Image Processing 04
Transformations
Part 1

SS 2020

Prof. Dr. Simone Frintrop

Computer Vision Group, Department of Informatics
University of Hamburg, Germany

Outline

- Part 1: Overview Spatial Operations & Point Operations
- Part 1: Intensity transformations 1
- Part 2: Intensity transformations 2
- Part 3: Geometric transformations 1
- Part 4: Geometric transformations 2

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

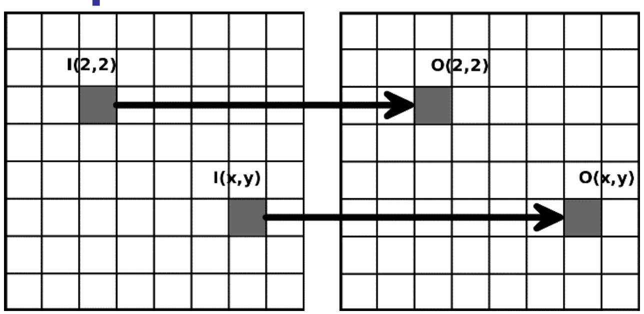
Spatial Operations

Spatial operations are operations directly applied to the pixels of an image.

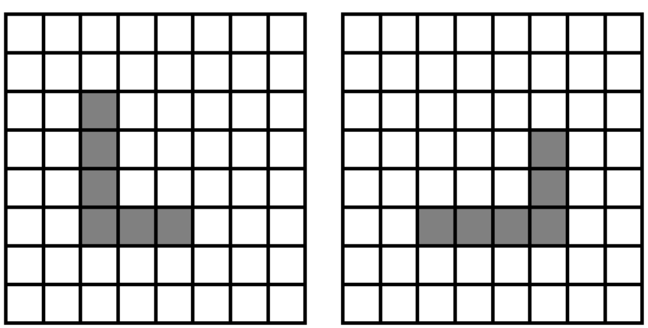
We distinguish 4 categories:

In this lecture

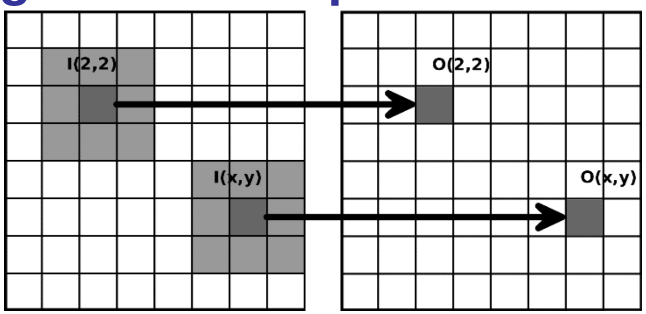
Point operations



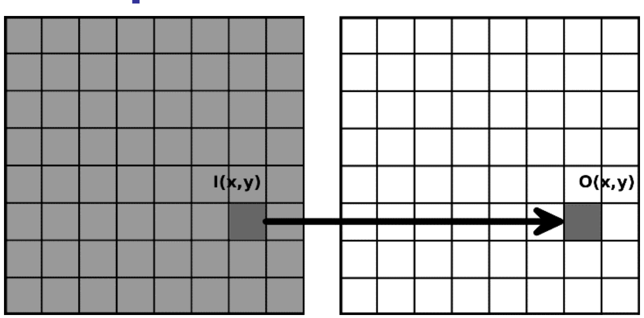
Geometric transformations



Neighborhood operations:



Global operations:

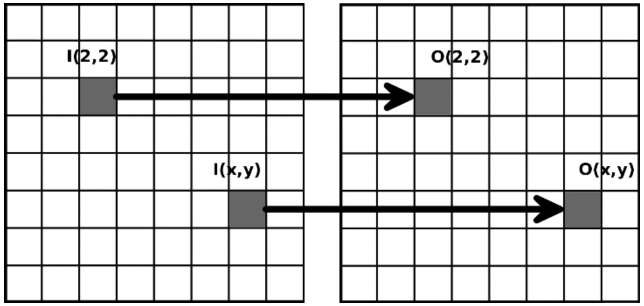


Enhancement

- „*Image Enhancement* is the process of manipulating an image so that the result is more suitable than the original for a specific application“ (Gonzales/Woods, p. 136)
- No general theory of enhancement
- The viewer is usually the judge
- Sometimes, it is the preprocessing step for a more complex machine vision system. Then, evaluation can be done by the performance of that system.

Point operations

- Point operations change the intensity or color of a pixel, only based on the current value of that pixel (not regarding its neighborhood/context)



- We will look here at intensity transformation functions

Intensity transformations

- Let f be an input image and g an output image.
- An intensity transformation function applied to image f is defined as:

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

- With $f(x, y) = r$ and $g(x, y) = s$, this becomes:

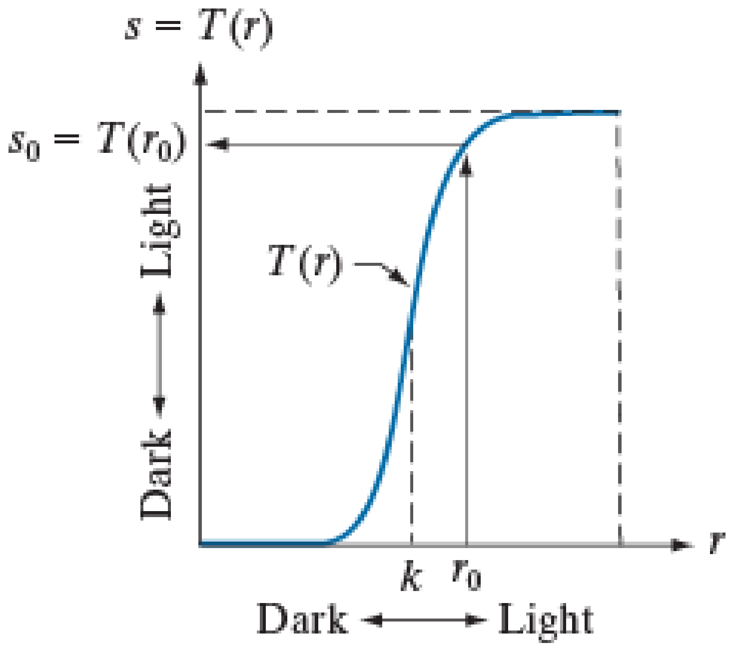
$$s = T(r)$$

Intensity transformations can be used for

- Contrast manipulation (in this lecture)
- Thresholding/Segmentation (later)

