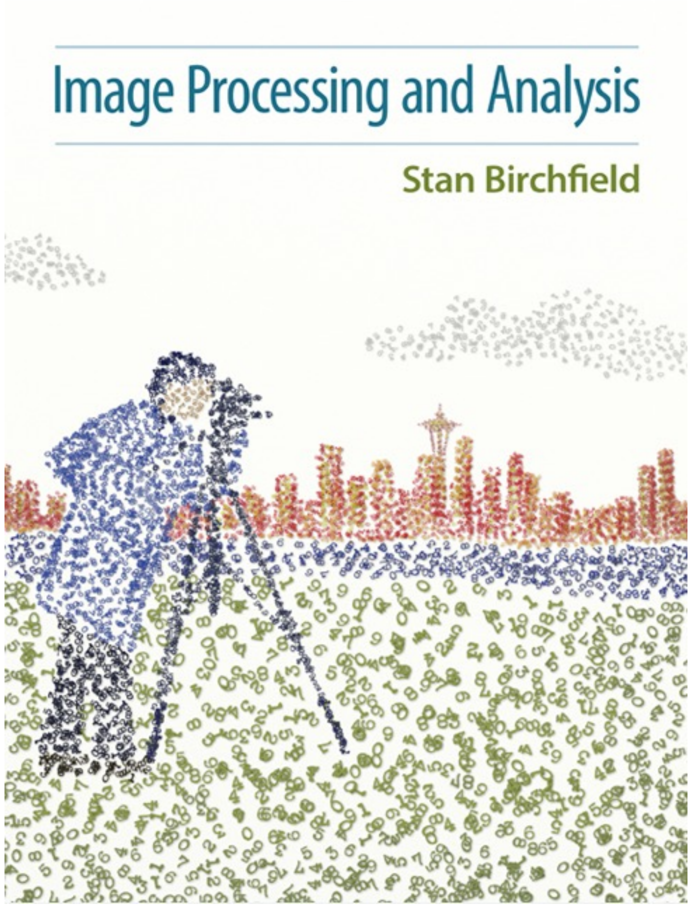
Prof. Kjersti Engan

ELE510 Image processing and computer vision

Image basics, (Chap 1, 9.1 Birchfield ++) 2023

University of Stavanger



Parts in course presentations is material from Cengage learning. It can be used for teaching, and it can be share it with students on access controlled web-sites (Canvas) for use in THIS course. **But it should not be copied or distributed further in any way** (by you or anybody).

