CS443: Digital Imaging and Multimedia
Histograms of Digital Images

Spring 2008
Ahmed Elgammal
Dept. of Computer Science
Rutgers University

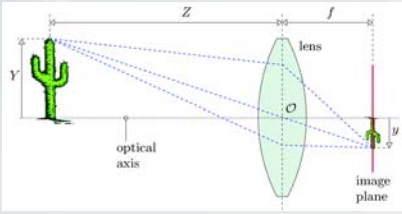
Outlines

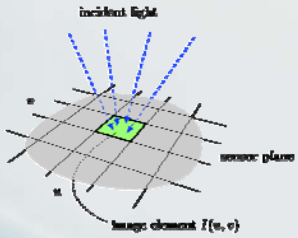
- Digitizing images
- Image histograms and its applications

- Sources:
  - Burger and Burge "Digital Image Processing" Ch. 4

Digitizing images

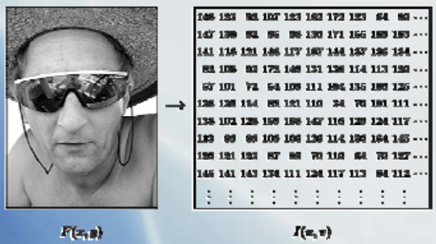
- What is projected on the image plan is a distribution of light energy that is:
  - Two-dimensional
  - Time-dependent
  - Continuous
- To go digital:
  - Spatial sampling
  - Temporal sampling
  - Quantization of pixel values



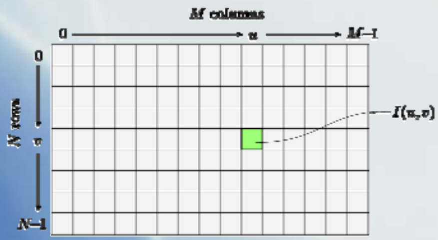


- Digital image: two-dimensional, ordered matrix of integers, i.e., a two-dimensional function of integer coordinates NxN that maps a range of image values P

$$I(u,v) \in \mathbb{P} \quad \text{and} \quad u,v \in \mathbb{N}.$$



- Image resolution: number of image elements per measurement.
- Image coordinate system



Grayscale (Intensity Images):

Chan.	*Bits/Pix.*	*Range*	*Use*
1	1	0...1	Binary image: document, illustration, fax
1	8	0...255	Universal: photo, scan, print
1	12	0...4095	High quality: photo, scan, print
1	14	0...16383	Professional: photo, scan, print
1	16	0...65535	Highest quality: medicine, astronomy

Color Images:

Chan.	*Bits/Pix.*	*Range*	*Use*
3	24	$[0...255]^3$	RGB, universal: photo, scan, print
3	36	$[0...4095]^3$	RGB, high quality: photo, scan, print
3	42	$[0...16383]^3$	RGB, professional: photo, scan, print
4	32	$[0...255]^4$	CMYK, digital prepress

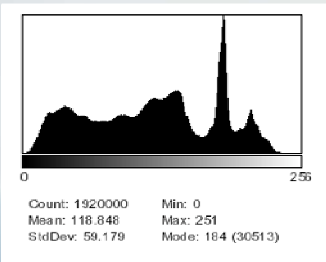
Special Images:

Chan.	*Bits/Pix.*	*Range*	*Use*
1	16	−32768...32767	Whole numbers pos./neg., increased range
1	32	$\pm 3.4 \cdot 10^{38}$	Floating point: medicine, astronomy
1	64	$\pm 1.8 \cdot 10^{308}$	Floating point: internal processing

Image Histograms

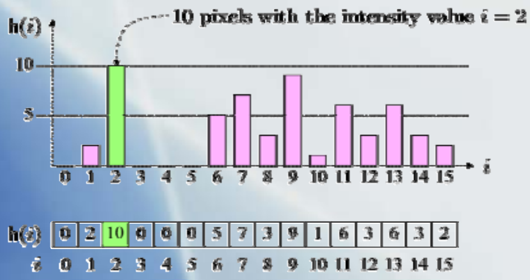
- Histograms are used to depict image statistics in an easily interpreted visual format
- Useful during image capturing: now already in digital cameras
- Used to improve the visual appearance of an image
- Can also be used to determine what type of processing has been applied to an image.

