EEE-6512: Image Processing and Computer Vision

August 23, 2017

Lecture #2: Introduction

Damon L. Woodard, Ph.D.
Dept. of Electrical and Computer Engineering
dwoodard@ece.ufl.edu

Chapter Outline

- Introduction to Image Processing and Analysis
- Examples of Applications of Image Processing and Analysis
- Basics of Image Processing
  - Accessing the image data
  - Image types
  - Conceptualizing images

Image Processing and Analysis

- **Image processing:** the field of study in which algorithms operate on input images to produce output images.
- **Image analysis:** the field of study in which algorithms operate on images to extract higher-level information.
- **Enhancement:** an image processing problem that involves transforming an input image into another image so as to improve its visual appearance.
- **Restoration:** an image processing problem that has as its purpose to restore an image that has been corrupted by some type of noise.

Image Processing and Analysis (cont'd)

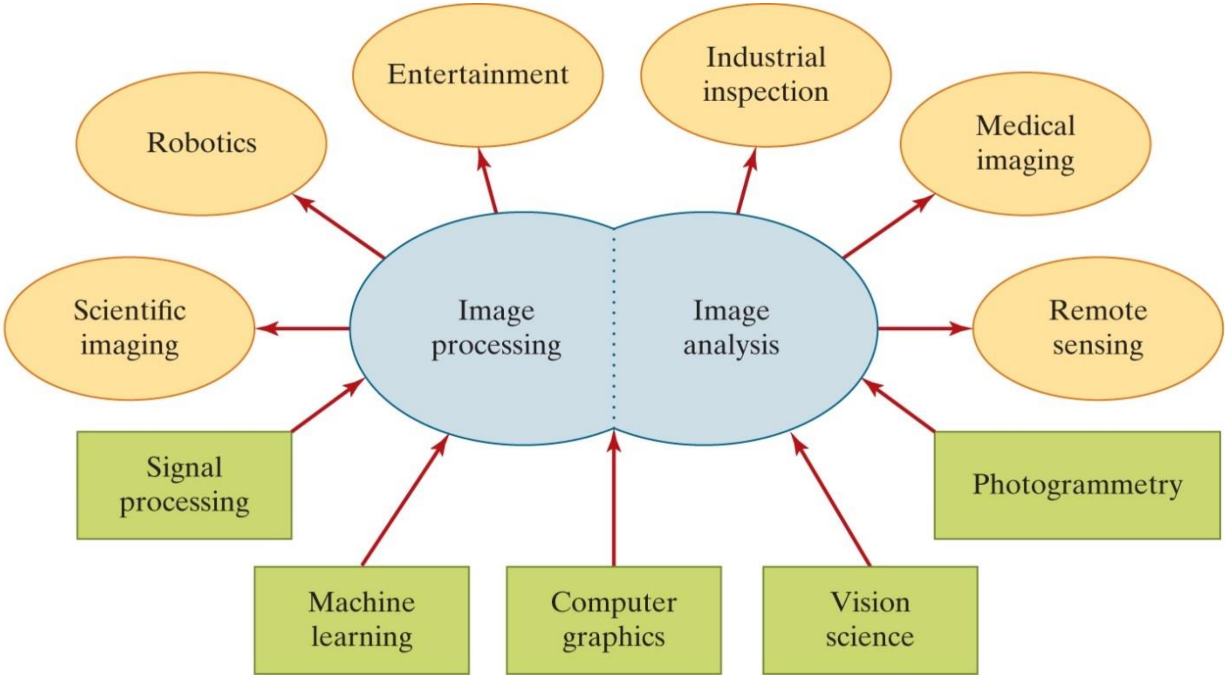
- **Compression:** an image processing problem that involves storing an image with fewer bits than are required by the original signal.
- **Segmentation:** an image analysis problem that involves the process of determining which pixels in an image belong together, that is, which pixels are projections of the same object in the scene.
- **Classification:** an image analysis problem that involves determining which pixels in an image belong to a model that has been created beforehand.

Image Processing and Analysis (cont’d)

- **Shape from X:** an image analysis problem that aims to recover the three-dimensional (3D) structure of the scene using any of a variety of techniques.
- **Machine vision:** refers to systems in an industrial setting in which the placement of the camera and lighting conditions can be controlled.
- **Computer vision:** refers to systems operating on images taken in unstructured settings, such as those taken by ordinary people in everyday life using their personal digital cameras.

