CS443: Digital Imaging and Multimedia
Point Operations in Digital Images

Spring 2008
Ahmed Elgammal
Dept. of Computer Science
Rutgers University

Outlines

- Point Operations
- Brightness and contrast adjustment
- Auto contrast
- Histogram equalization
- Histogram specifiation

- Source: Burger & Burge "Digital Image Processing"

Point Operations

- Point Operations perform a mapping of the pixel values without changing the size, geometry, or local structure of the image
- Each new pixel value $I'(u,v)$ depends on the previous value $I(u,v)$ at the same position and on a mapping function $f()$
- The function $f()$ is independent of the coordinates
- Such operation is called "homogeneous"

$$a' \leftarrow f(a)$$
$$I'(u,v) \leftarrow f(I(u,v))$$

Example of homogeneous point operations:
- Modifying image brightness or contrast
- Applying arbitrary intensity transformation (curves)
- Quantizing (posterizing) images
- Global thresholding
- Gamma correction
- Color transformations

- A nonhomogeneous point operation *g()* would also take into account the current image coordinate *(u,v)*

$$a' \;\leftarrow\; g(a, u, v)$$
$$I'(u, v) \;\leftarrow\; g\left(I(u, v), u, v\right)$$

- Changing contrast and brightness

$$f_{\text{contr}}(a) = a \cdot 1.5 \qquad \text{and} \qquad f_{\text{bright}}(a) = a + 10$$
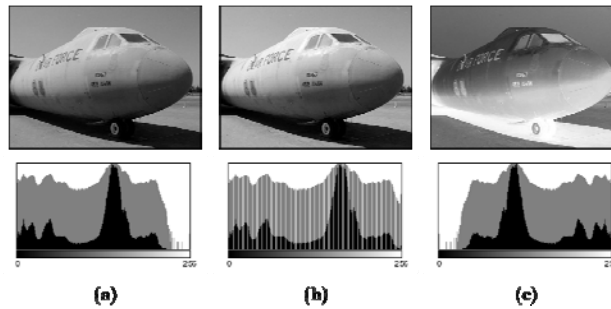
- Limiting Results by Clamping

```
if (a > 255)  a = 255;
if (a < 0)   a = 0;
```

```
1   public void run(ImageProcessor ip) {
2     int w = ip.getWidth();
3     int h = ip.getHeight();
4
5     for (int v = 0; v < h; v++) {
6       for (int u = 0; u < w; u++) {
7         int a = (int) (ip.get(u, v) * 1.5 + 0.5);
8         if (a > 255)
9           a = 255;   // clamp to maximum value
10        ip.set(u, v, a);
11      }
12    }
13  }
```

- Inverting Images

$$f_{\text{invert}}(a) = -a + a_{\max} = a_{\max} - a$$
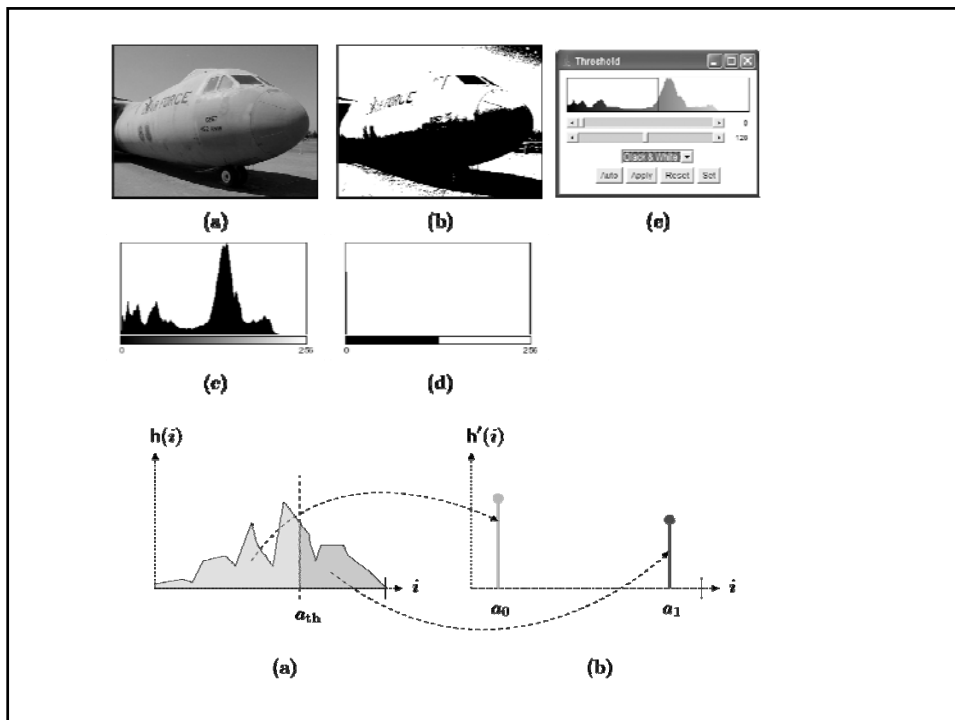

