ECE 468 / CS 519:
Digital Image Processing

Interpolation, Intensity Transforms

Prof. Sinisa Todorovic

sinisa@eecs.oregonstate.edu

OSU Oregon State University

Spatial Image Transformations

Affine transforms:

- Translation
- Scaling
- Rotation
- Shear

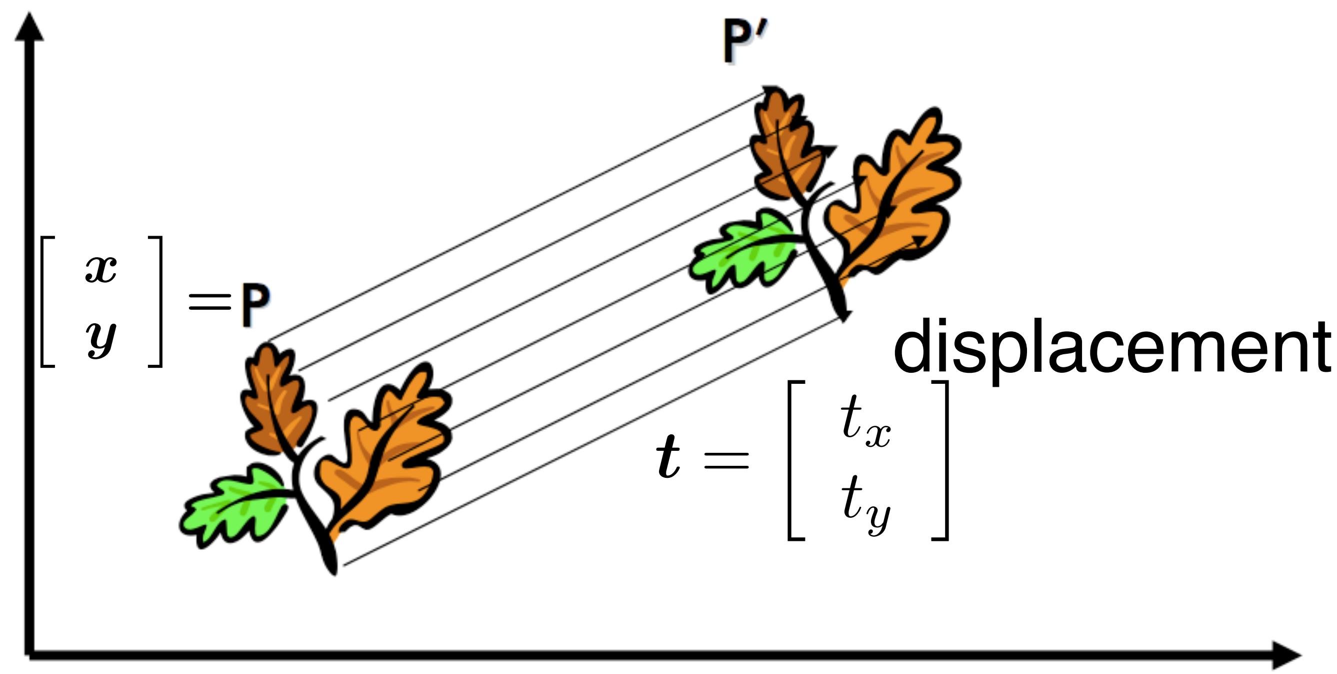
Example

$$(x', y') = T\{(x, y)\}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = T \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

homogeneous coordinates

2D Translation

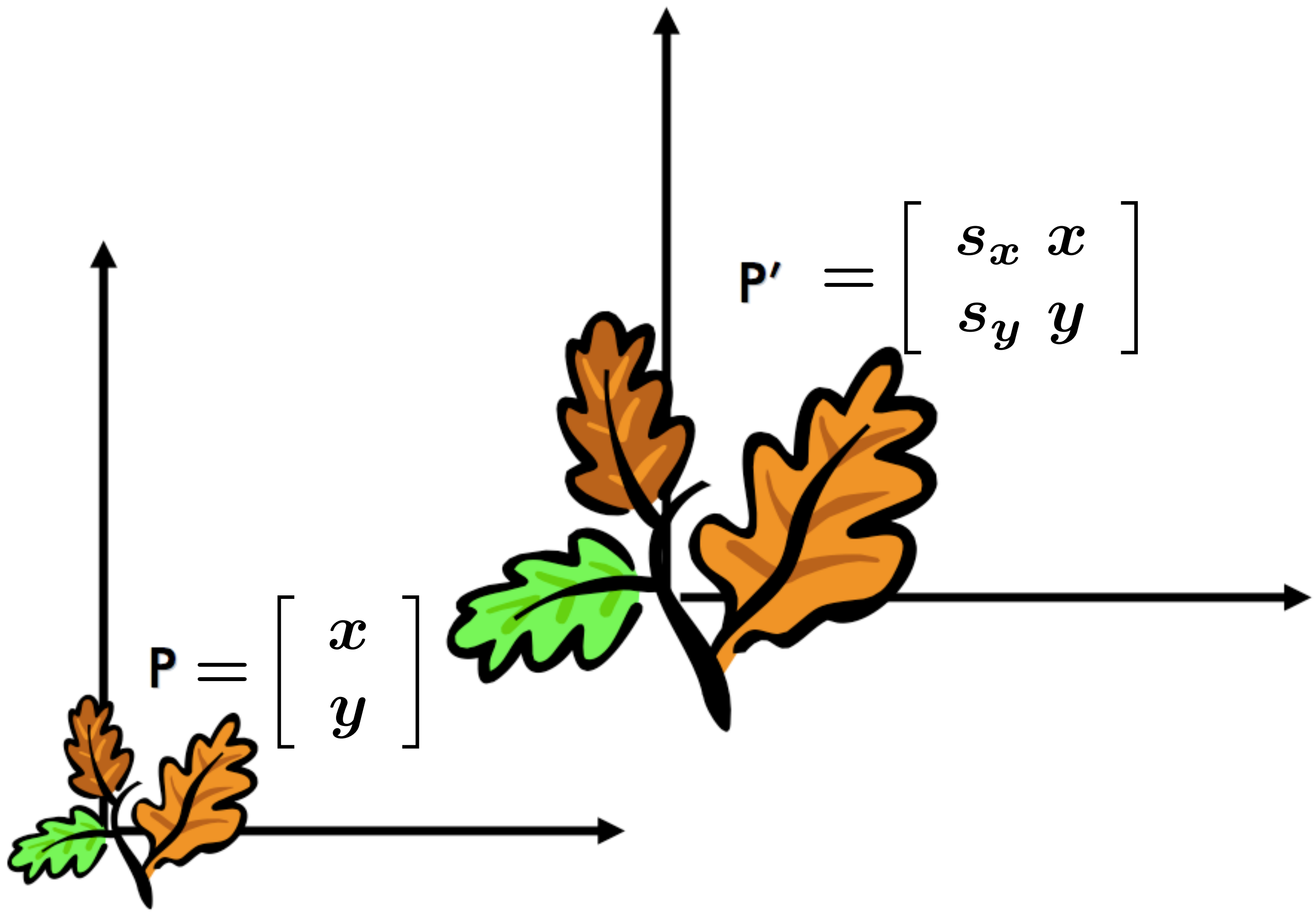


$$P' = P + \boldsymbol{t}$$

$$\Rightarrow \quad P' \rightarrow \begin{bmatrix} x + t_x \\ y + t_y \\ 1 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}}_{\text{translation matrix}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