- Image histogram: describes the frequency of the intensity values that occur in an image



h(i) = the *number* of pixels in I with the intensity value i

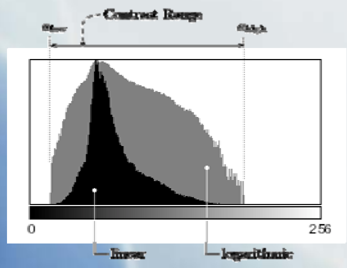
$$\mathsf{h}(i) = \mathrm{card}\{(u,v) \mid I(u,v) = i\}$$



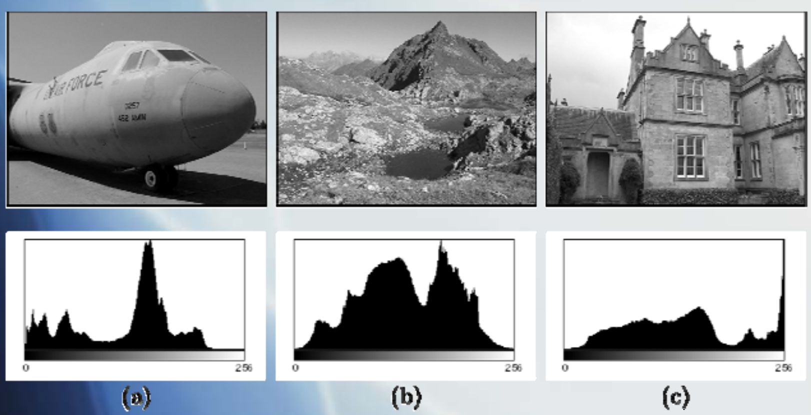
- Histograms don’t encode information about the spatial arrangement of pixels in the image
- We cannot reconstruct an image given only it’s histogram

Interpreting Histograms

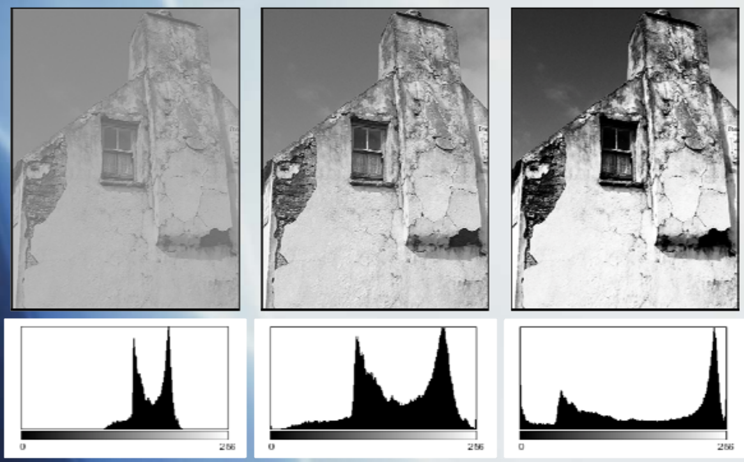
- Histograms depicts problems that originate during image acquisition
  - Exposure, contrast, dynamic range
- Histograms can be used to detect a wide range of image defects: saturation, spikes and gaps, impact of image compression