(a)   (b)   (c)

Threshold Operation

- Thresholding an image is a special type of quantization that separates the pixel values in two classes, depending on a given threshold value a_{th}
- The threshold function maps all the pixels to one of two fixed intensity values a_o, a_1

$$f_{\text{threshold}}(a) = \begin{cases} a_0 & \text{for } a < a_{\text{th}} \\ a_1 & \text{for } a \geq a_{\text{th}} \end{cases}$$

$$0 < a_{\text{th}} \leq a_{\max}$$

- Example: binarization: $a_o=0, a_1=1$

4

(a)   (b)   (c)

(c)   (d)

$h(i)$ $h'(i)$

a_th a_0 a_1

(a)   (b)

Point Operations and Histograms

- The effect of some point operations on histograms are easy to predict: ex: increasing the brightness, raising the contrast, inverting an image
- Point operations can only shift and merge histogram entries
- Operations that result in merging histogram bins are irreversible

$h'(i)$ $h'(a_2) \leftarrow h(a_1) + h(a_2)$

(a)   (b)
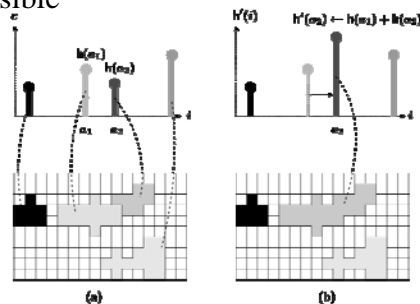
Automatic Contrast Adjustment

- Auto-contrast: a point operation that modifies the pixels such that the available range of values is fully covered.

- Linear stretching of the intensity range - can result in gaps in the new histogram
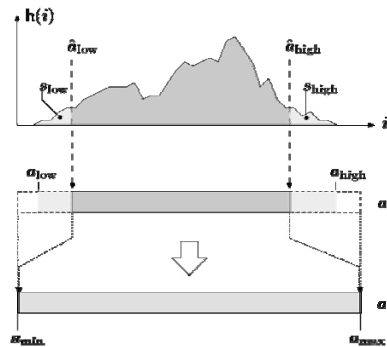
$$f_{ac}(a) = a_{min} + (a - a_{low}) \cdot \frac{a_{max} - a_{min}}{a_{high} - a_{low}}$$