source: S. Savarese

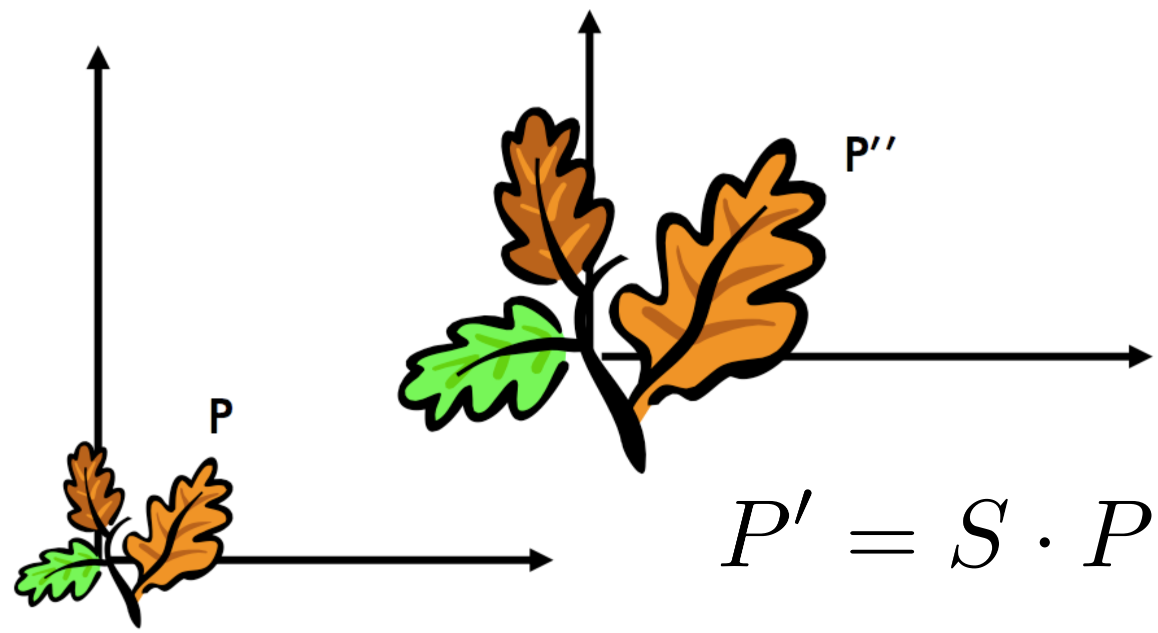
2D Scaling



$$\begin{bmatrix} s_x\ x \\ s_y\ y \\ 1 \end{bmatrix} = \underbrace{\overset{S}{\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}}_{\text{scaling matrix}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

source: S. Savarese

Complex Affine: First Scaling and then Translation



Is the ordering important?

$$P' = S \cdot P$$
$$P'' = T \cdot P' \quad \Rightarrow \quad P'' = (T \cdot S) \cdot P$$

$$\begin{bmatrix} x'' \\ y'' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

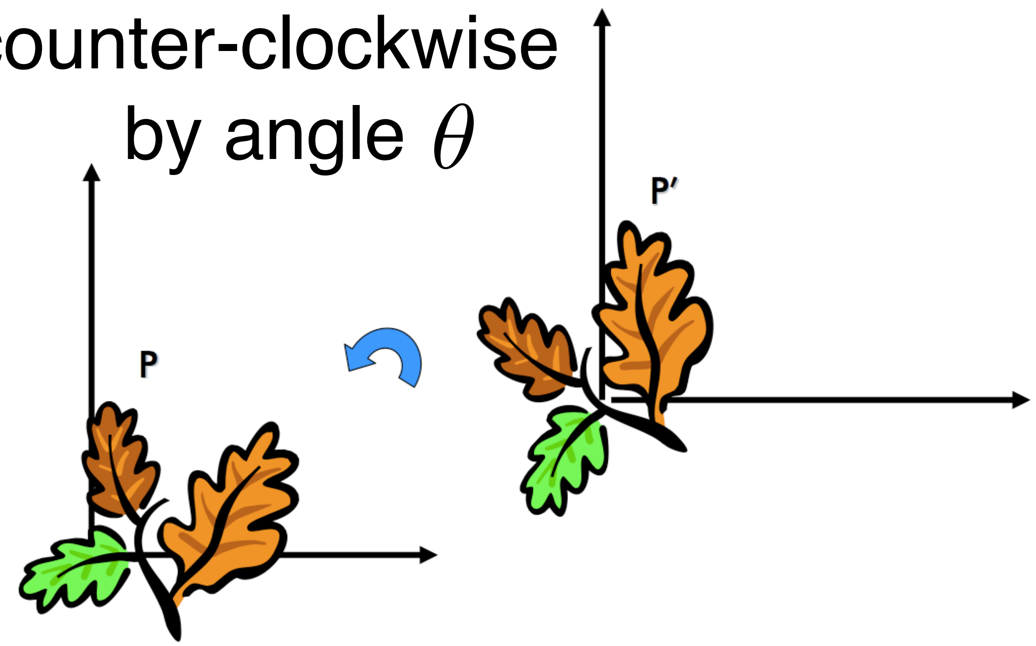
$$A = \begin{bmatrix} s_x & 0 & t_x \\ 0 & s_y & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

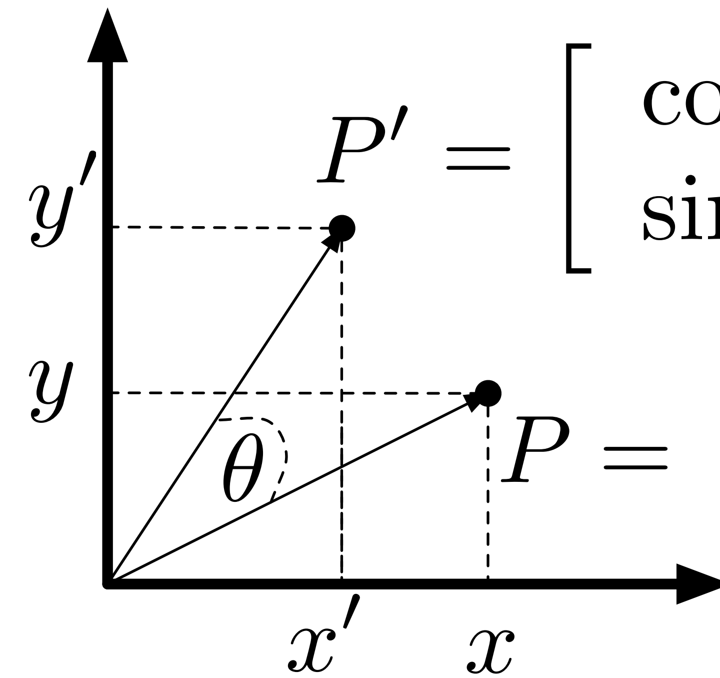
scaling + translation matrix

source: S. Savarese

2D Rotation

counter-clockwise
by angle θ





$$P' = \begin{bmatrix} \cos\theta \; x - \sin\theta \; y \\ \sin\theta \; x + \cos\theta \; y \end{bmatrix}$$

$$P = \begin{bmatrix} x \\ y \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \overset{R}{\underbrace{\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{\text{rotation matrix}}} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