History and Related Fields

Figure 1.3: Image processing and analysis, along with related fields (bottom rectangles) and sample applications (top ovals).



Sample Applications

- Industrial inspection
- Document image analysis
- Transportation
- Security and surveillance
- Remote sensing
- Scientific imaging
- Medical imaging
- Robotics

Image Basics

- The word *image* comes from the Latin word (*imago*), meaning “likeness.”
- **Digital image**: the image captured by a camera is usually a digitized version of some two-dimensional sensory input.

Accessing Image Data

- **Digital image:** a discrete two-dimensional array of values, like a matrix.
  - *Width:* refers to the number of columns in the image.
  - *Height:* refers to the number of rows in the image.
- **Aspect ratio:** width divided by height.
- **Pixel:** Each element of the array.
- An image is stored in memory as a 1D array in two ways:
  - *Column major order:* the first column is stored, then the second column, and so on until the final column.
  - *Row major order:* refers to storing the first row, then the second row, and so on until the final row.

Accessing Image Data

Figure 1.5: Top: Image as a 2D array, showing the 1D index of each pixel. Bottom: Internal representation of image as a 1D array using row major order.

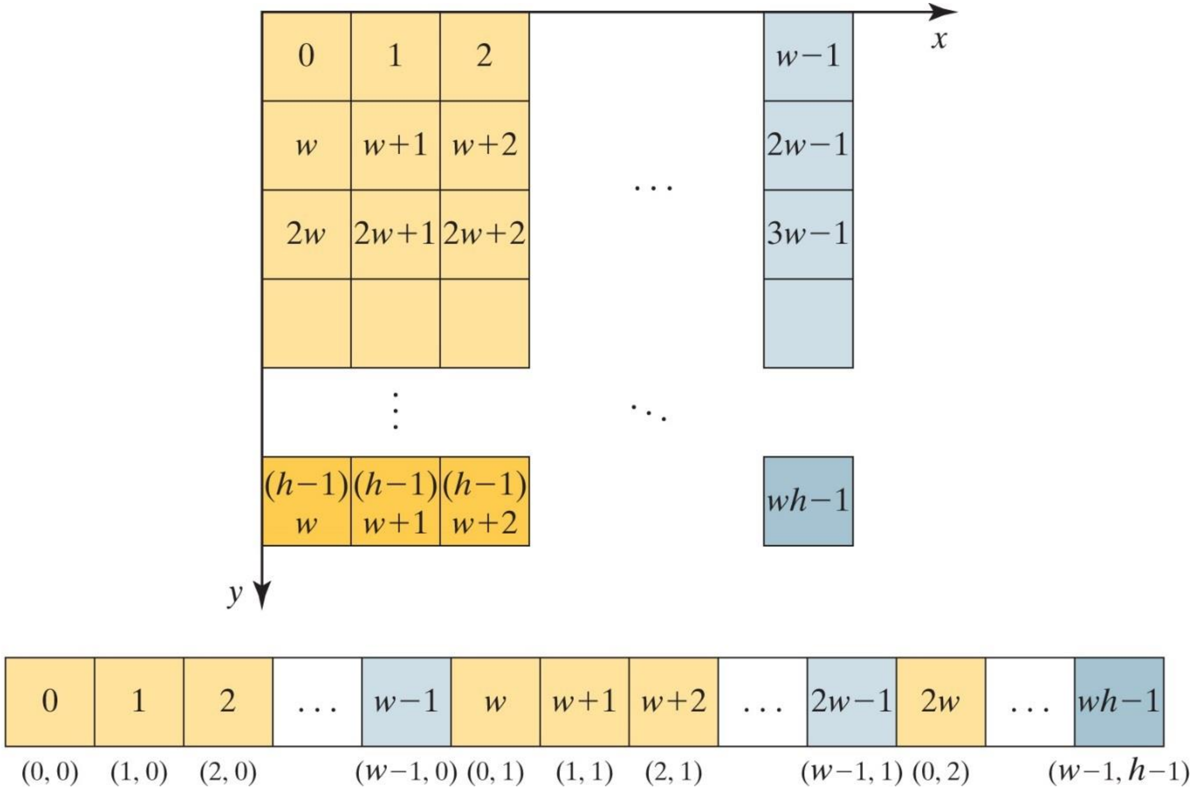


Image Types

- **Grayscale image**: the value of each pixel is a scalar indicating the amount of light captured.
  - These values are quantized into a finite number of discrete levels called **gray levels**.