Image Basics

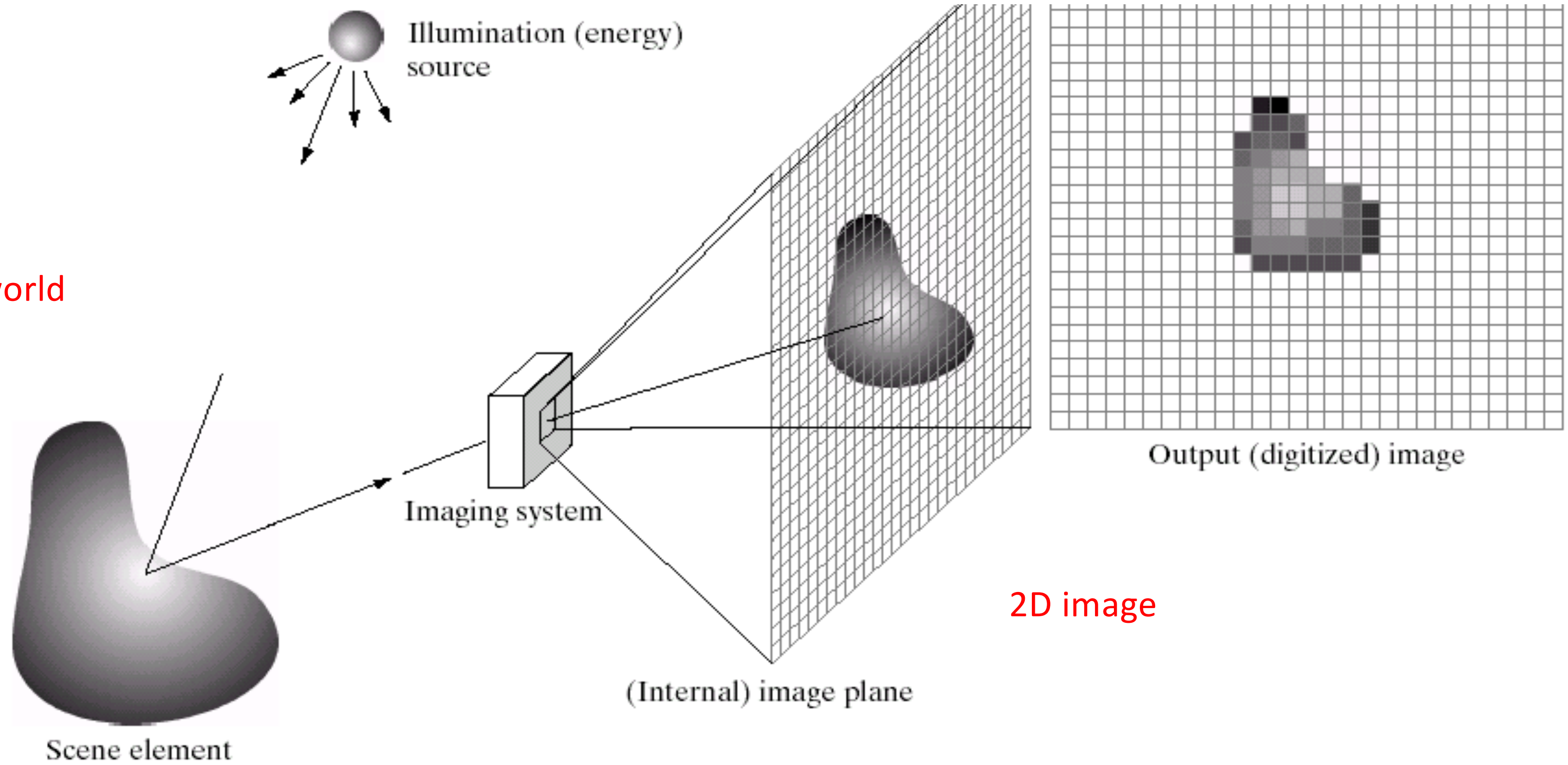
Image Basics

Topic in three points:

1. What is a digital image – how can we represent it?
2. How is color represented in a digital image?
3. What do we mean by resolution for a digital image?

Imaging System

3D world



2D image

What are digital images?

- An image is a 2-d rectilinear array of *pixels*
- The word pixel is derived from “picture element”.