Contrast stretching

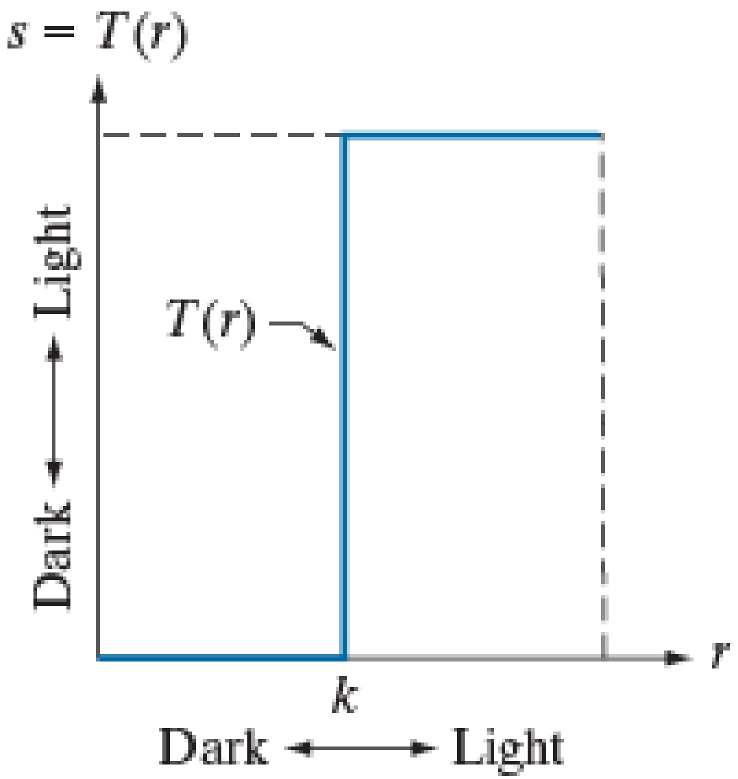
- *Example 1: Contrast stretching* produces an image with higher contrast, e.g. with this function:



[Gonzales/Woods, Fig. 3.2]

Thresholding

- *Example 2: Thresholding* produces a two-level (binary) image:



- Thresholding will be covered later

[Gonzales/Woods], Fig. 3.2

Intensity Transformation

- Some basic intensity transformation functions:

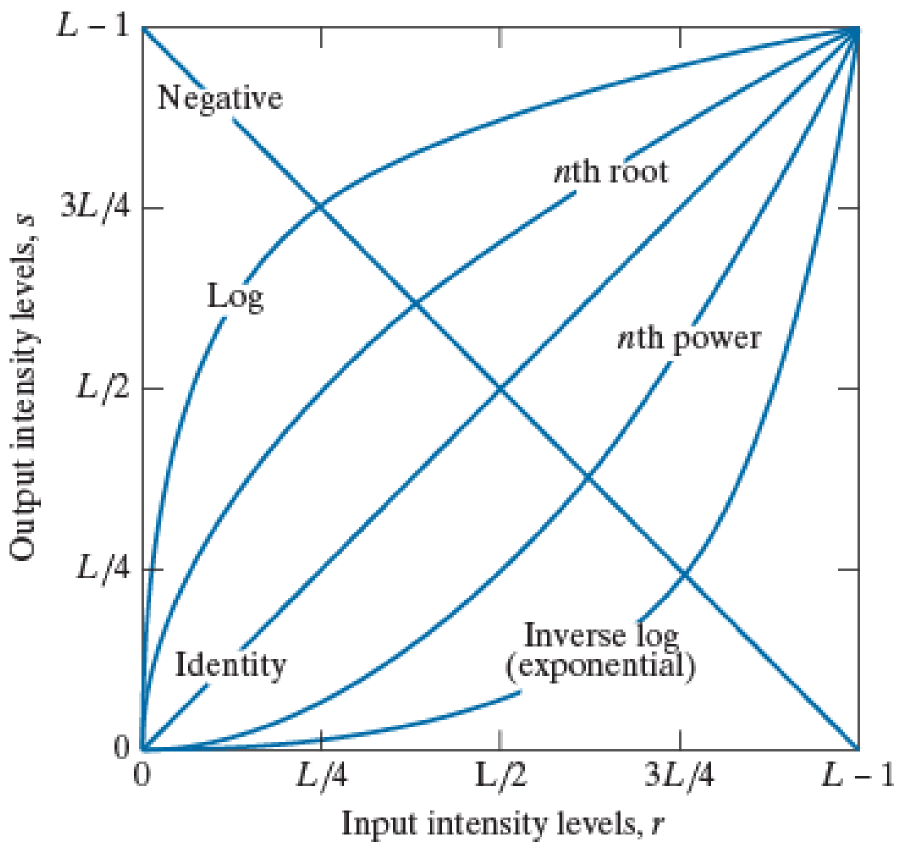
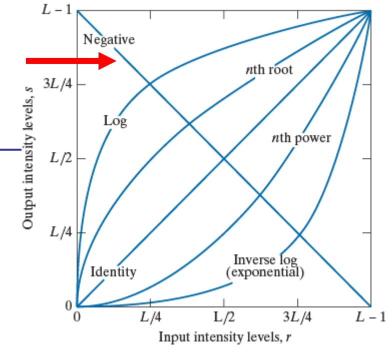


FIGURE 3.3 Some basic intensity transformation functions. Each curve was scaled *independently* so that all curves would fit in the same graph. Our interest here is on the *shapes* of the curves, not on their relative values.

Let us look at some of these functions in detail

[Gonzales/Woods]

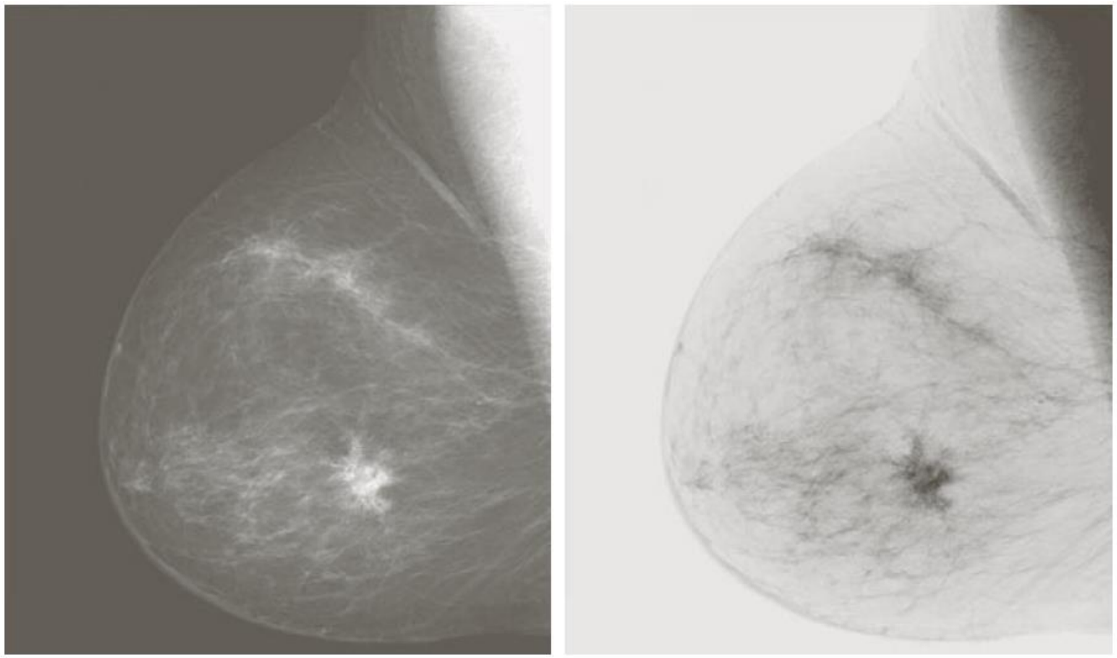
Image Negatives



- The negative of an image with the intensity range $[0, L-1]$ is obtained by: $s = L - 1 - r$

a b

FIGURE 3.4 (a) A digital mammogram. (b) Negative image obtained using Eq. (3-3). (Image (a) Courtesy of General Electric Medical Systems.)