$$f_{ac}(a) = (a - a_{low}) \cdot \frac{255}{a_{high} - a_{low}}$$

(a)   (b)   (c)
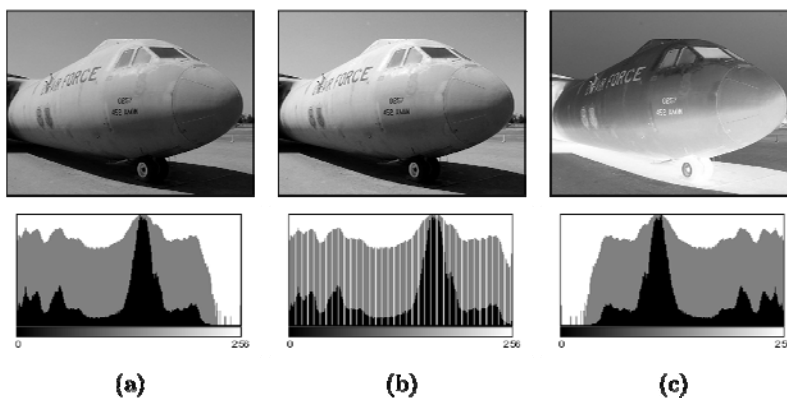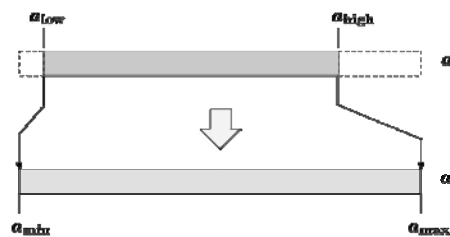
Better Auto-contrast

- It's better to map only a certain range of the values and get rid of the tails (usually noise) based on predefined percentiles (s_{low}, s_{high})

$$\hat{a}_{\mathrm{low}} = \min\big\{\, i \mid \mathsf{H}(i) \geq M\cdot N\cdot s_{\mathrm{low}} \,\big\}$$

$$\hat{a}_{\mathrm{high}} = \max\big\{\, i \mid \mathsf{H}(i) \leq M\cdot N\cdot (1-s_{\mathrm{high}}) \,\big\}$$

$$f_{\mathrm{mac}}(a) = \begin{cases} a_{\mathrm{min}} & \text{for } a \leq \hat{a}_{\mathrm{low}} \\ a_{\mathrm{min}} + (a - \hat{a}_{\mathrm{low}})\cdot \dfrac{a_{\mathrm{max}} - a_{\mathrm{min}}}{\hat{a}_{\mathrm{high}} - \hat{a}_{\mathrm{low}}} & \text{for } \hat{a}_{\mathrm{low}} < a < \hat{a}_{\mathrm{high}} \\ a_{\mathrm{max}} & \text{for } a \geq \hat{a}_{\mathrm{high}} \end{cases}$$

Histogram Equalization

- Adjust two different images in such a way that their resulting intensity distribution are similar
- Useful when comparing images to get rid of illumination variations
- The goal is to find and apply a point operation such that the histogram of the modified image approximates a uniform distribution.
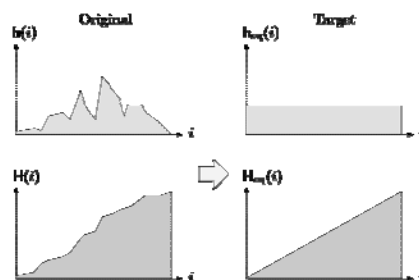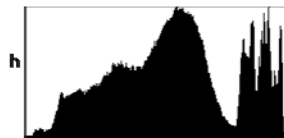
- Linear Histogram equalization
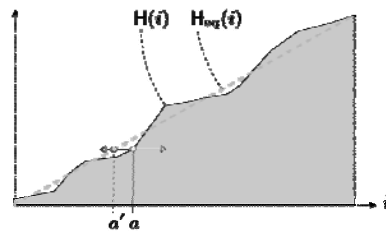
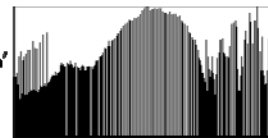$$f_{eq}(a) = \left\lfloor H(a) \cdot \frac{K-1}{MN} \right\rfloor$$