Pixels as samples

- A pixel is a *sample* of a continuous function

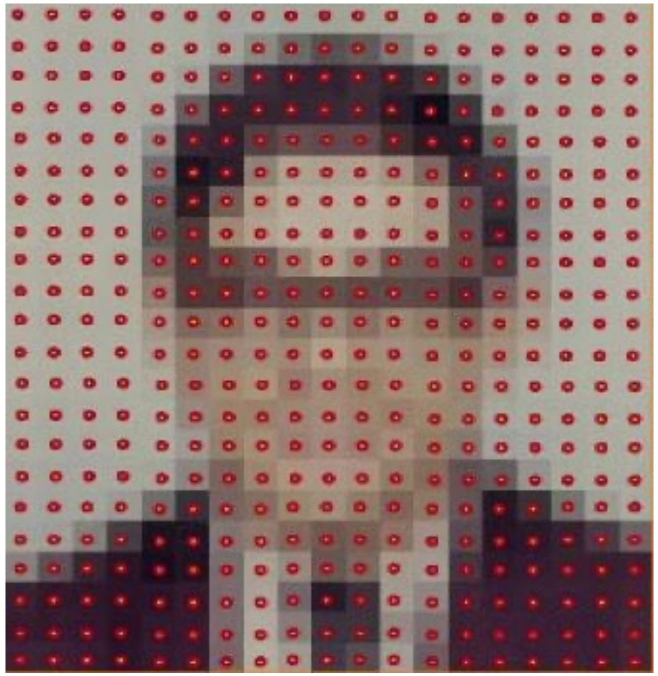
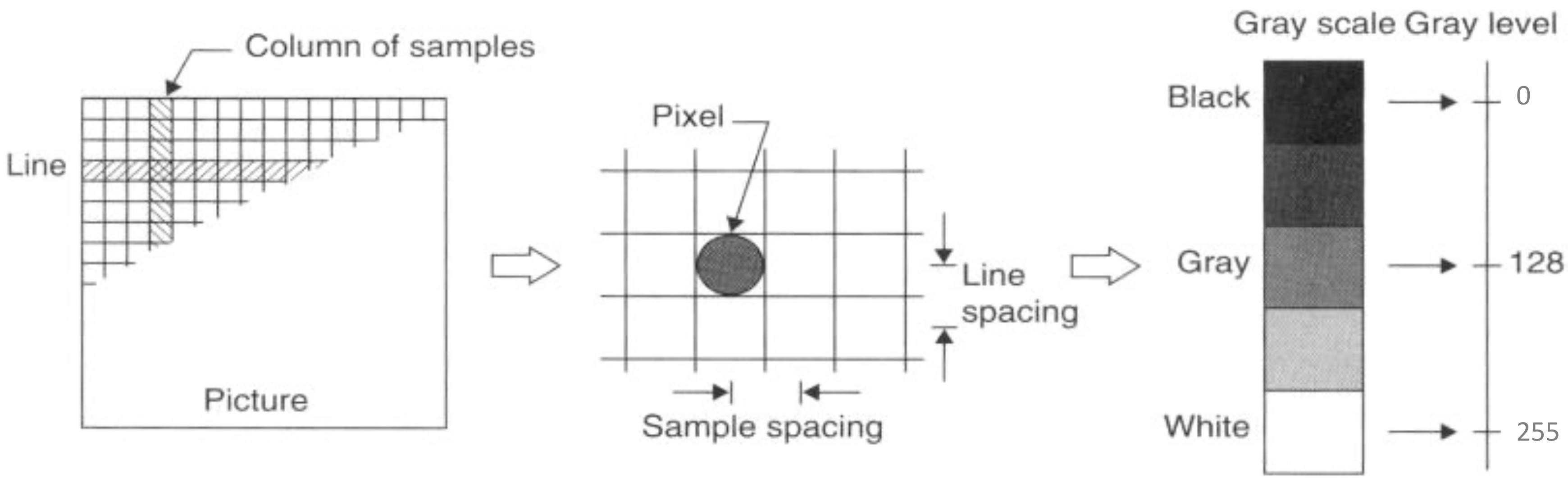
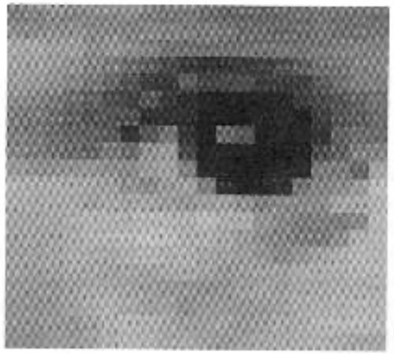


Image Digitization
(sampling and quantization)



Sampling: measuring the value of an image position at a finite number of points.
Quantization: representation of the measured value at the sampled point by a finite number of bits.

Representation in the computer



117	125	133	127	130	130	133	121	116	115	100	91	93	94	99	103	112	105	109	106
134	133	138	138	132	134	130	133	128	123	121	113	106	102	99	106	113	109	109	113
146	147	138	140	125	134	124	115	102	96	93	94	99	96	99	100	103	110	109	110
144	141	136	130	120	108	88	74	53	37	31	37	35	39	53	79	93	100	109	116
139	136	129	119	102	85	58	31	41	77	51	53	53	33	37	41	69	94	105	108
132	127	117	102	87	57	49	77	42	28	17	15	13	13	17	41	53	69	88	100
124	120	108	94	72	74	72	31	35	31	15	13	15	11	15	13	46	75	83	96