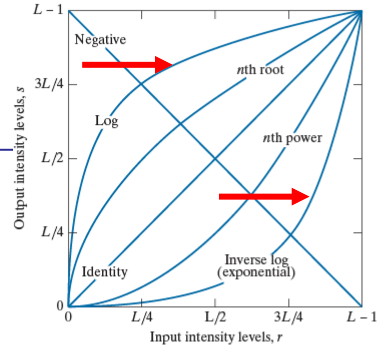


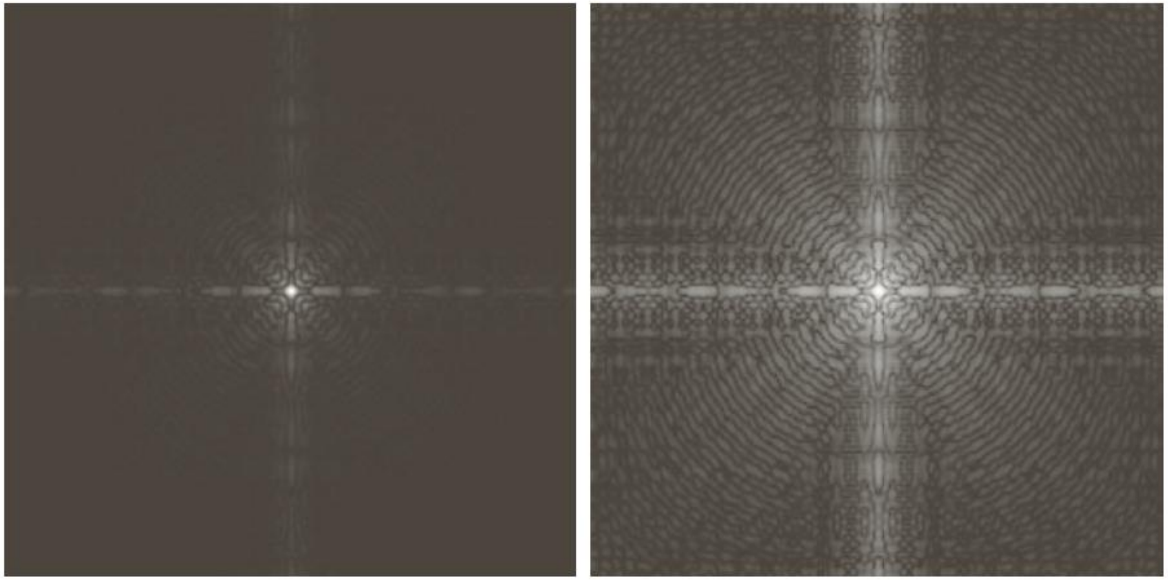
[Gonzales/Woods]

Log Transformations

- Log transformations are obtained by:

$$s = c \cdot log(1 + r)$$





a b

FIGURE 3.5 (a) Fourier spectrum displayed as a grayscale image. (b) Result of applying the log transformation in Eq. (3-4) with $c = 1$. Both images are scaled to the range [0, 255].

[Gonzales/Woods]

Outline

- Part 1: Overview Spatial Operations & Point Operations
- Part 1: Intensity transformations 1
- Part 2: Intensity transformations 2
- Part 3: Geometric transformations 1
- Part 4: Geometric transformations 2

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Image Processing 04 Transformations Part 2

SS 2020

Prof. Dr. Simone Frintrop

Computer Vision Group, Department of Informatics
University of Hamburg, Germany

Outline

- Part 1: Overview Spatial Operations & Point Operations
- Part 1: Intensity transformations 1
- Part 2: Intensity transformations 2
- Part 3: Geometric transformations 1
- Part 4: Geometric transformations 2

Intensity Transformation

- Some basic intensity transformation functions:

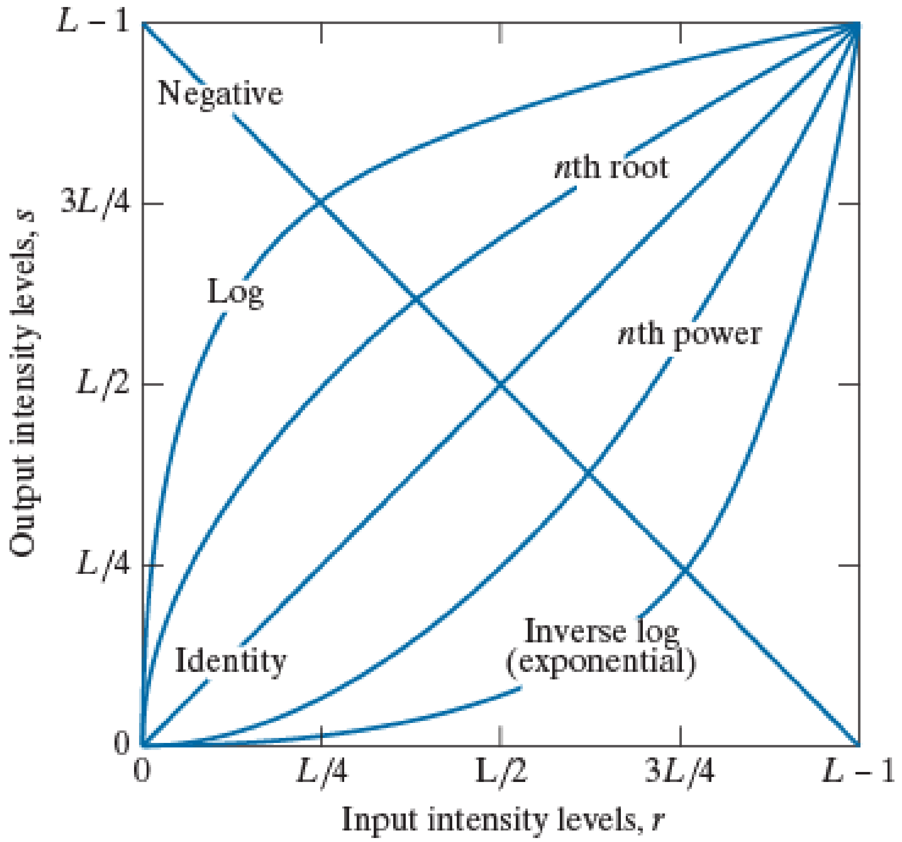
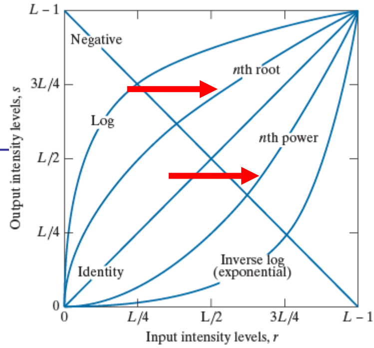


FIGURE 3.3 Some basic intensity transformation functions. Each curve was scaled *independently* so that all curves would fit in the same graph. Our interest here is on the *shapes* of the curves, not on their relative values.