Histogram Specification

- Real images never show uniform distribution
- In most real images the distribution of pixel values is more similar to a Gaussian Distribution
- Histogram specification modifies the image to match an arbitrary intensity distribution, including the histogram of a given image.
- Also depends on the alignment of the cumulative histograms by applying a homogeneous point operation.
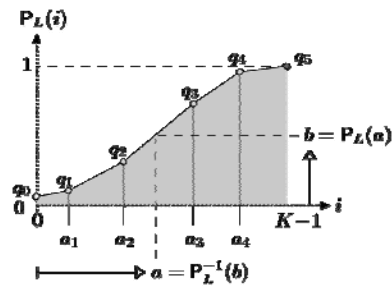
Histogram Specification

- Find a mapping such that

$$a' = f_{hs}(a)$$

$$P_{A'}(i) \approx P_R(i) \quad \text{for} \ 0 \leq i < K$$

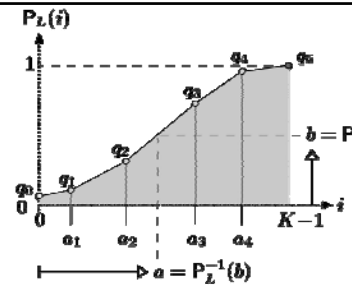$$f_{hs}(a) = a' = P_R^{-1}(P_A(a))$$
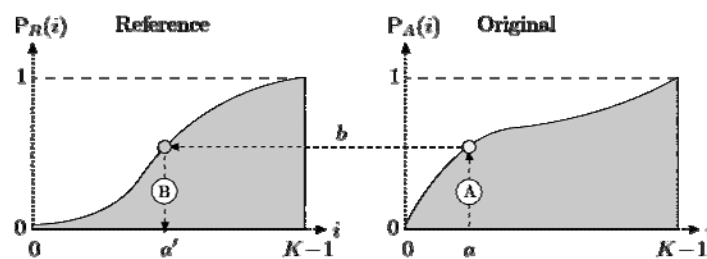
Adjusting piecewise linear distribution

$$\mathcal{L} = [\langle a_0, q_0 \rangle, \langle a_1, q_1 \rangle, \ldots \langle a_k, q_k \rangle, \ldots \langle a_N, q_N \rangle]$$

$$\langle 0, q_0 \rangle \qquad \text{and} \qquad \langle K-1, 1 \rangle$$

$$P_L(i) = \begin{cases} q_m + (i - a_m) \cdot \dfrac{(q_{m+1} - q_m)}{(a_{m+1} - a_m)} & \text{for } 0 \leq i < K-1 \\ 1 & \text{for } i = K-1 \end{cases}$$