125	115	102	93	88	82	42	79	113	41	19	100	82	11	11	17	31	91	99	100
124	116	109	99	91	113	99	140	144	57	20	20	15	11	15	17	63	87	119	124
136	133	133	135	138	133	132	144	150	120	24	17	15	15	17	20	115	113	88	150
158	157	157	154	149	145	133	127	146	150	116	35	20	19	28	105	124	128	141	171
155	154	156	155	146	155	154	154	147	139	148	150	138	120	128	129	130	151	156	165
150	151	154	162	166	167	169	174	172	167	177	166	164	140	134	120	121	120	127	172
145	149	151	157	165	169	173	179	176	166	166	157	145	136	129	124	120	136	163	168
144	148	153	160	159	158	165	172	165	169	157	151	149	141	130	140	151	162	169	167
144	141	147	155	154	149	156	151	157	157	151	144	147	147	149	159	158	159	166	165
139	140	140	150	153	151	150	146	140	139	138	140	145	151	149	156	156	162	162	161
136	134	138	146	156	164	153	146	145	136	139	139	140	141	149	157	159	161	169	166
136	133	136	135	144	159	168	159	151	142	141	145	139	146	153	156	164	167	172	168
133	129	140	142	146	159	167	165	154	151	146	141	147	154	156	160	161	157	153	154