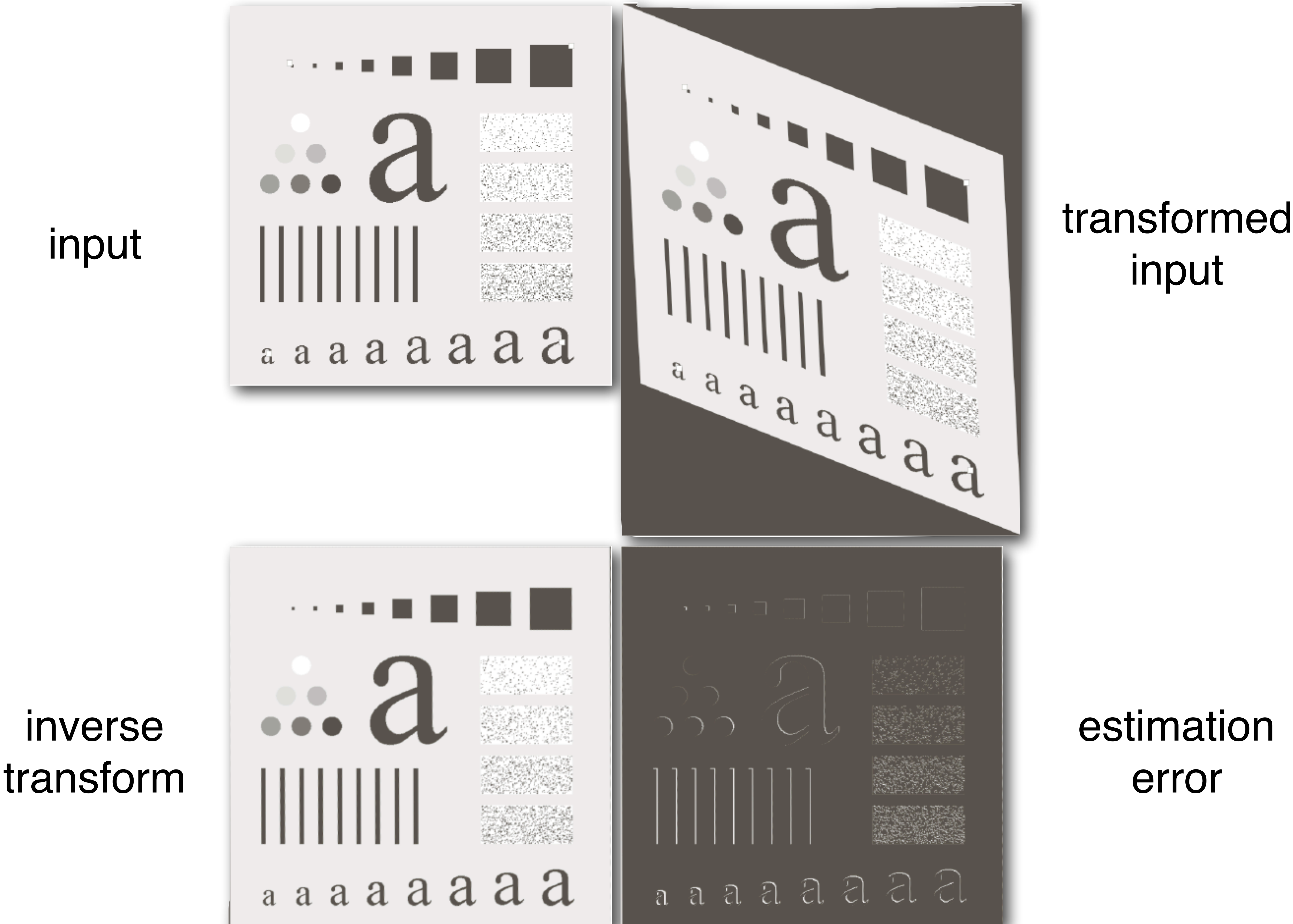
source: S. Savarese

Complex: First Scaling, Then Rotation, Finally Translation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \underbrace{\overset{\boldsymbol{t}}{\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}}}_{\text{translation matrix}} \underbrace{\overset{R}{\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}}}_{\text{rotation matrix}} \underbrace{\overset{S}{\begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix}}}_{\text{scaling matrix}}$$

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} R\ S & \boldsymbol{t} \\ \mathbf{0} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Estimating the Spatial Transform



MATLAB Example

```
>>img=imread(image_name); %input image

>>tform = maketform('affine',T); % set transform

>>img_out = imtransform(img, tform, interp);

>>imshow(img_out)
```

