,u,v)$; *Inverse transeform Kernel*

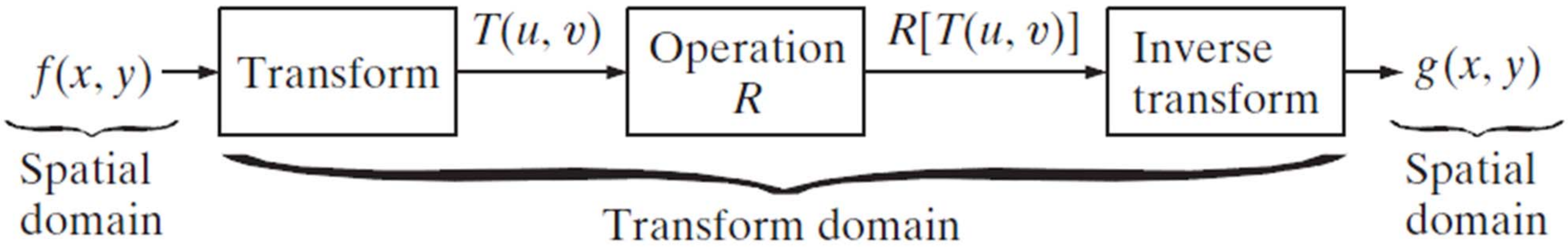
$x, y$ ; *Spatial Variables*

$u, v$ ; *Transform Variables*

$M, N$ ; *row and column dimensions*

분리 가능(separable) : $r(x,y,u,v) = r_1(x,u)r_2(y,v)$

대칭적(symmetric) : $r(x,y,u,v) = r_1(x,u)r_1(y,v)$



# 영상변환

$$T = AFA$$
$$BTB = BAFAB$$
$$F = BTB$$
$$\hat{F} = BAFAB$$

$T; T(u,v)$값을 갖는$M \times M$ 변환
F; f(x, y)의 $M \times M$ 매트릭스
$A; a_{ij} = r_1(i,j)$를 갖는$M \times M$ 매트릭스
$B$; 역변환 매트릭스

$$\mathrm{T}(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)r(x,y,u,v)$$

$$f(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} T(x,y)s(x,y,u,v)$$

참고 ; 2-D Fourier 변환

$$\mathrm{r}(\mathrm{x},\mathrm{y},\mathrm{u},\mathrm{v}) = \mathrm{e}^{-\mathrm{j}2\pi\left(\frac{\mathrm{ux}}{\mathrm{M}}+\frac{\mathrm{vy}}{\mathrm{N}}\right)}$$

$$s(x,y,u,v) = \frac{1}{MN}e^{j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

$$T(u,v) = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\mathrm{e}^{-\mathrm{j}2\pi\left(\frac{\mathrm{ux}}{\mathrm{M}}+\frac{\mathrm{vy}}{\mathrm{N}}\right)}$$

$$\mathrm{f}(\mathrm{x},\mathrm{y}) = \frac{1}{MN}\sum_{u=0}^{M-1}\sum_{v=0}^{N-1} T(u,v)e^{j2\pi\left(\frac{ux}{M}+\frac{vy}{N}\right)}$$

# 확률적 방법

계산 목적에는 유용하지만, 일반적으로 영상의 모양에 관해서는 알려주지 않음

$$p(n,k) = \frac{n_k}{MN}$$

$$\sum_{k=0}^{L-1} p(z_k) = 1$$

$$m = \sum_{k=0}^{L-1} z_k p(z_k)$$

$$\sigma^2 = \sum_{k=0}^{L-1} (z_k - m)^2 p(z_k)$$

$$\mu_n(z) \sum_{k=0}^{L-1} (z_k - m)^n p(z_k)$$