By a matrix of numbers
Integers

8 bit words, unsigned integers
0, 1, 2, ………... 255

Image representation – image as a function

- A digital image is represented as a 2D (light) intensity function $f(x,y)$, (x,y) are spatial coordinates.
- $f(x,y)$ has values proportional to the brightness at that point.
- $f(x,y)$ can represent a gray scale image -> one scalar value for each (x,y)
- $f(x,y)$ can represent a color image -> 3D vector with values for each (x,y), representing color. For example as RGB (red-green-blue)
- $f(x,y)$ can represent a hyperspectral image -> vector of size P>3 for each (x,y).

Image representation

An image can be represented as an M×N numerical array.

For multispectral (color) images there are one for each color/spectral band). Image as a function with numbering as in the book:

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

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.

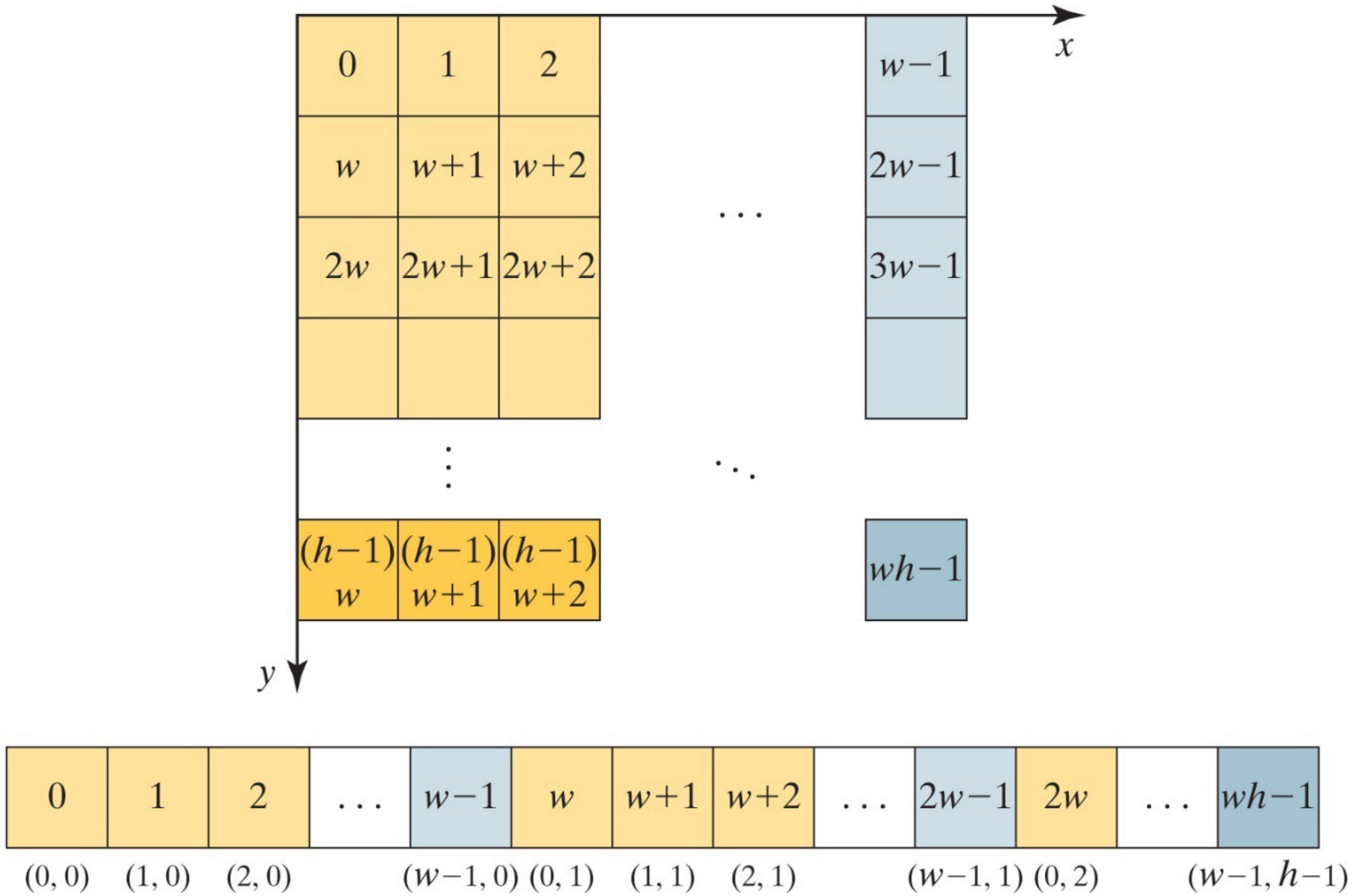


Image representation - image as a matrix

Image as a MxN matrix

$$f(row, column) = \begin{pmatrix} f(1,1) & f(1,2) & \cdots & f(1,N) \\ f(2,1) & f(2,2) & \cdots & f(2,N) \\ \vdots & \vdots & \ddots & \vdots \\ f(M,1) & f(M,2) & \cdots & f(M,N) \end{pmatrix}$$

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.