Image Interpolation

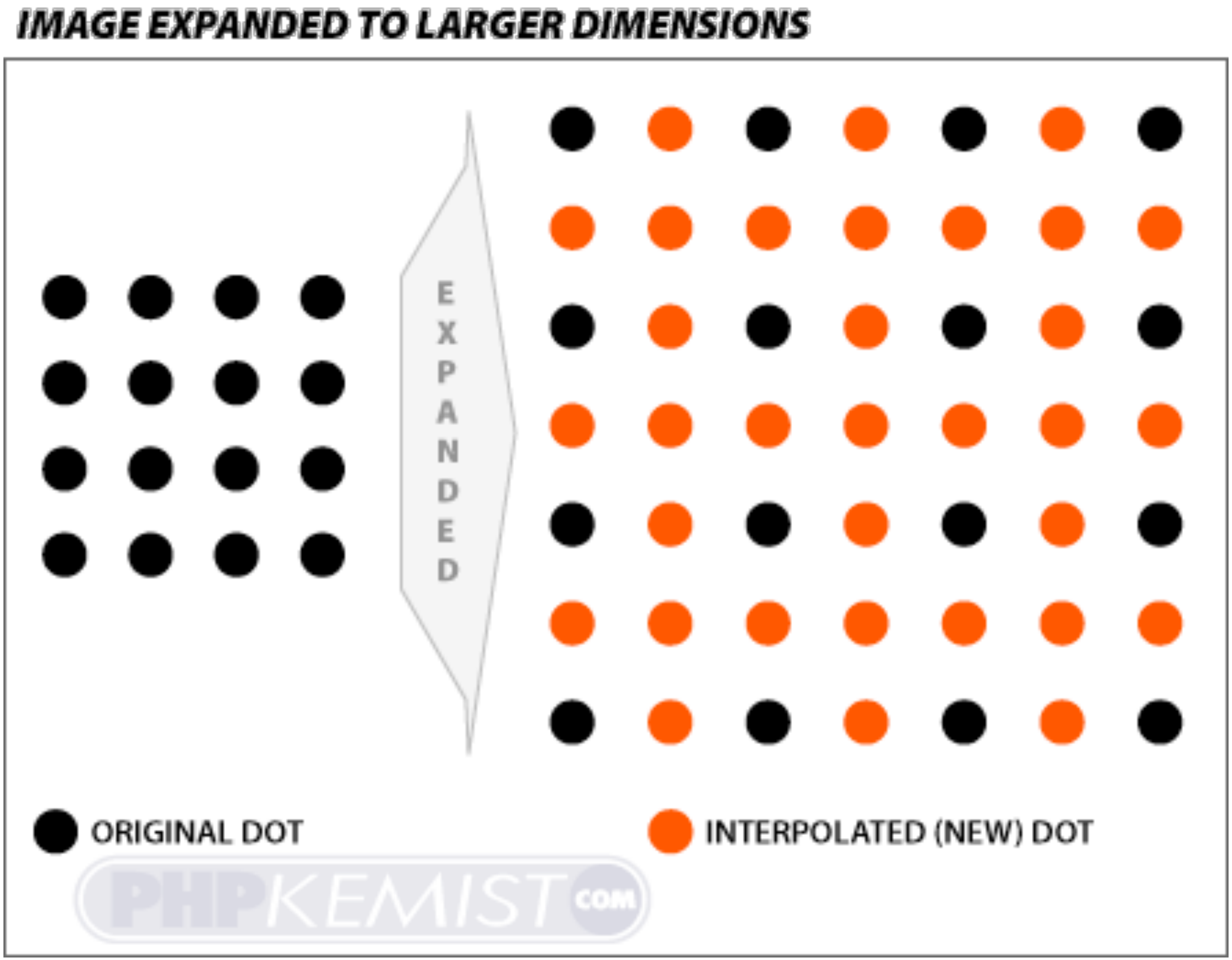

original


resampling

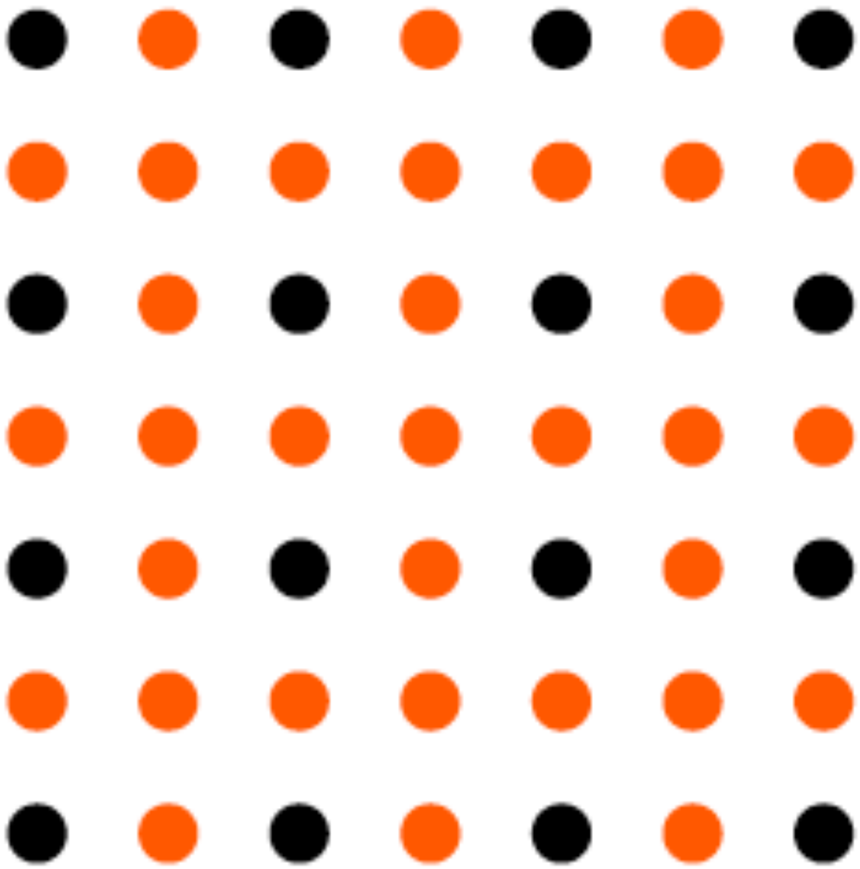

shrinking


zooming

Image Interpolation



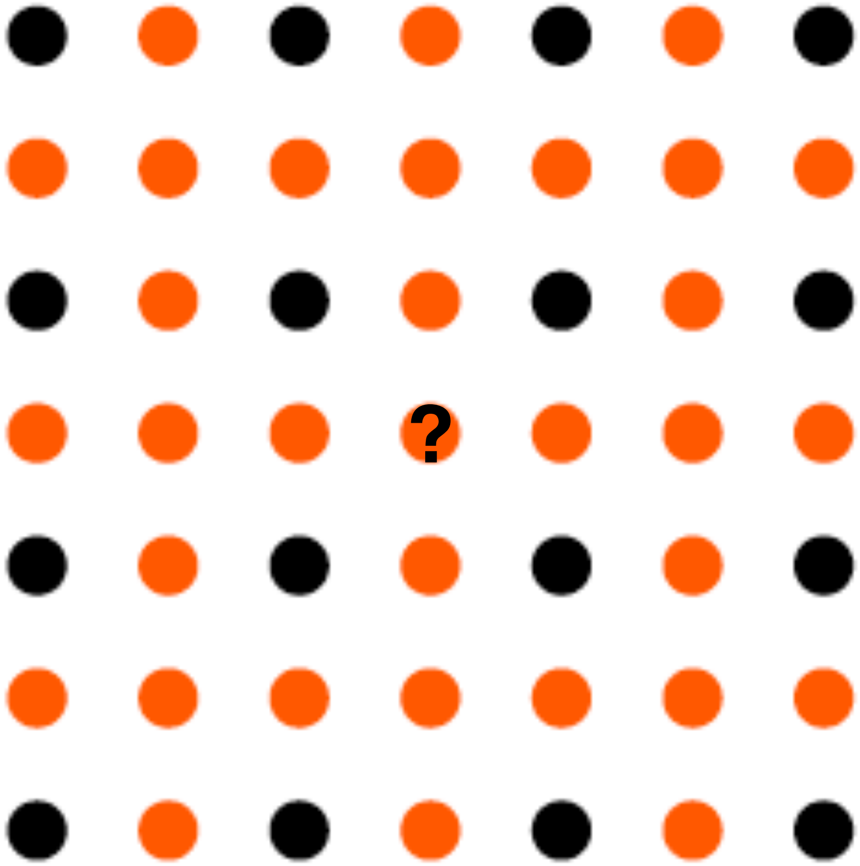
Bilinear Interpolation



$$f(x,y) = ax + by + cxy + d$$

coefficients that need to be estimated

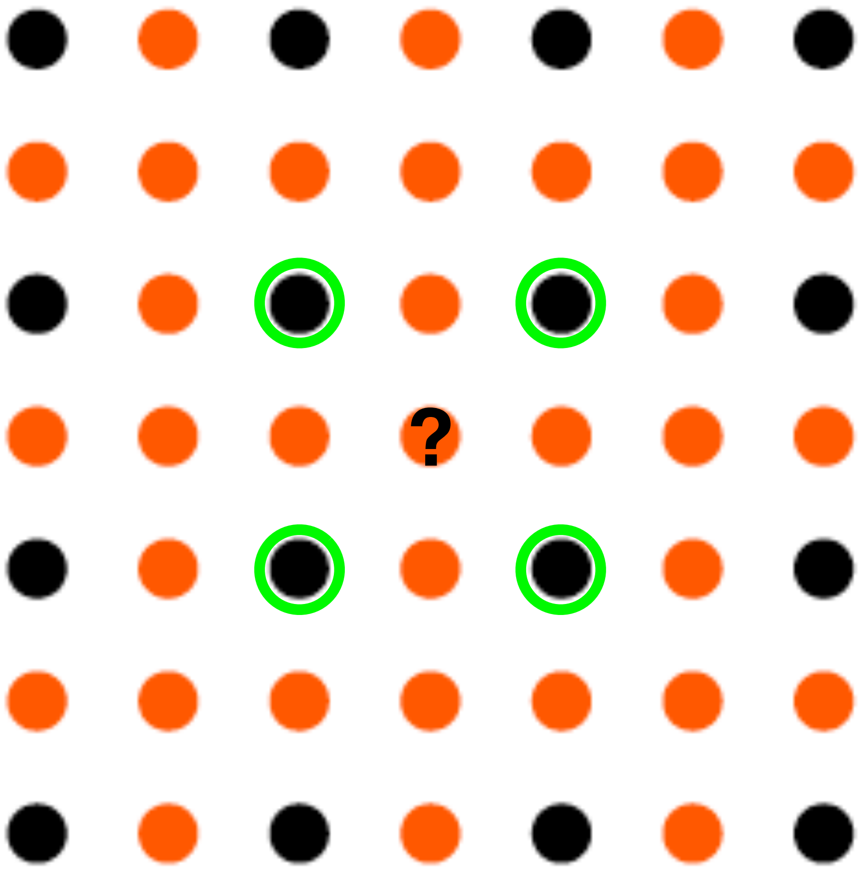
Bilinear Interpolation



$$f(x,y) = ax + by + cxy + d$$

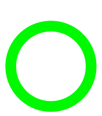
coefficients that need to be estimated