$$P_L^{-1}(b) = \begin{cases} 0 & \text{for } 0 \leq b < P_L(0) \\ a_n + (b - q_n) \cdot \dfrac{(a_{n+1} - a_n)}{(q_{n+1} - q_n)} & \text{for } P_L(0) \leq b < 1 \\ K-1 & \text{for } b \geq 1 \end{cases}$$
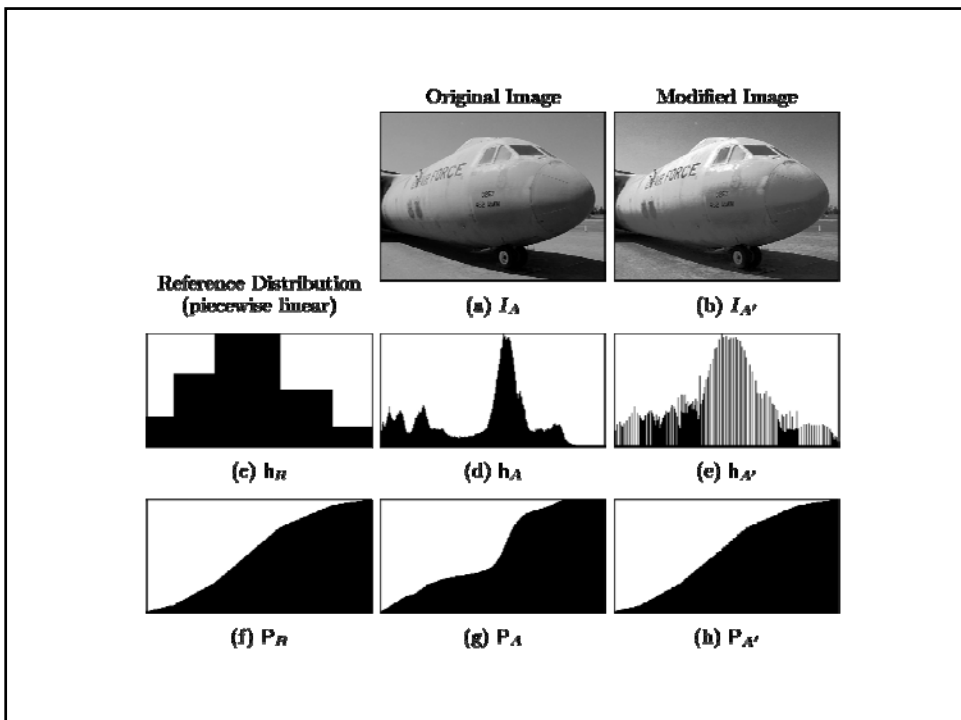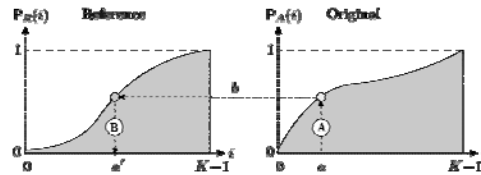
$$n = \max\{j \in \{0, \ldots N-1\} \mid q_j \leq b\}$$

```
1:  PiecewiseLinearHistogram(h_A, L_R)
        h_A: histogram of the original image.
        L_R: reference distribution function, given as a sequence of N+1
        control points L_R = [⟨a_0, q_0⟩, ⟨a_1, q_1⟩, ... ⟨a_N, q_N⟩], with 0 ≤ a_k < K
        and 0 ≤ q_k ≤ L.
2:      Let K ← Size(h_A)
3:      Let P_A ← Cdf(h_A)                          ▷ cdf for h_A (Alg. 5.1)
4:      Create a table f_hs[] of size K             ▷ mapping function f_hs
5:      for a ← 0 ... (K−1) do
6:          b ← P_A(a)
7:          if (b ≤ q_0) then
8:              a' ← 0
9:          else if (b ≥ 1) then
10:             a' ← K−1
11:         else
12:             n ← N−1
13:             while (n ≥ 0) ∧ (q_n > b) do         ▷ find line segment in L_R
14:                 n ← n−1
15:             a' ← a_n + (b − q_n) · (a_{n+1} − a_n)/(q_{n+1} − q_n)   ▷ see Eqn. (5.23)
16:         f_hs[a] ← a'
17:     return f_hs.
```


Reference Distribution (piecewise linear)