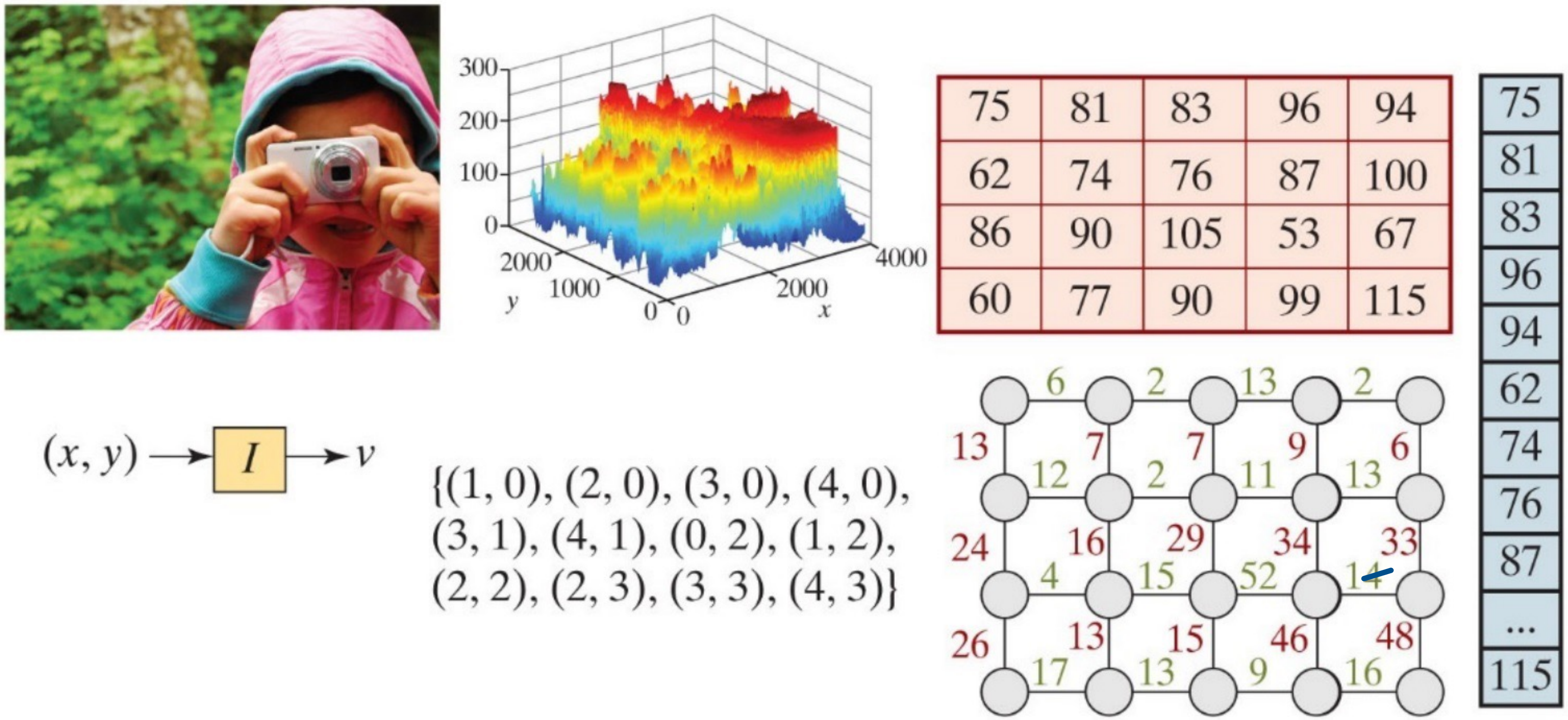


Image representation – remarks

- A digital image an be regaded as a function, as an array, as a matrix, graph etc. $f(x, y) \quad f_{x,y}$

- The value of a specific point: $f(x_i, y_j) = v$

- Often for image of size NxM: $N = 2^{n1}, M = 2^{n2}$

$$0 \leq f(x, y) \leq (G - 1) \qquad G = 2^m$$

m bit pr. pixel

How many pixels (NxM)?

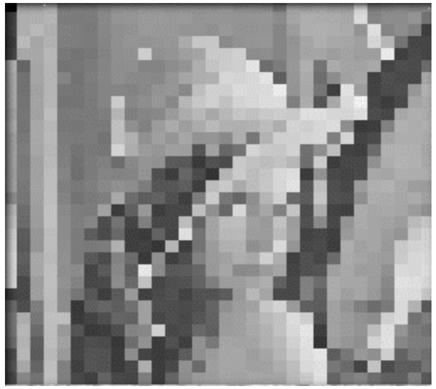
Reducing the number of pixels produces **pixelated image (here printed as the same size for illustration)**

original image

sampled by a factor of 2

sampled by a factor of 4

sampled by a factor of 8



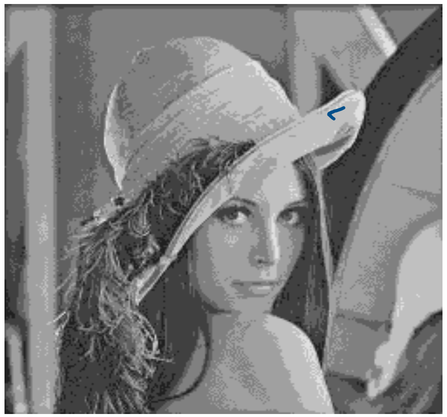
How many gray levels (number of bits pr pixel)?

- 256 gray levels (8bits/pixel)

32 gray levels (5 bits/pixel)

16 gray levels (4 bits/pixel)

- 8 gray levels (3 bits/pixel)