Bilinear Interpolation

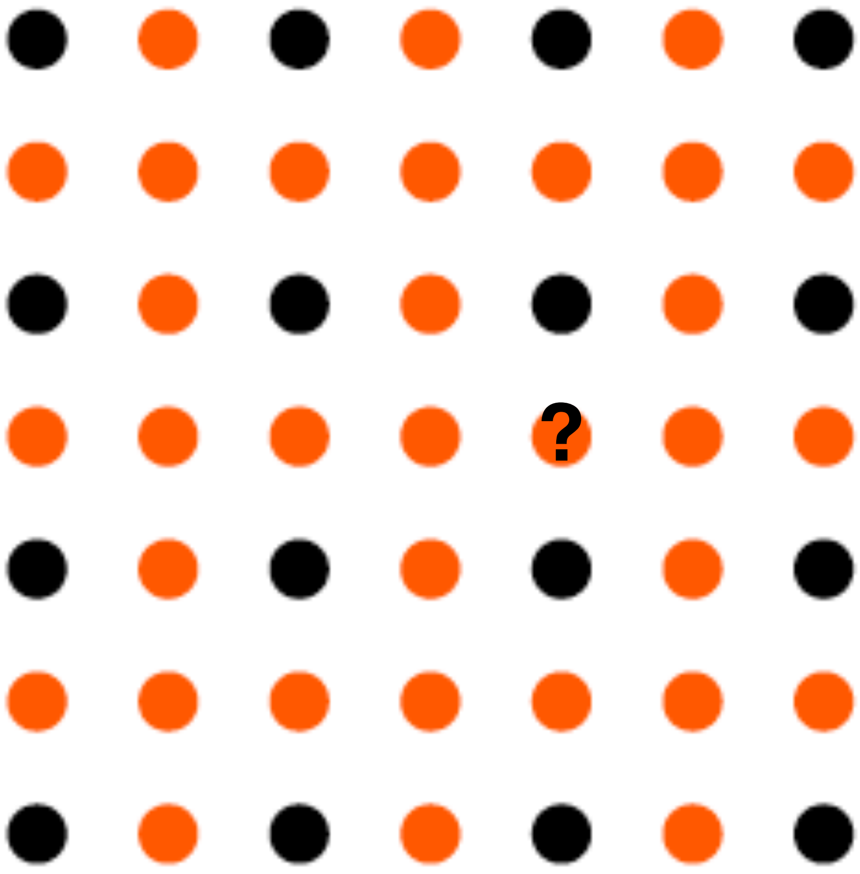


$$f(x,y) = ax + by + cxy + d$$

coefficients that need to be estimated



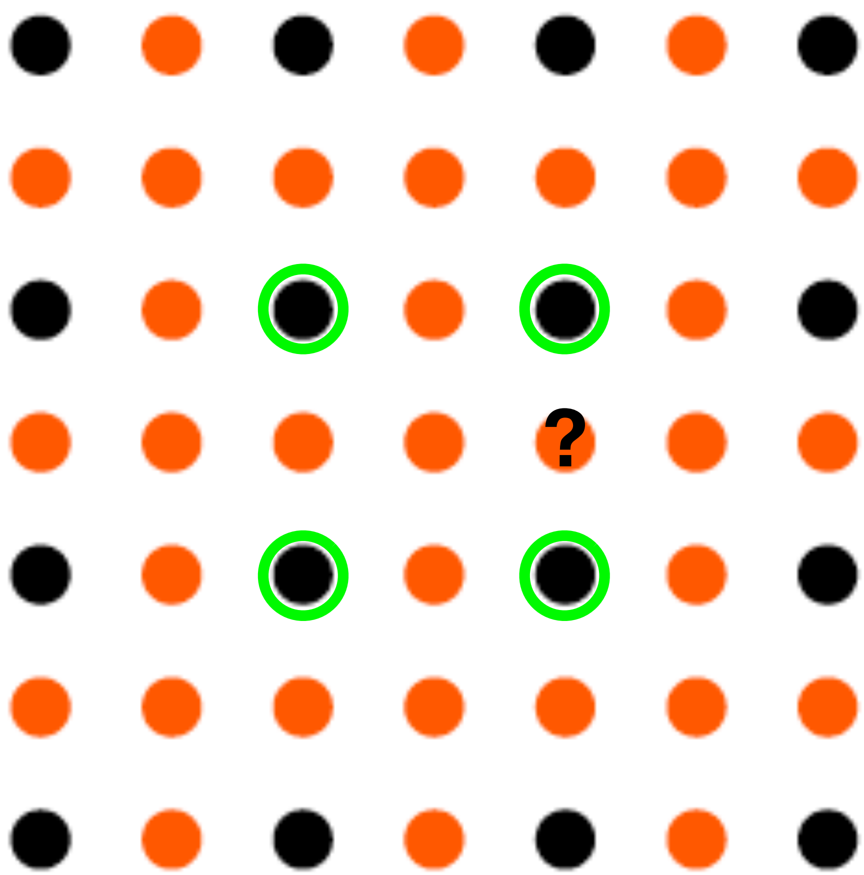
Bilinear Interpolation



$$f(x, y) = ax + by + cxy + d$$

coefficients that need to be estimated

Bilinear Interpolation

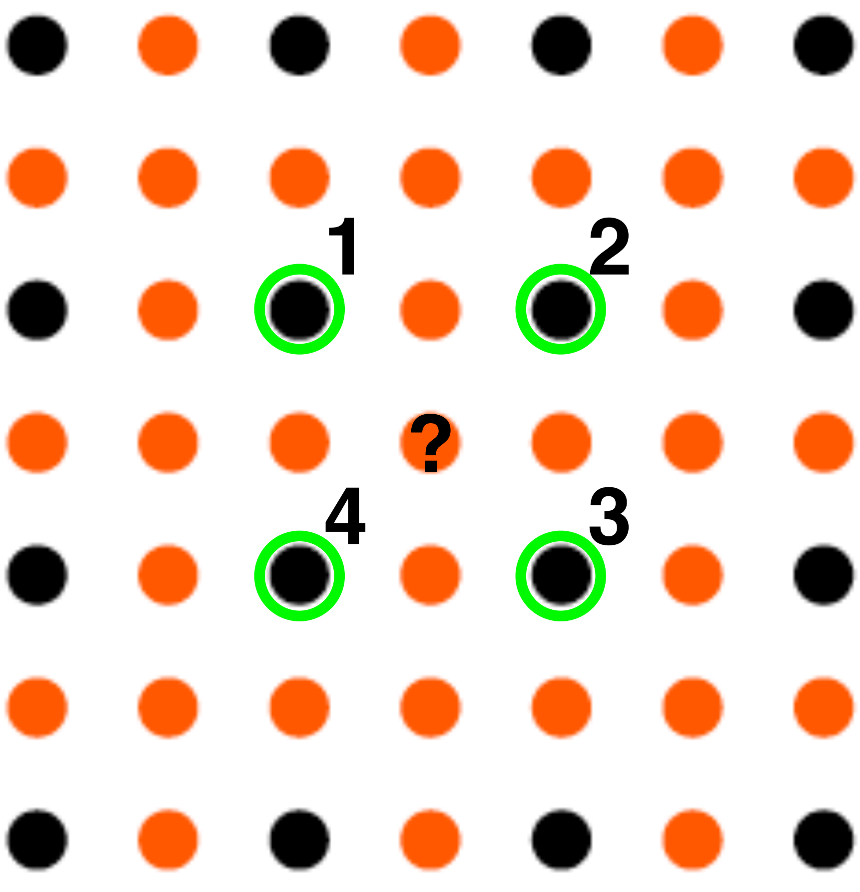


nearest neighbor

$$f(x, y) = ax + by + cxy + d$$

coefficients that need to be estimated

Bilinear Interpolation