Histograms and Exposure

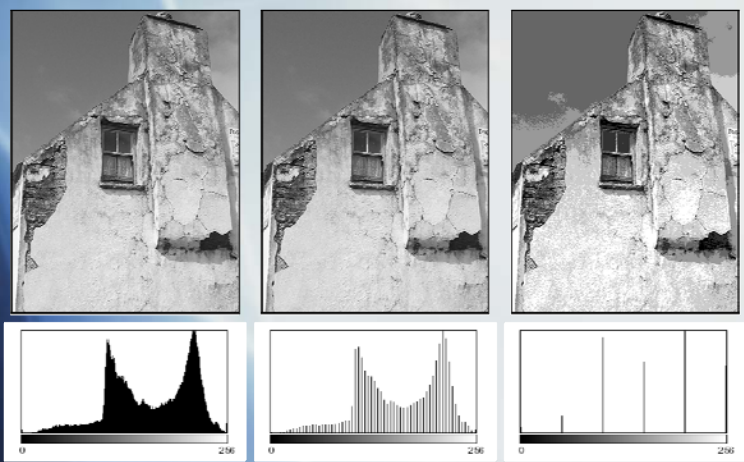


Histogram and Contrast



(a) (b) (c)

Dynamic Range

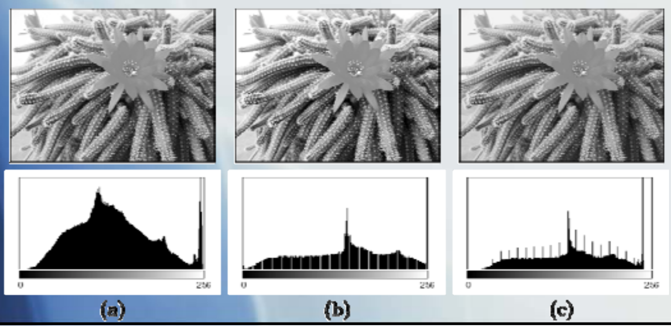


(a) (b) (c)

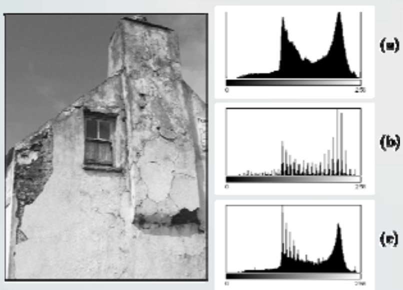
Dynamic Range: the number of distinct pixel value in an image

Detecting Image Defects

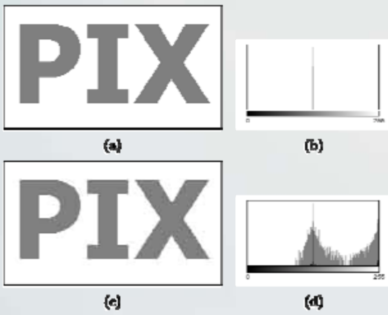
- There is no ideal or optimal histogram shape. It depends on the image and on the application
- Image Defects:
  - Saturation: the illumination values lying outside of the sensor's range are mapped to its maximum or minimum values: spike at the tails
  - Spikes and Gaps in manipulated images. Why?
  - Impact of image compression



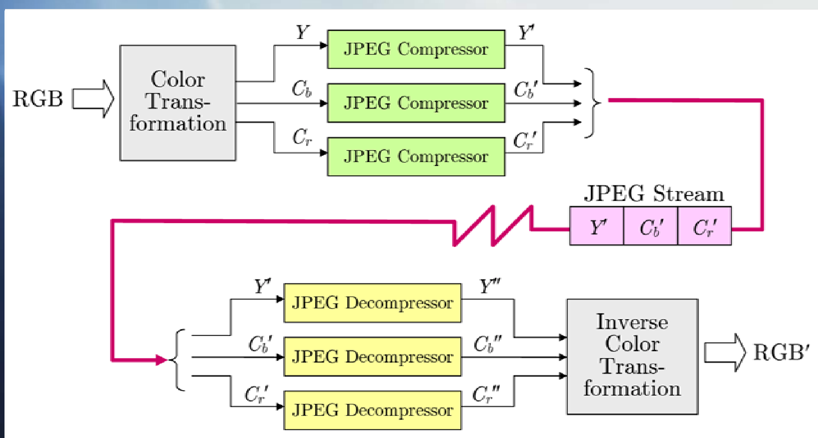
- Histograms show the impacts of image compression
- Ex: in GIF compression, the dynamic range is reduced to only few intensities (quantization)



- Ex: JPEG compression on a line graphics.