Original Image — (a) I_A

Modified Image — (b) $I_{A'}$

(c) h_R

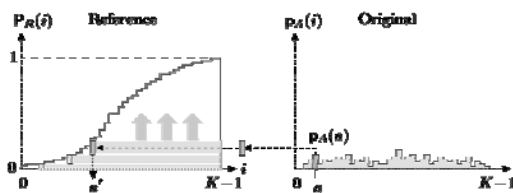(d) h_A

(e) $h_{A'}$

(f) P_R

(g) P_A

(h) $P_{A'}$

11

Adjusting to a given histogram

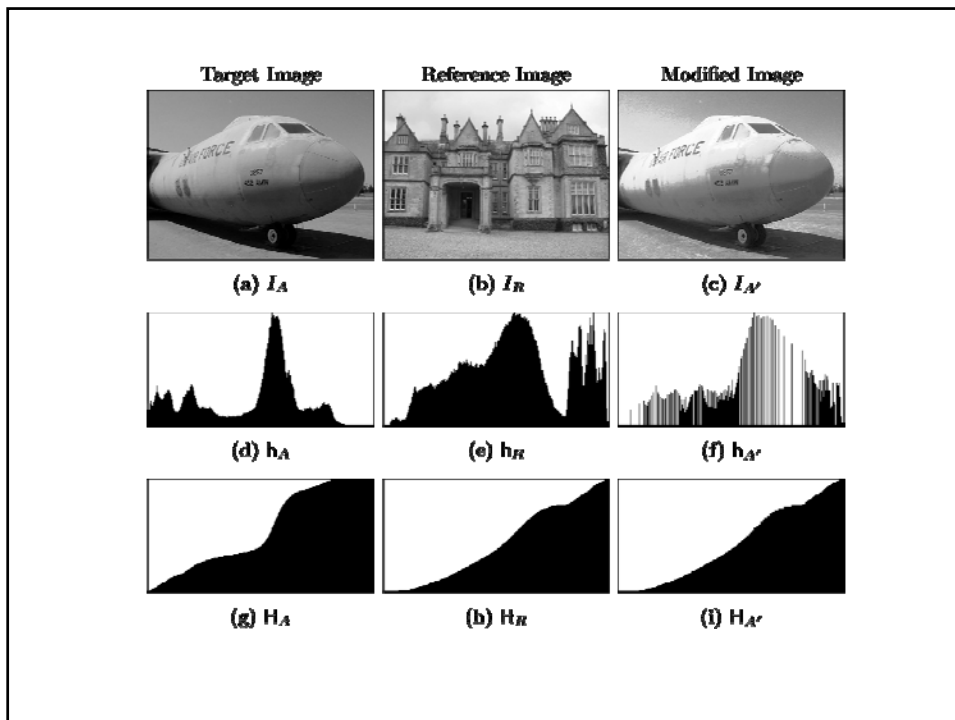$$f_{\text{hs}}(a) = a' = \min\{\, j \mid (0 \le j < K) \wedge \big(\mathsf{P}_A(a) \le \mathsf{P}_R(j)\big)\,\}$$
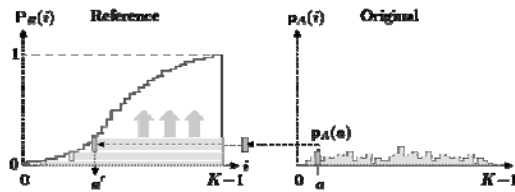

```
1:  MATCHHISTOGRAMS(h_A, h_R)
       h_A: histogram of the target image
       h_R: reference histogram (of same size as h_A)
2:     Let K ← Size(h_A)
3:     Let P_A ← Cdf(h_A)                    ▷ cdf for h_A (Alg. 5.1)
4:     Let P_R ← Cdf(h_R)                    ▷ cdf for h_R (Alg. 5.1)
5:     Create a table f_hs[ ] of size K       ▷ pixel mapping function f_hs
6:     for a ← 0 … (K−1) do
7:         j ← K−1
8:         repeat
9:             f_hs[a] ← j
10:            j ← j − 1
11:        while (j ≥ 0) ∧ (P_A(a) ≤ P_R(j))
12:    return f_hs.
```

	Original Image	Gaussian ($\sigma = 50$)	Gaussian ($\sigma = 100$)

(a) I_A (b) I_{G50} (c) I_{G100}

Reference Histogram
$h_R(i)$

Cumulative
Reference Histogram
$H_R(i)$

(d)   (e)