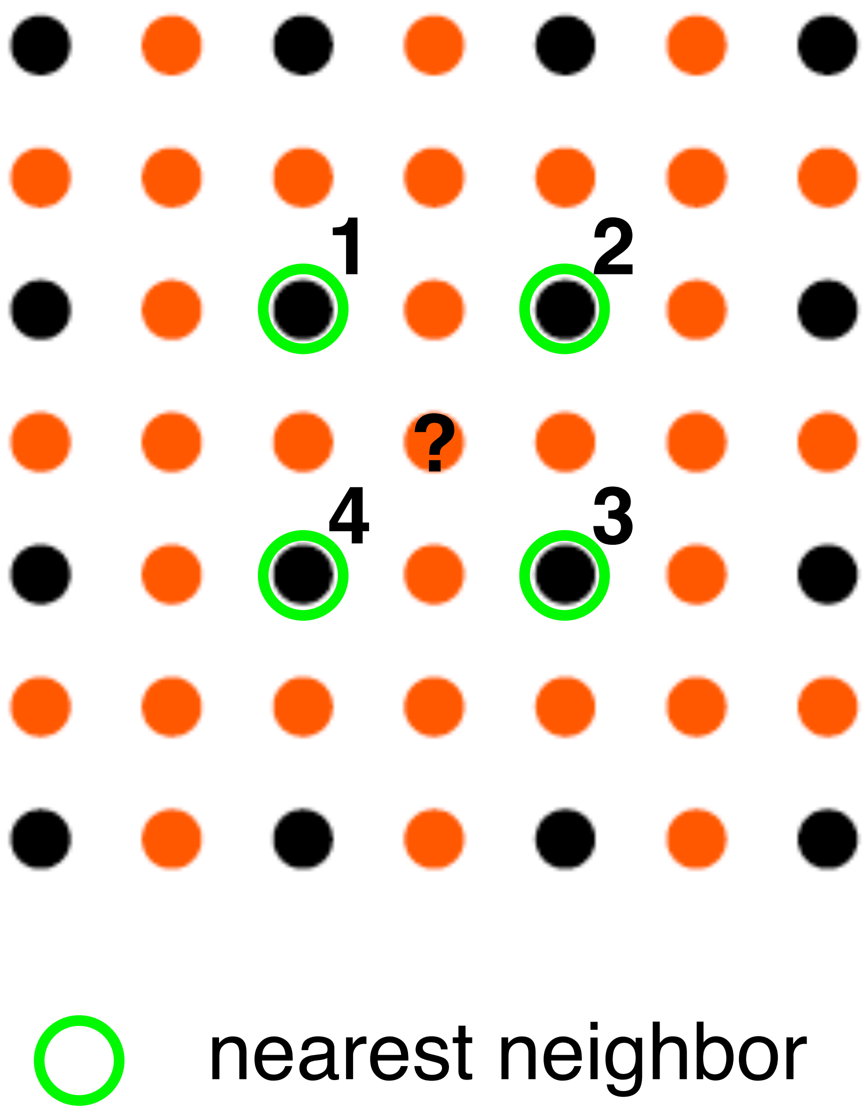


nearest neighbor

$$f(x, y) = ax + by + cxy + d$$

coefficients that need to be estimated

Bilinear Interpolation



$$f(x, y) = ax + by + cxy + d$$

coefficients that need to be estimated

$$ax_1 + by_1 + cx_1y_1 + d = f(x_1, y_1)$$

$$ax_2 + by_2 + cx_2y_2 + d = f(x_2, y_2)$$

$$ax_3 + by_3 + cx_3y_3 + d = f(x_3, y_3)$$

$$ax_4 + by_4 + cx_4y_4 + d = f(x_4, y_4)$$

$$\Downarrow$$

$$a,\ b,\ c,\ d$$

Example: Zooming-in