[Gonzales/Woods]

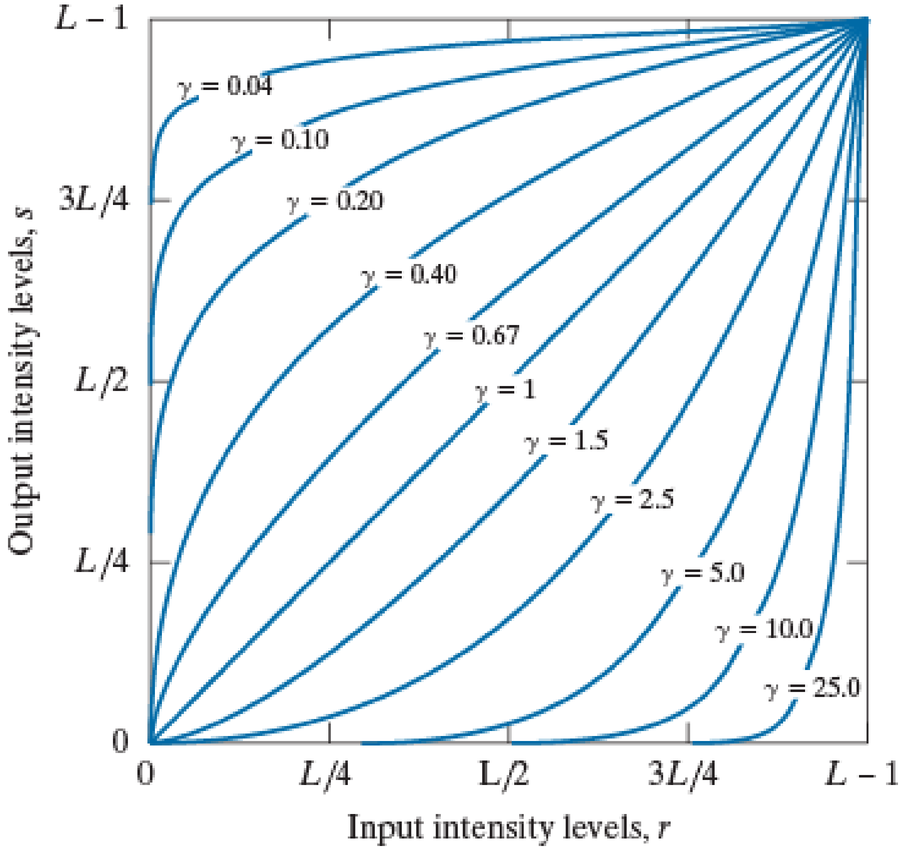
Power-law transformations



- Power-law transformations have the form:

$$s = c \cdot r^{\gamma}$$

FIGURE 3.6 Plots of the gamma equation $s = cr^{\gamma}$ for various values of γ ($c = 1$ in all cases). Each curve was scaled *independently* so that all curves would fit in the same graph. Our interest here is on the *shapes* of the curves, not on their relative values.



[Gonzales/Woods]

Gamma correction

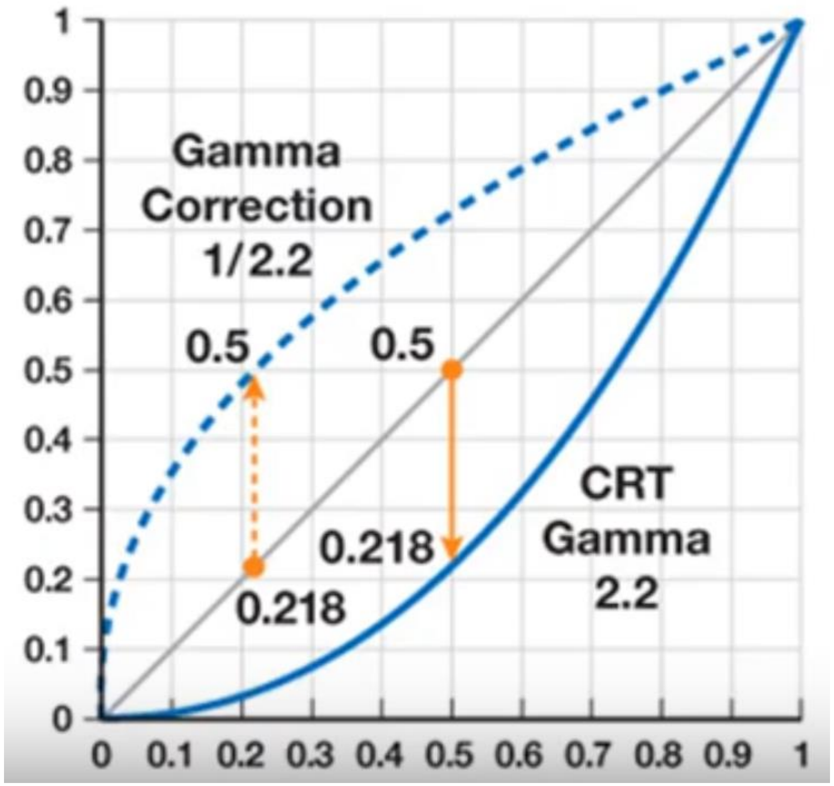
- The response of many devices for image capture, printing, and display obey power law
- E.g., monitors: gamma in range 1.8 - 2.5:

- The process used to correct these power-law response phenomena is called *gamma correction*
- It means to apply the inverse of the monitor transformation to the image before displaying

[Gonzales/Woods]

Gamma correction

- Example:

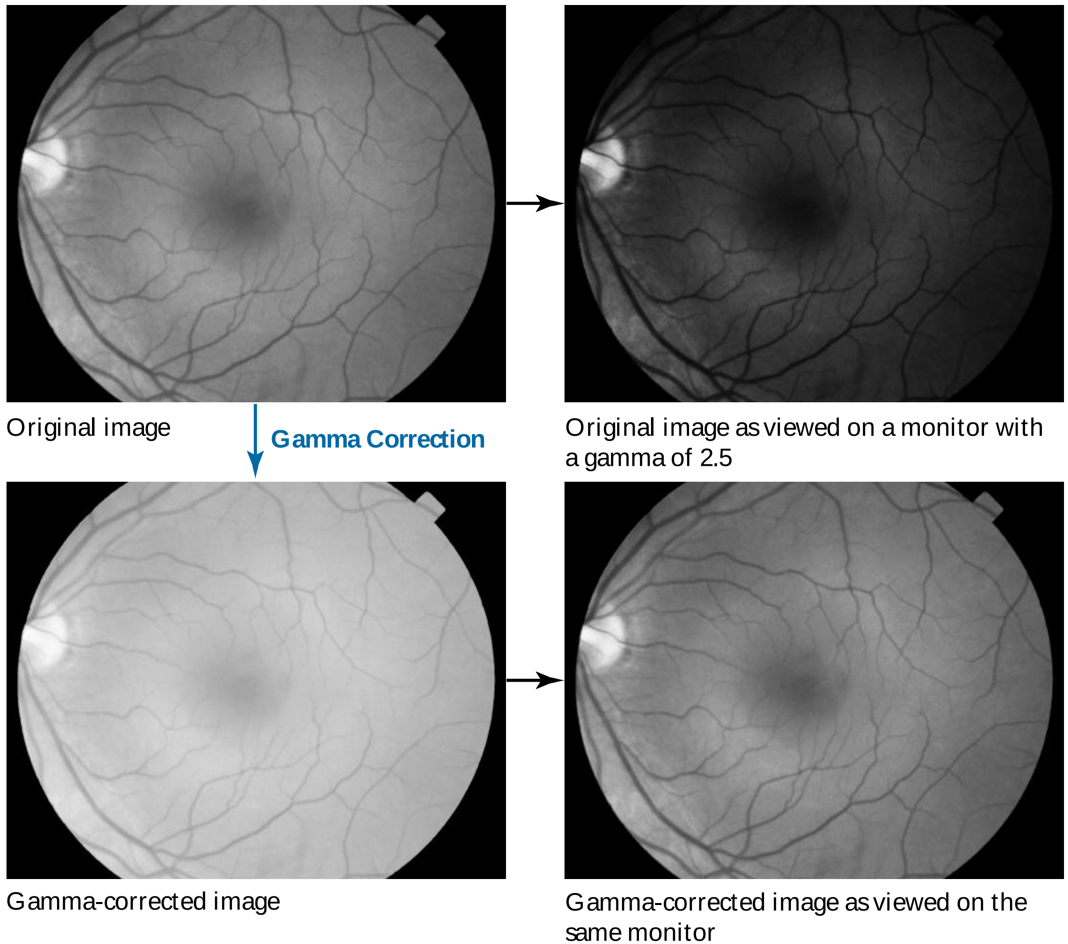


- This happens automatically in monitors

[Bertoa 2016]

Gamma correction

Example for gamma correction:



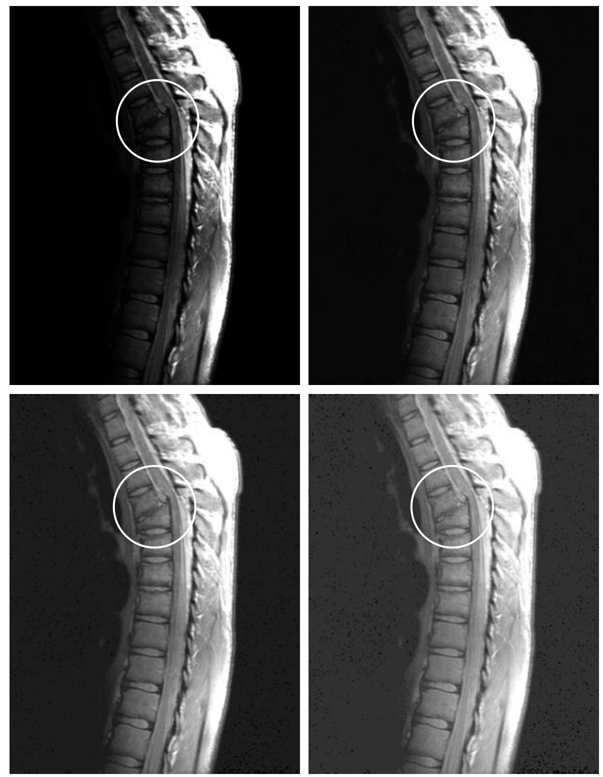
a b
c d

FIGURE 3.7
(a) Image of a human retina.
(b) Image as as it appears on a monitor with a gamma setting of 2.5 (note the darkness).
(c) Gamma-corrected image.
(d) Corrected image, as it appears on the same monitor (compare with the original image).
(Image (a) courtesy of the National Eye Institute, NIH)

[Gonzales/Woods]

Contrast manipulation

- Power-law transformations are also useful for contrast manipulation in general:
- Example: MRI images of human upper thoracic spine (Brustwirbelsäule):



a b
c d

FIGURE 3.8 (a) Magnetic resonance image (MRI) of a fractured human spine (the region of the fracture is enclosed by the circle). (b)–(d) Results of applying the transformation in Eq. (3-5) with $c = 1$ and $\gamma = 0.6, 0.4,$ and 0.3, respectively. (Original image courtesy of Dr. David R. Pickens, Department of Radiology and Radiological Sciences, Vanderbilt University Medical Center.)

[Gonzales/Woods]

Contrast manipulation

- Example 2: Power-law transformation on aerial image

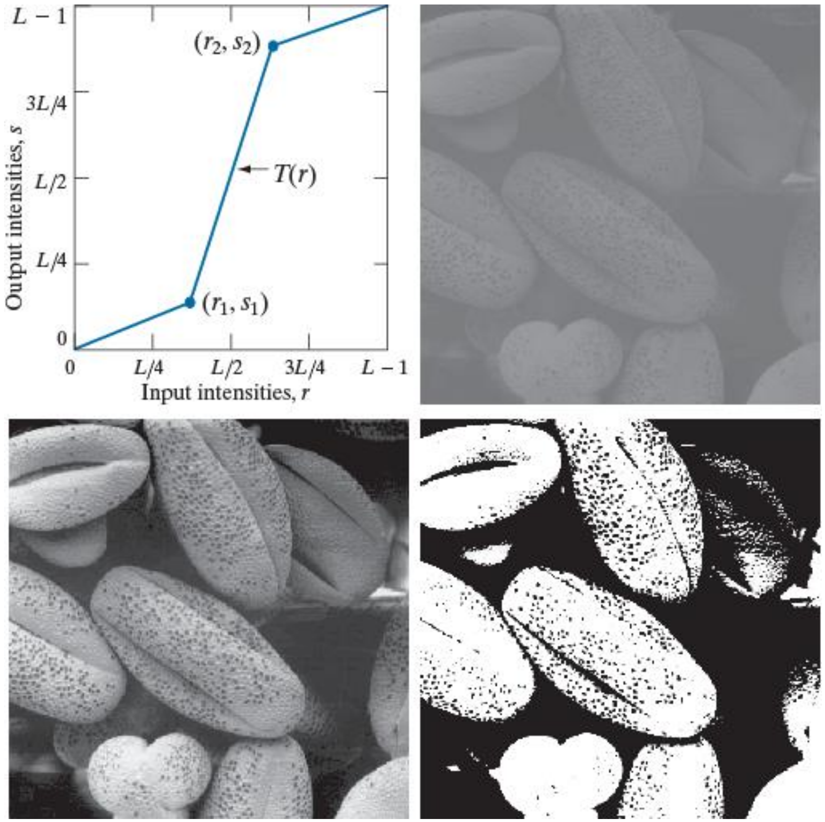
a b
c d

FIGURE 3.9
(a) Aerial image. (b)–(d) Results of applying the transformation in Eq. (3-5) with $\gamma = 3.0, 4.0$, and 5.0, respectively. ($c = 1$ in all cases.) (Original image courtesy of NASA.)