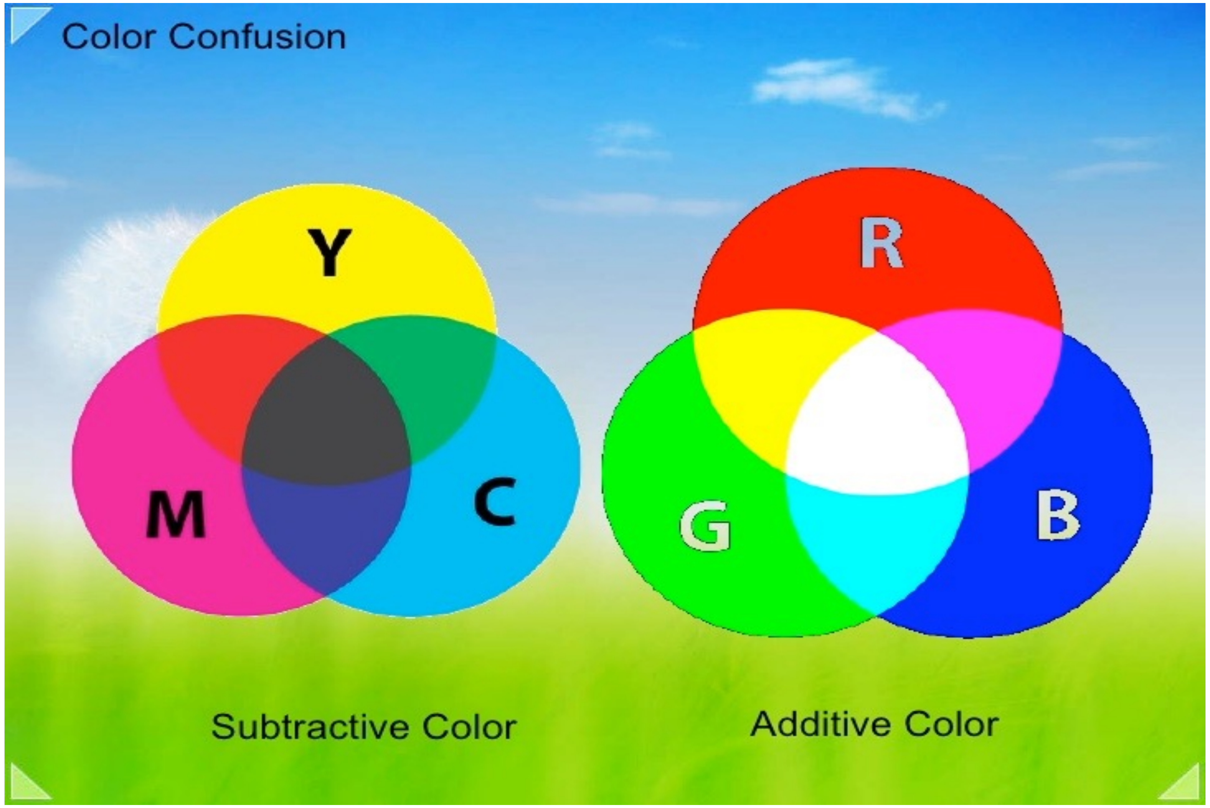


4 gray levels (2 bits/pixel)

2 gray levels (1 bit/pixel)

Reducing the number of grey levels produces **false contouring**

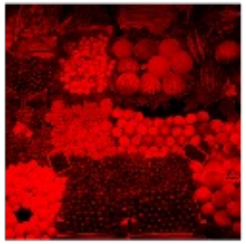
Color images



- RGB - additive color mixing: it describes what kind of *light* that must be *emitted* to produce a given color.
- CMY(K) - subtractive color mixing: it describes what kind of colored ink that has to be applied so the light *reflected* from the paper produces a given color.

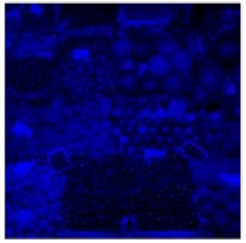
Color image – 3D vector for each pixel

RGB



Red

Green



Blue

Back to color: representation in the computer

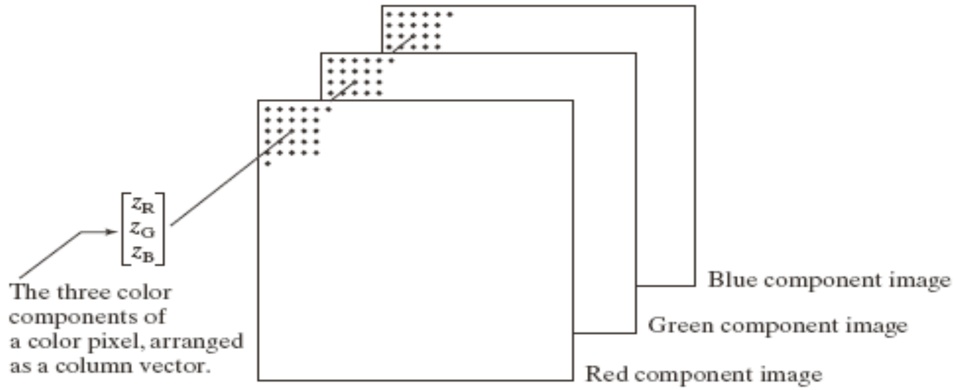


FIGURE 7.1 Schematic showing how pixels of an RGB color image are formed from the corresponding pixels of the three component images.