nearest neighbor

bilinear

Image Interpolation

- Bilinear N = 1
- Bicubic N = 3

$$f(x,y) = \sum_{i=0}^{N}\sum_{j=0}^{N} a_{ij}x^i y^j$$

new locations

new locations

estimated from the known neighboring locations

Intensity Transformations of Images

$$g(x, y) = T\{f(x, y)\}$$

transformed
input

input

Example: Shading Correction

$$g(x, y) = f(x, y)h(x, y)$$

input

shading profile

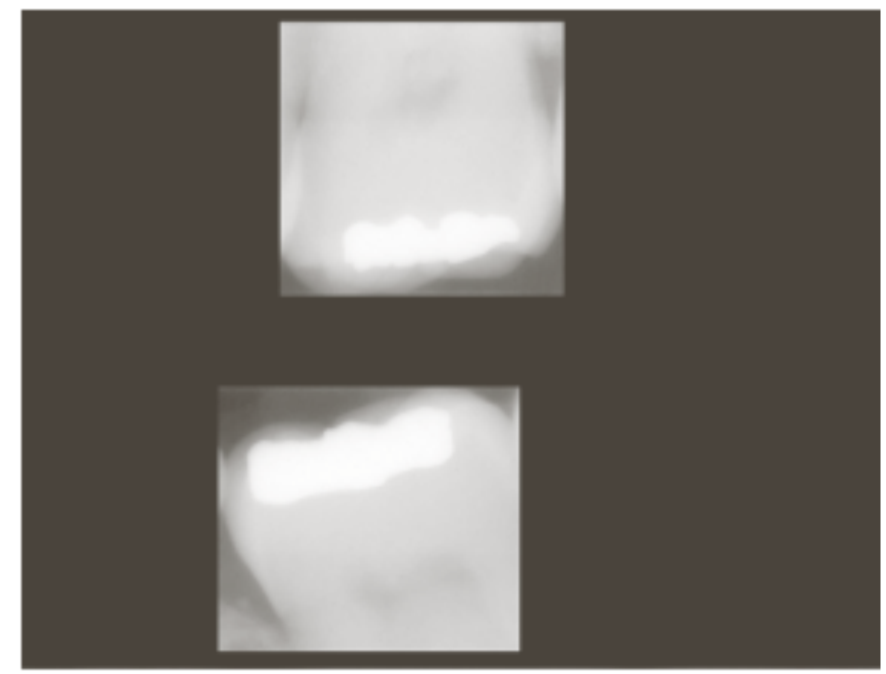
f(x,y)/shading

Example: Masking