[Gonzales/Woods]

Contrast stretching

- Contrast stretching can be also done with piecewise linear transformation functions:



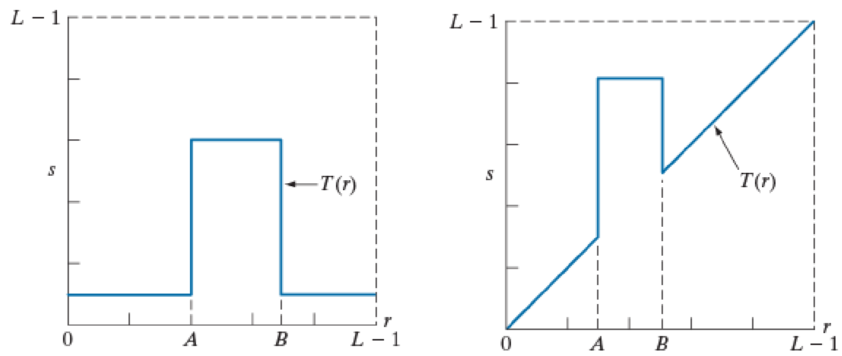
a b
c d

FIGURE 3.10 Contrast stretching. (a) Piecewise linear transformation function. (b) A low-contrast electron microscope image of pollen, magnified 700 times. (c) Result of contrast stretching. (d) Result of thresholding. (Original image courtesy of Dr. Roger Heady, Research School of Biological Sciences, Australian National University, Canberra, Australia.)

[Gonzales/Woods]

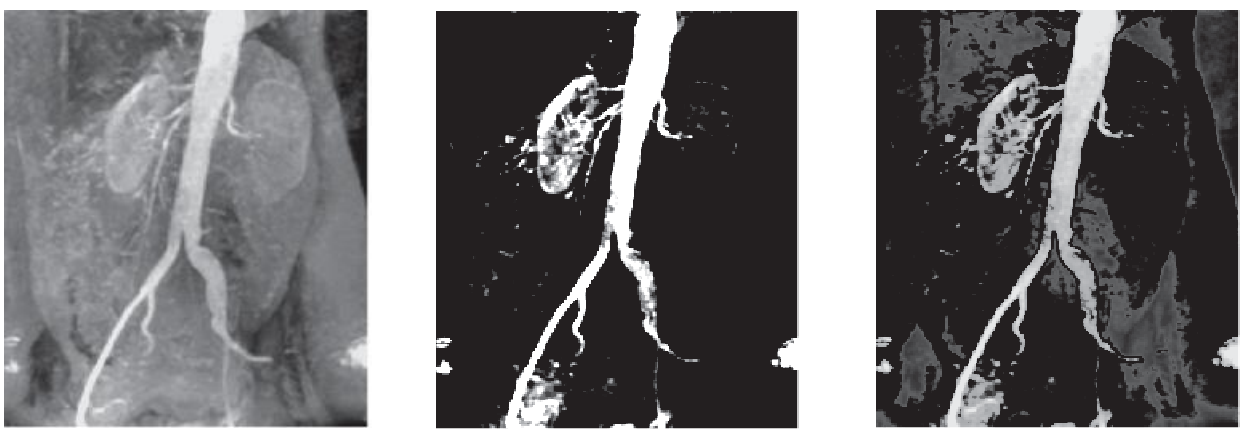
Intensity-level slicing

- Intensity-level slicing highlights a specific range of images



a b

FIGURE 3.11 (a) This transformation function highlights range $[A, B]$ and reduces all other intensities to a lower level. (b) This function highlights range $[A, B]$ and leaves other intensities unchanged.



a b c

FIGURE 3.12 (a) Aortic angiogram. (b) Result of using a slicing transformation of the type illustrated in Fig. 3.11(a), with the range of intensities of interest selected in the upper end of the gray scale. (c) Result of using the transformation in Fig. 3.11(b), with the selected range set near black, so that the grays in the area of the blood vessels and kidneys were preserved. (Original image courtesy of Dr. Thomas R. Gest, University of Michigan Medical School.)

[Gonzales/Woods]

Outline

- Part 1: Overview Spatial Operations & Point Operations
- Part 1: Intensity transformations 1
- Part 2: Intensity transformations 2
- Part 3: Geometric transformations 1
- Part 4: Geometric transformations 2

Image Processing 04
Transformations
Part 3

SS 2020

Prof. Dr. Simone Frintrop

Computer Vision Group, Department of Informatics
University of Hamburg, Germany

Outline

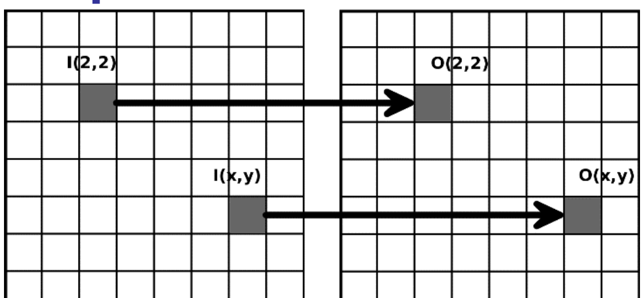
- Part 1: Overview Spatial Operations & Point Operations
- Part 1: Intensity transformations 1
- Part 2: Intensity transformations 2
- Part 3: Geometric transformations 1
- Part 4: Geometric transformations 2

Spatial Operations

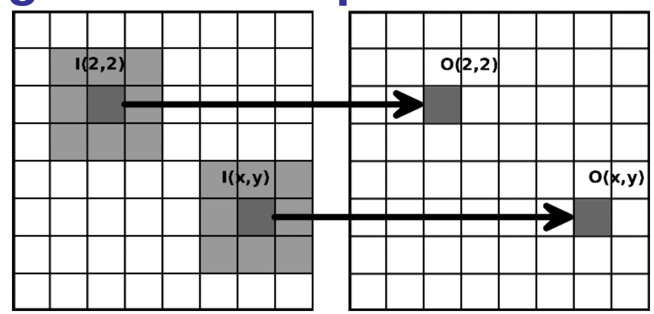
Spatial operations are operations directly applied to the pixels of an image.