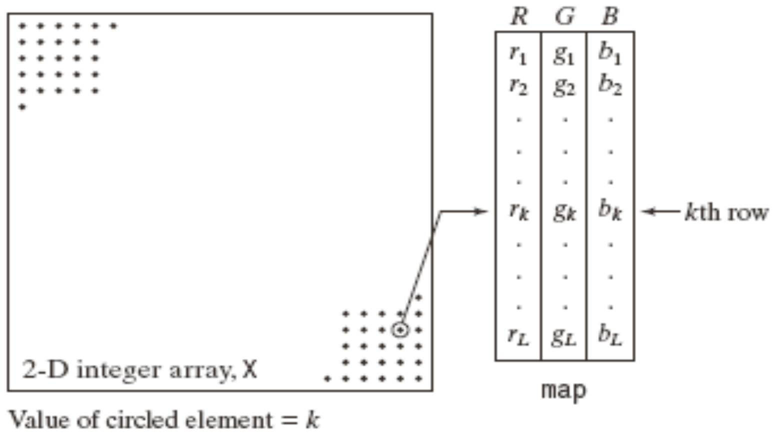
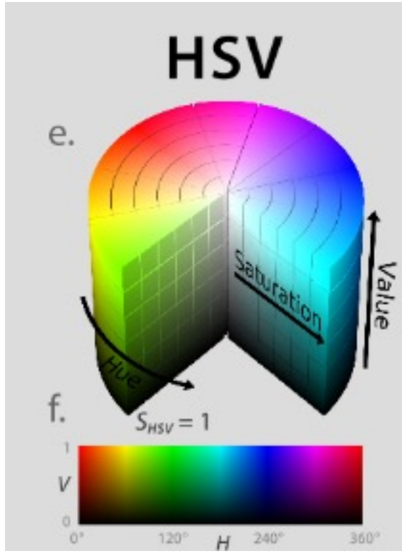


FIGURE 7.3 Elements of an indexed image. The value of an element of integer array X determines the row number in the color map. Each row contains an RGB triplet, and L is the total number of rows.

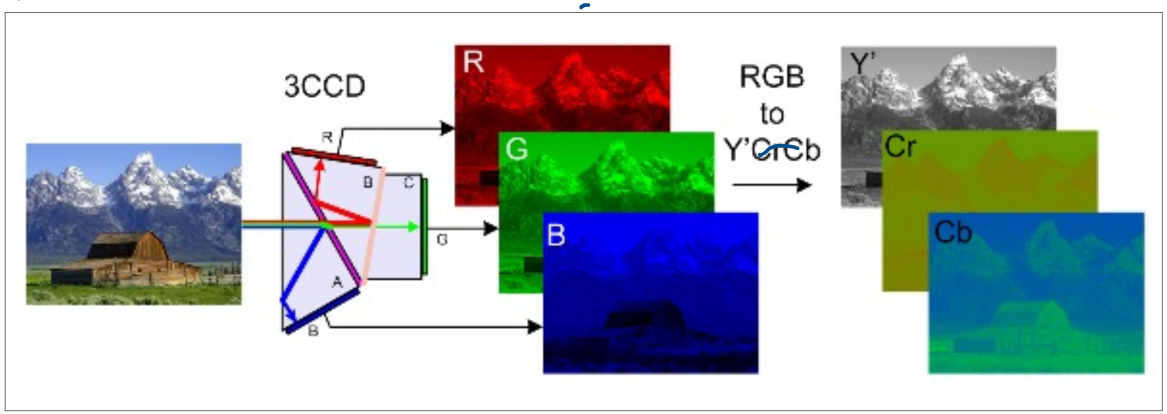
Color models, examples

- RGB – (red green blue), captured by camera

- HSV – Hue, saturation and Value(intensity). Close to how humans observe color
  - Hue - describe pure color
  - Saturation – degree of pure color diluted by white light
  - Value/Intensity – gray-level intensity information



- YCbCr - Used in television transmission, video/image coding etc. (MPEG, JPEG)
  - Y – luminance (gray level)
  - Cb and Cr – are the blue-difference and red-difference chrominance (color) components



Conversion expressions

RGB to **YCbCr**:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{pmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.000 \\ 112.000 & -93.786 & -18.214 \end{pmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

RGB to **HSV** and **HSV** to **RGB** : the expressions are more complicated especially for the hue value. Conversion functions exist.

More information about color spaces can ex. be found at http://en.wikipedia.org/wiki/Color_space

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 (2)

- **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 a single- or double-precision floating point number

- **Integer-valued image:** the value of each pixel is an integer.

	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.

Image Basics - summary

We will go through three points:

1. What is a digital image – how can we represent it?
   ✓ Pixels in a matrix / array, function

2. What do we mean by resolution for a digital image?
   ✓ Number of pixels AND bit-depth (nubmer of bits per pixel)

3. How is color represented in a digital image?
   ✓ 3 x matrix / array. One for Red, Green and Blue channels.