$$g(x, y) = f(x, y)h(x, y)$$

input

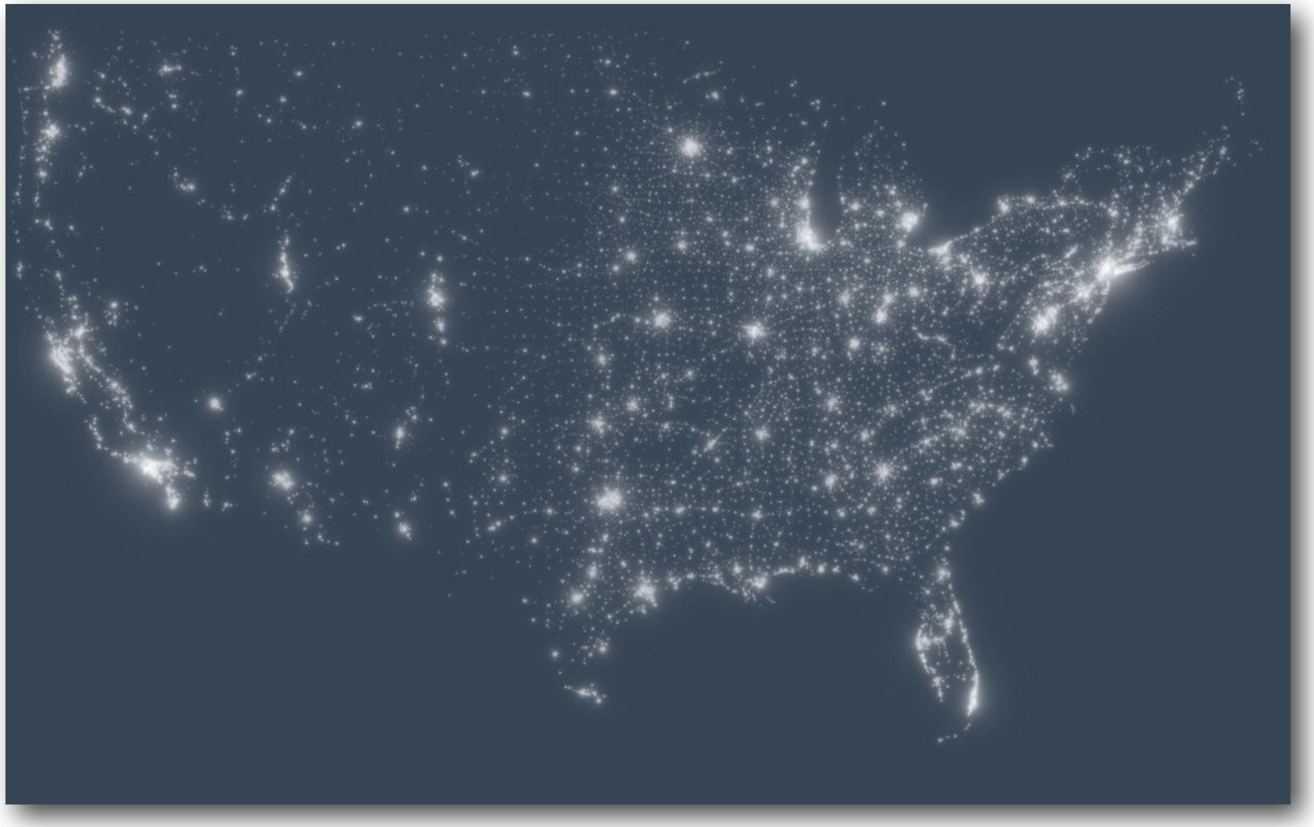
mask



f(x,y) * mask

Rescaling Intensity Values

$$g(x,y) = K\frac{f(x,y) - \min[f(x,y)]}{\max[f(x,y)] - \min[f(x,y)]}$$

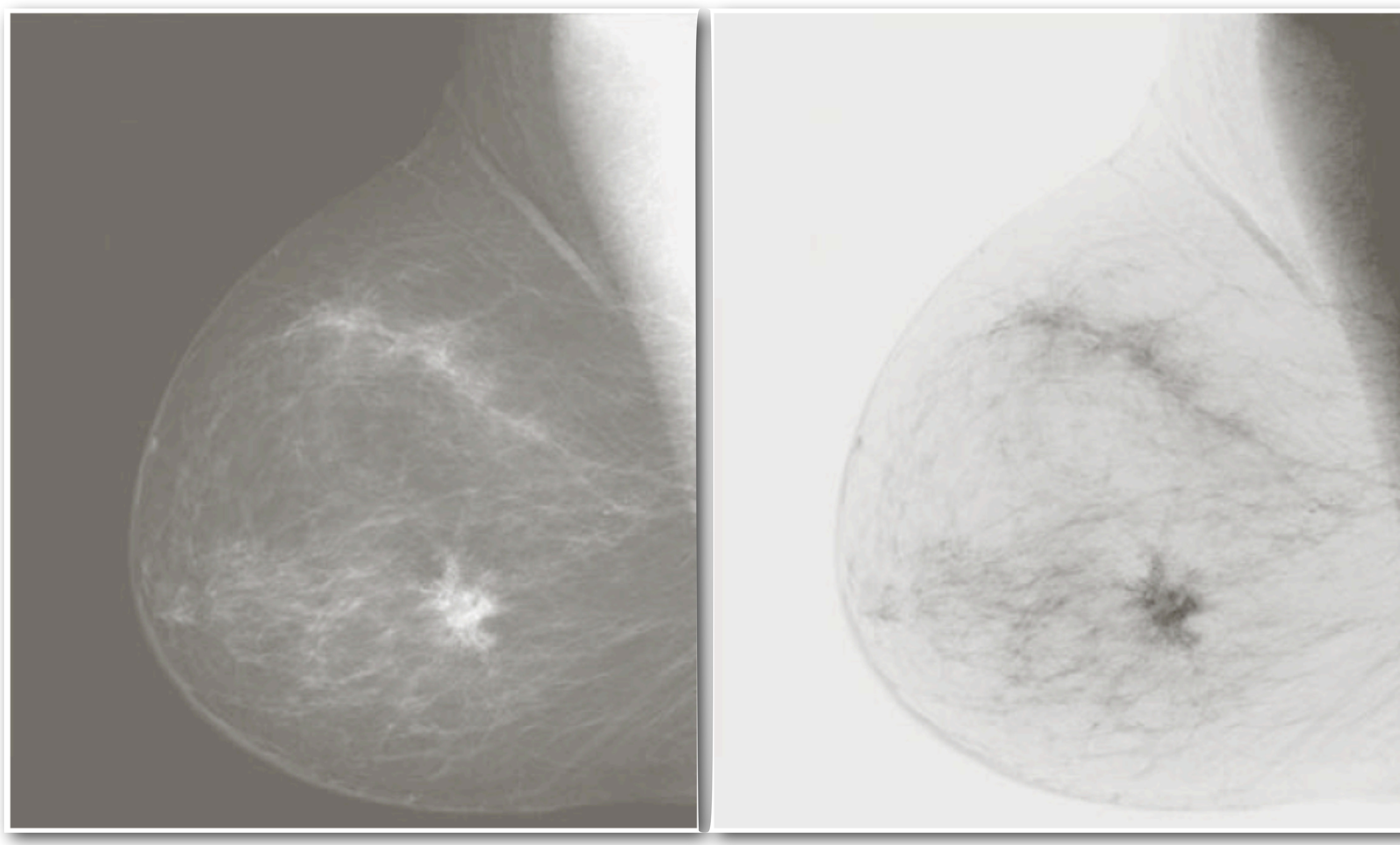


input

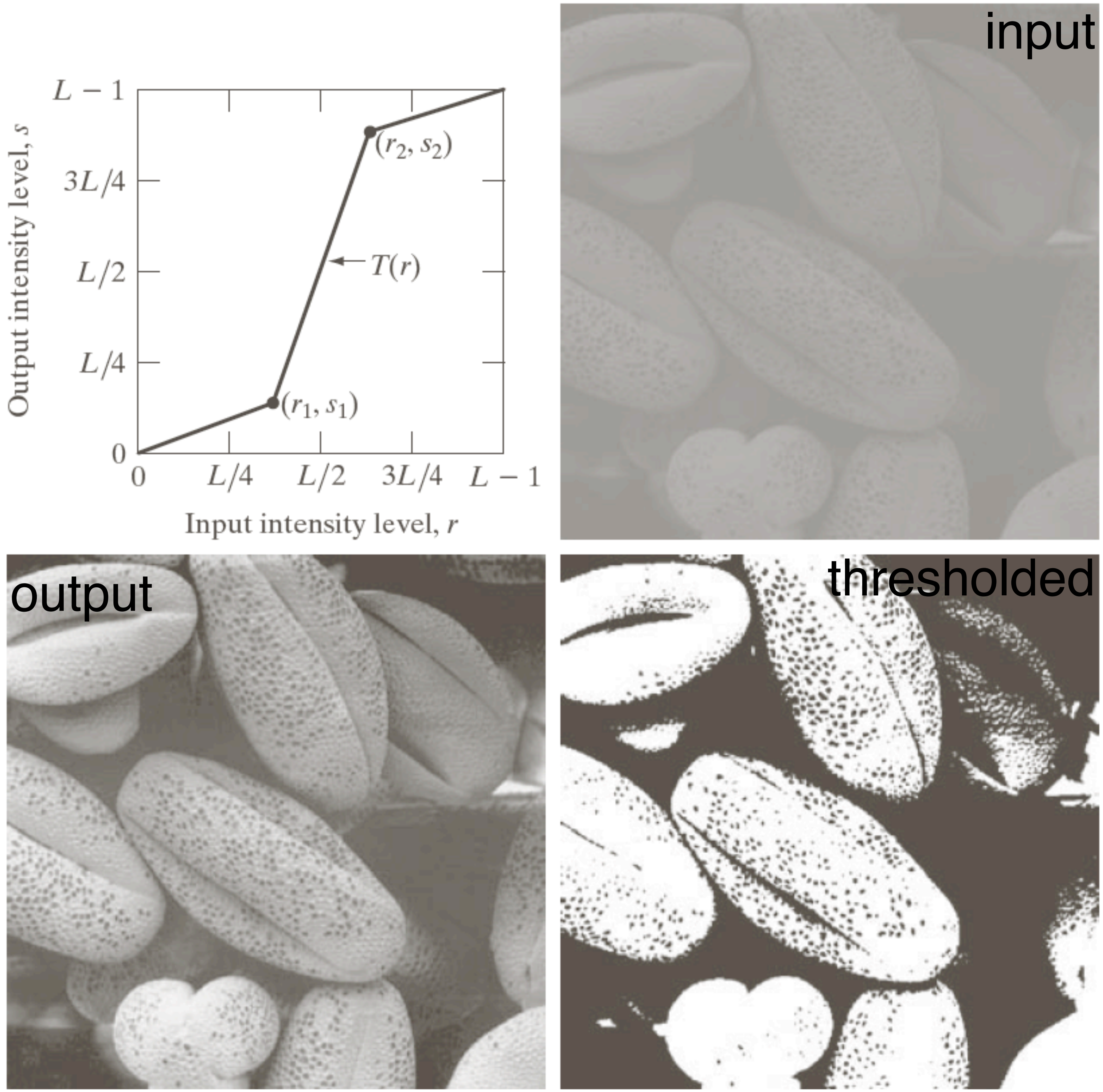
output

Image Negatives

$$g(x,y) = L - 1 - f(x,y)$$

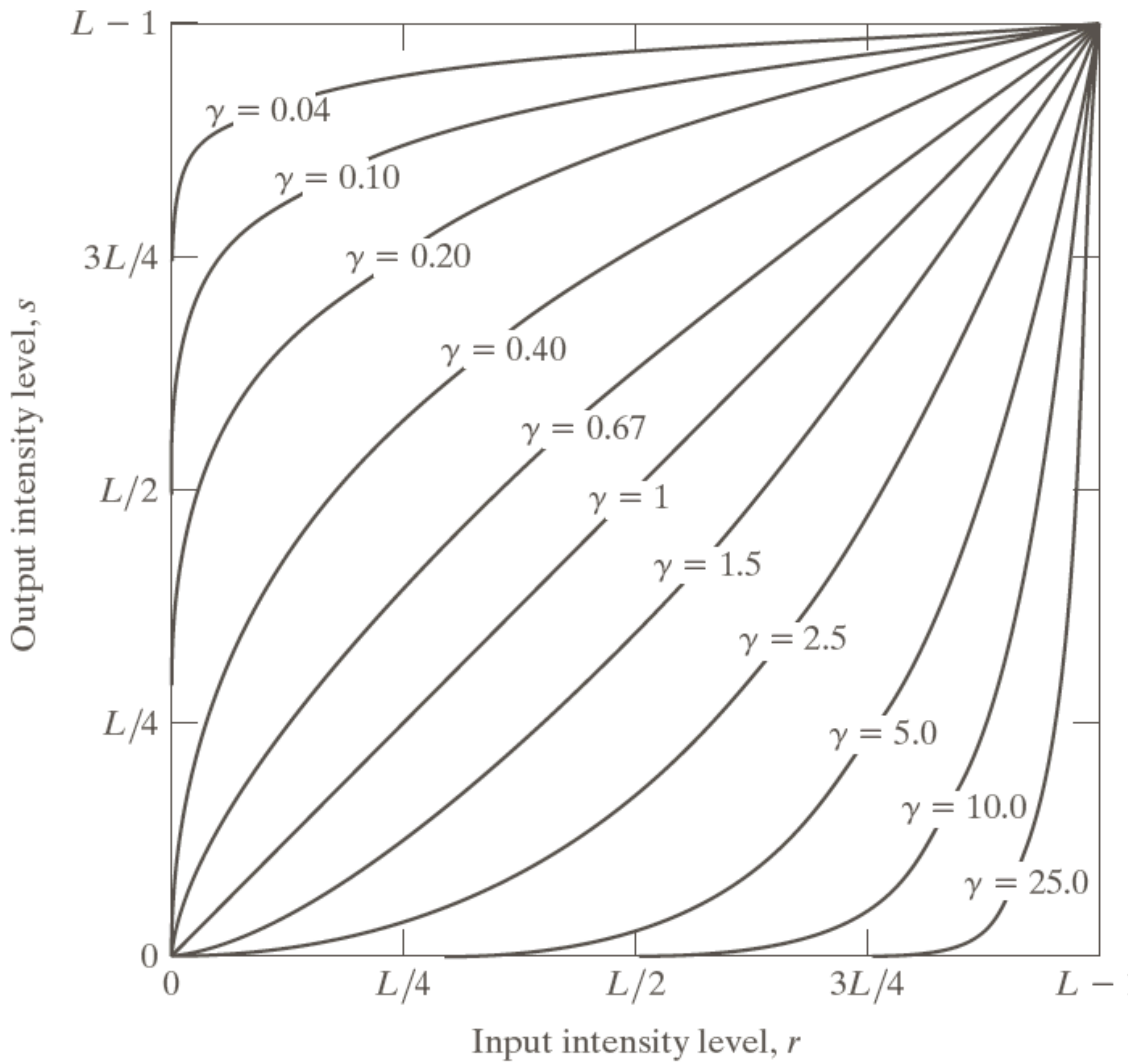


Contrast Stretching



Gamma Correction

$$s = cr^\gamma$$



Example: Gamma Correction