We distinguish 4 categories:

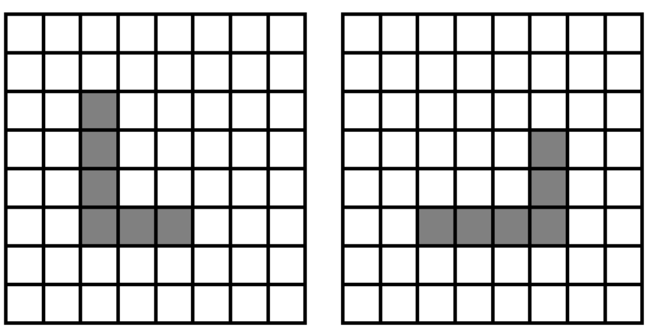
Point operations



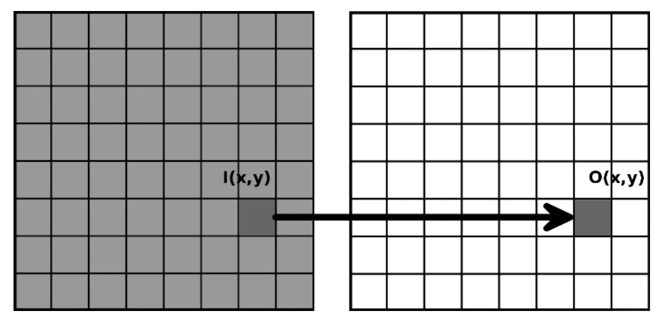
Neighborhood operations:



Geometric transformations

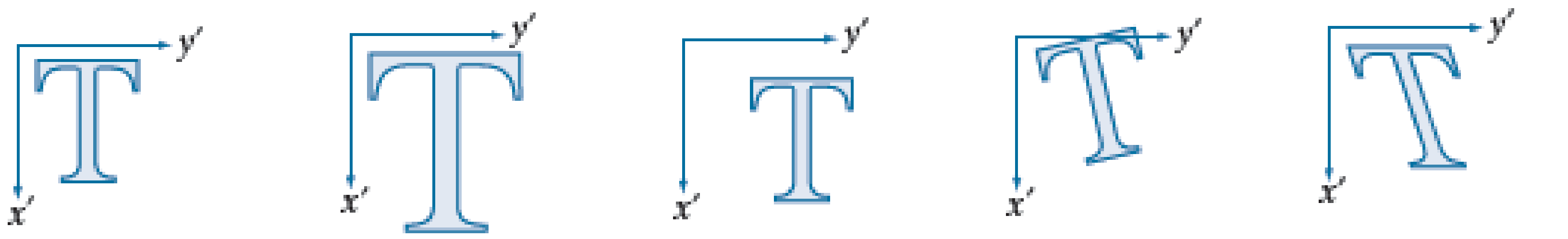


Global operations:



Geometric Transformations

- Geometric transformations modify the spatial arrangement of pixels
- Called also: rubber-sheet transformations (print image on rubber-sheet and stretch/shrink sheet according to some rules)
- Examples:



Scaling   Translation   Rotation   Shear (horizontal)

Geometric Transformations

Geometric transformations consist of two steps:

- Spatial transformation of coordinates
- Intensity interpolation, assigning intensity values to the transformed pixels

- Let us first look at the spatial transformation of coordinates…

Geometric Transformations

- Given the pixel coordinates (x, y) in the original image, and $(x^{'}, y^{'})$ in the transformed image, the transformation of coordinates (for all affine transformations except translation) is defined as:

$$\begin{bmatrix} x^{'} \\ y^{'} \end{bmatrix} = T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

- This corresponds to the equations:

$$x^{'} = t_{11} * x + t_{12} * y$$
$$y^{'} = t_{21} * x + t_{22} * y$$

Geometric Transformations

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

- A simple example: $(x', y') = (\frac{x}{2}, \frac{y}{2})$
- What does this transformation do ?
- Answer: It shrinks the image to half its size in both dimensions
- How does matrix T look like for this example?

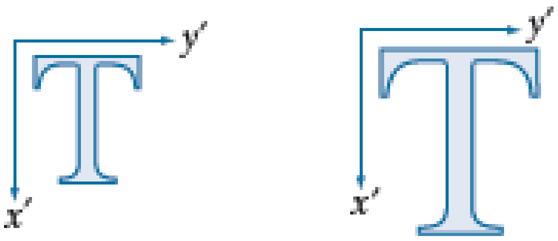
$$\begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Geometric Transformations

- In general:
- The transformation matrix for scaling is:

$$\begin{bmatrix} c_x & 0 \\ 0 & c_y \end{bmatrix}$$



- Thus, the coordinate equations are:

$$\begin{aligned} x' &= c_x x \\ y' &= c_y y \end{aligned}$$

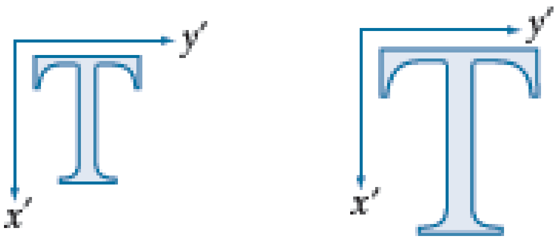
- What happens if c_x and c_y are < 1 (> 1)?
- Values of $c < 1$ shrink the image
- Values of $c > 1$ enlarge the image

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Geometric Transformations

- In general:
- The transformation matrix for scaling is:

$$\begin{bmatrix} c_x & 0 \\ 0 & c_y \end{bmatrix}$$



- Thus, the coordinate equations are:

$$x' = c_x x$$
$$y' = c_y y$$

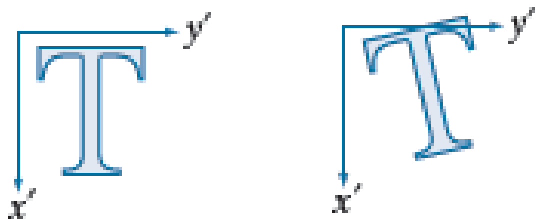
- What happens if we set $c_x = -1$ and $c_y = 0$?
- Answer: see exercises

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

Geometric Transformations

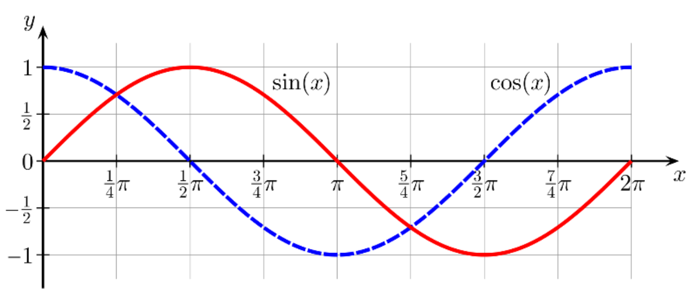
- In general:
- The transformation matrix for rotation by θ (about the origin) is:

$$\begin{bmatrix} cos(\theta) & -sin(\theta) \\ sin(\theta) & cos(\theta) \end{bmatrix}$$



- Thus, the coordinate equations are:

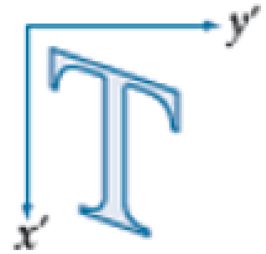
$$x' = xcos\theta - ysin\theta$$
$$y' = xsin\theta + ycos\theta$$



Geometric Transformations

- Exercise: how do the matrix and the equations look like for shearing in horizontal and vertical direction?