- In an 8-bit grayscale image, a pixel whose value is 0 represents black, whereas a pixel whose value is 255 represents white.

- **RGB color image**: the pixel values are triples containing the amount of light captured in the three color channels: red, green, and blue.
  - **Interleaved:** all three values for one pixel are stored before the three values of the next pixel
  - **Planar:** the red, green, and blue channels are stored as separate one-byte-per-pixel images
  - **Alpha value or opacity:** used for blending multiple images

Image Types (cont'd)

- **Binary image:** The logical values can be stored using one bit per pixel, (0 for off or 1 for on), or they can be stored using one byte per pixel, where their values are usually 0 (hexadecimal 00) or 255 (hexadecimal FF).
- **Real-valued image**, or **floating-point image:** each pixel contains a real number.
  - The number is stored in the computer as an IEEE single- or double-precision floating point number
- **Integer-valued image:** the value of each pixel is an integer.

Image Types (cont'd)

- **Channels:** For example, an RGB color image is an 8-bit image with three channels.
  - **Complex-valued image:** arises from computing the Fourier transform of an image.
    - It contains two floating-point values for each pixel, one for the real component and one for the imaginary component.

	grayscale	RGB color	binary	integer-valued	real-valued	complex-valued
channels	1	3	1	1	1	2
bit depth	8	24	1	32/64	32/64	64/128
value range	$\{0, \ldots, 255\}$	$\{0, \ldots, 255\}^3$	$\{0, 1\}$	$\mathbb{Z}$	$\mathbb{R}$	$\mathbb{R}^2$

TABLE 1.3: Common image types, shown with the number of channels, the most commonly encountered bit depth (number of bits per pixel), and the set of possible values. In the final three columns this set is conceptual only, since the integers $\mathbb{Z}$ and real numbers $\mathbb{R}$ are infinite sets.

Conceptualizing Images

- A digital image is stored in the computer as a discrete array of values, which can be visualized either by:
  - considering the raw pixel values themselves arranged in a 2D lattice
  - equivalently as a height map, or 3D surface plot

- $I(x, y)$: evaluate the function at the position (x, y).

Conceptualizing Images (cont'd)

- Set of pixels
  - The grayscale image: Example:

$$I = \begin{bmatrix} 3 & 8 & 0 \\ 2 & 9 & 4 \end{bmatrix}$$

  - The binary image: Example:

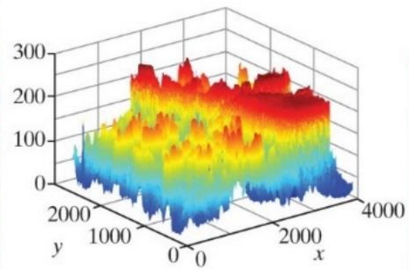
$$I = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

  - The following matrix **A** is an $m \times n$ matrix whose $(i, j)^{th}$ entry is given by a_{ij}

$$\mathbf{A}_{\{m \times n\}} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

Conceptualizing Images (cont'd)

Figure 1.6: Different ways to visualize an image: as a picture, as a height map, as an array of values, as a function, as a set, as a graph, and as a vector. The 5 × 4 array is a small portion of the image; the set contains the coordinates of all pixels in the array whose value is greater than 80; and the weights of the edges in the graph are the absolute differences between values in the array.



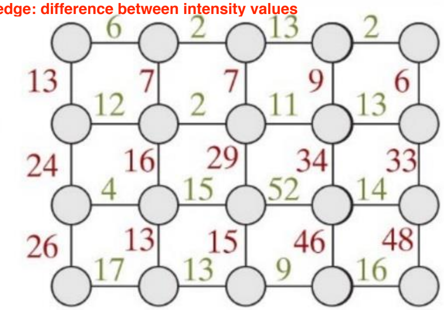
75	81	83	96	94
62	74	76	87	100
86	90	105	53	67
60	77	90	99	115

75
81
83
96
94
62
74
76
87
...
115

$$(x, y) \longrightarrow \boxed{I} \longrightarrow v$$

{(1, 0), (2, 0), (3, 0), (4, 0), (3, 1), (4, 1), (0, 2), (1, 2), (2, 2), (2, 3), (3, 3), (4, 3)}

edge: difference between intensity values



Up Next: Chapter #2 Fundamentals of Imaging

Questions?