JPEG compression



Computing Histograms

```java
public class Compute_Histogram implements PlugInFilter {

    public int setup(String arg, ImagePlus img) {
        return DOES_8G + NO_CHANGES;
    }

    public void run(ImageProcessor ip) {
        int[] H = new int[256]; // histogram array
        int w = ip.getWidth();
        int h = ip.getHeight();

        for (int v = 0; v < h; v++) {
            for (int u = 0; u < w; u++) {
                int i = ip.getPixel(u,v);
                H[i] = H[i] + 1;
            }
        }
        ... //histogram H[] can now be used
    }

} // end of class Compute_Histogram
```

- Histograms of images with more than 8 bits:
  - Binning

$$h(j) = \mathrm{card}\,\{(u,v) \mid a_j \le I(u,v) < a_{j+1}\} \quad \text{for } 0 \le j < B$$

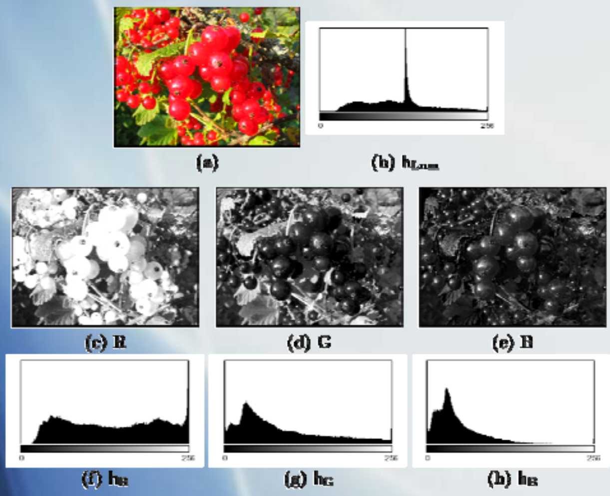
$$a_j = j \cdot \frac{K}{B} = j \cdot k_B$$

$$\begin{array}{lcrcl}
\mathsf{h}(0) & \leftarrow & 0 \le I(u,v) < & & 64 \\
\mathsf{h}(1) & \leftarrow & 64 \le I(u,v) < & & 128 \\
\mathsf{h}(2) & \leftarrow & 128 \le I(u,v) < & & 192 \\
\vdots & & \vdots \quad \vdots & & \vdots \\
\mathsf{h}(j) & \leftarrow & a_j \le I(u,v) < & & a_{j+1} \\
\vdots & & \vdots \quad \vdots & & \vdots \\
\mathsf{h}(255) & \leftarrow & 16320 \le I(u,v) < & & 16384
\end{array}$$

- Ex: B=256 for 14 bit image

K=16384, bin width = 64

Color Image Histograms



Color Image Histograms

- For color images, two kind of histograms:
  - Intensity histogram
  - Individual Color Channel Histograms
- Both provides useful information about lighting, contrast, dynamic range and saturation effects for individual color components
- They provide no information about the actual color distribution!

Cumulative Histograms

$$\mathsf{H}(i) = \sum_{j=0}^{i} \mathsf{h}(j) \quad \text{for } 0 \le i < K$$

$$\mathsf{H}(i) = \begin{cases} \mathsf{h}(0) & \text{for } i = 0 \\ \mathsf{H}(i-1) + \mathsf{h}(i) & \text{for } 0 < i < K \end{cases}$$

$$\mathsf{H}(K-1) = \sum_{j=0}^{K-1} \mathsf{h}(j) = M \cdot N$$