Shear (vertical)



Shear (horizontal)

Outline

- Part 1: Overview Spatial Operations & Point Operations
- Part 1: Intensity transformations 1
- Part 2: Intensity transformations 2
- Part 3: Geometric transformations 1
- Part 4: Geometric transformations 2

Image Processing 04
Transformations
Part 4

SS 2020

Prof. Dr. Simone Frintrop

Computer Vision Group, Department of Informatics
University of Hamburg, Germany

Outline

- Part 1: Overview Spatial Operations & Point Operations
- Part 1: Intensity transformations 1
- Part 2: Intensity transformations 2
- Part 3: Geometric transformations 1
- Part 4: Geometric transformations 2

Geometric Transformations

- Up to now, we had this equation for geometric transformations:

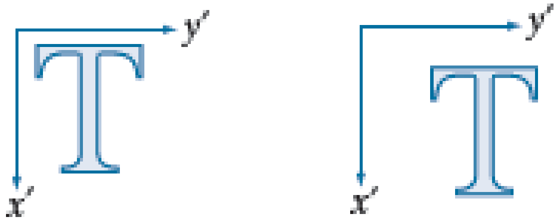
$$\begin{bmatrix} x' \\ y' \end{bmatrix} = T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

- This corresponds to the equations:

$$x' = t_{11} * x + t_{12} * y$$
$$y' = t_{21} * x + t_{22} * y$$

Geometric Transformations

- Let us look at *translation*.



- What do the equations look like?

$$x^{'} = x + t_x$$
$$y^{'} = y + t_y$$

$$x^{'} = t_{11} * x + t_{12} * y$$
$$y^{'} = t_{21} * x + t_{22} * y$$

Compare with previous equation

- What does the matrix look like?

$$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 1 & 1 \end{bmatrix}$$

Thus, we have a larger matrix for translation than for the other transformations
Unfortunate. We‘d like matrices of the same size

Geometric Transformations

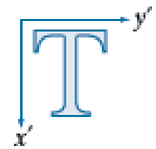
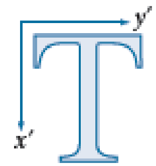
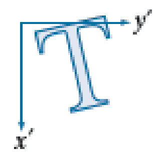
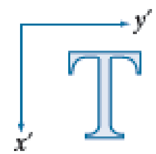
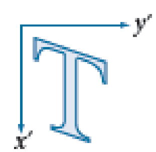
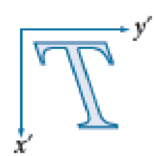
- Homogeneous coordinates allow us to express all four affine transformations using a 3x3 matrix.
- General form of transformation matrix:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = A \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Exercise: go through the transformations of part3. How does the 3x3 matrix look like for those?
- Advantage: for a sequence of operations, (resize, rotate, translate) we can create a 3x3 matrix as the product of the operations and apply this matrix to the image (exercise)

Geometric Transformations

TABLE 2.3
Affine transformations based on Eq. (2-45).

Transformation Name	Affine Matrix, A	Coordinate Equations	Example
Identity	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = y$	
Scaling/Reflection (For reflection, set one scaling factor to −1 and the other to 0)	$\begin{bmatrix} c_x & 0 & 0 \\ 0 & c_y & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = c_x x$ $y' = c_y y$	
Rotation (about the origin)	$\begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x\cos\theta - y\sin\theta$ $y' = x\sin\theta + y\cos\theta$	
Translation	$\begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x + t_x$ $y' = y + t_y$	
Shear (vertical)	$\begin{bmatrix} 1 & s_v & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x + s_v y$ $y' = y$	
Shear (horizontal)	$\begin{bmatrix} 1 & 0 & 0 \\ s_h & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$x' = x$ $y' = s_h x + y$	

[Gonzales/Woods]

Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG

Forward & inverse mapping

We can apply the transformation in two ways:

- **Forward mapping:** scan input image, compute corresponding values of output image. Problem?
  - Some pixels can transform to the same output pixel
  - Some output pixels may not be assigned a pixel
- **Inverse mapping:** scan output image, compute corresponding location in input image with

  $$(x, y) = A^{-1}(x', y')$$

  Interpolate among nearest input pixels.
  Inverse mappings are more efficient and used in many commercial implementations.

Inverse mapping

Example for inverse mapping:

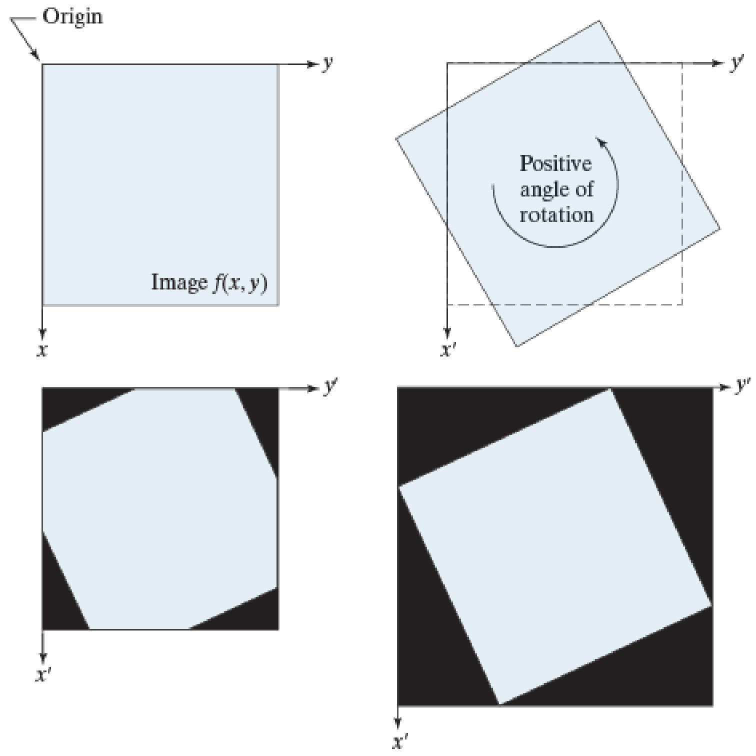
$$(x, y) = A^{-1}(x, y)$$

2x2 scaling matrix: $\begin{bmatrix} c_x & 0 \\ 0 & c_y \end{bmatrix}$

What does the inverse matrix look like?

Rotation

- Be careful: for some transformations, you might need an output image which is larger than the input:



a b
c d

FIGURE 2.41 (a) A digital image. (b) Rotated image (note the counterclockwise direction for a positive angle of rotation). (c) Rotated image cropped to fit the same area as the original image. (d) Image enlarged to accommodate the entire rotated image.

Outline

- Part 1: Overview Spatial Operations & Point Operations
- Part 1: Intensity transformations 1
- Part 2: Intensity transformations 2
- Part 3: Geometric transformations 1
- Part 4: Geometric transformations 2

Literature

Overview of spatial operations:

- Gonzales/Woods: end of chapter 2.6

Intensity transformations:

- Gonzales/Woods: chapter 3.1 & 3.2
- Nicolas Bertoa: Gamma correction, Youtube Video (2016): https://www.youtube.com/watch?v=FvwXQnP7nLQ&t=629s

Geometric transformations:

- Gonzales/Woods: end of chapter 2.6