(f) h_A (g) h_{G50} (h) h_{G100}

(i) H_A (j) H_{G50} (k) H_{G100}

Target Image	Reference Image	Modified Image

(a) I_A (b) I_R (c) $I_{A'}$

(d) h_A (e) h_R (f) $h_{A'}$

(g) H_A (h) H_R (i) $H_{A'}$
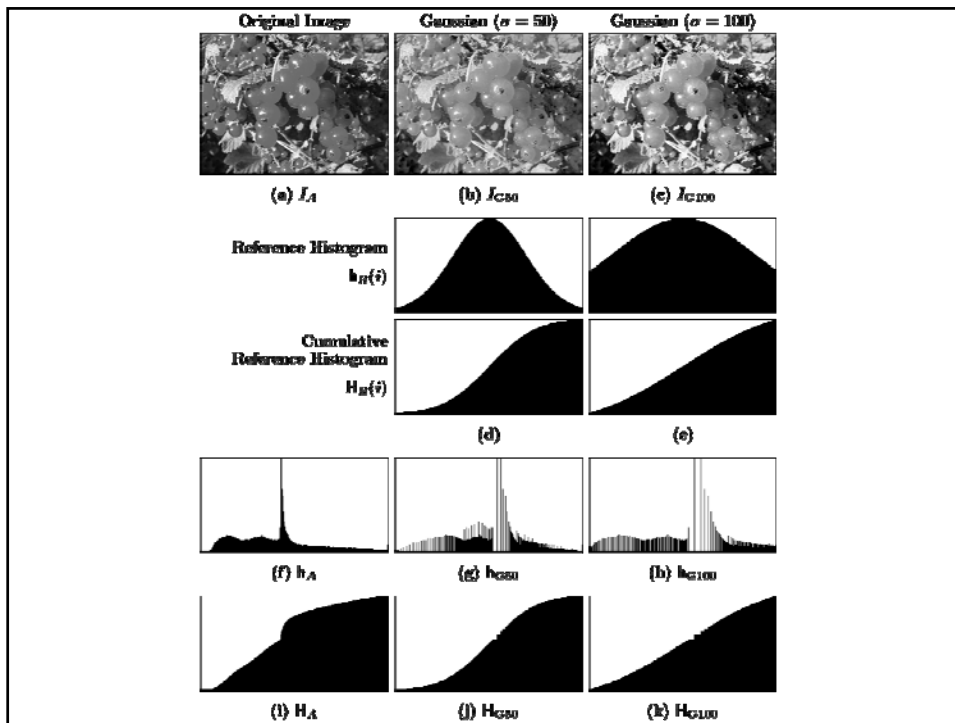
Gamma Correction

- What is the relation between the amount of light falling onto a sensor and the "intensity" or "brightness" measured at the corresponding pixel.
- What is the relation between the intensity of a pixel and the actual light emanating from that pixel on the display?
- The relation between a pixel value and the corresponding physical quantity is usually complex and nonlinear.
- Approximation ?

What is Gamma?

- Originates from analog photography
- Exposure function: the relationship between the logarithmic light intensity and the resulting film density.
- Gamma is the slope of the linear range of the curve.
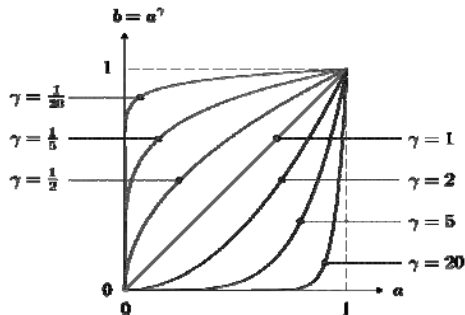- The same in TV broadcasting
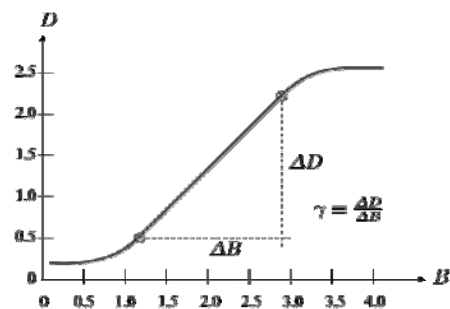
The Gamma function

- Gamma function is a good approximation for the exposure curve.
- The inverse of a Gamma function is another gamma function with

$$\bar{\gamma} = 1/\gamma$$

- Gamma of CRT and LCD monitors: 1.8-2.8 (typically 2.4)

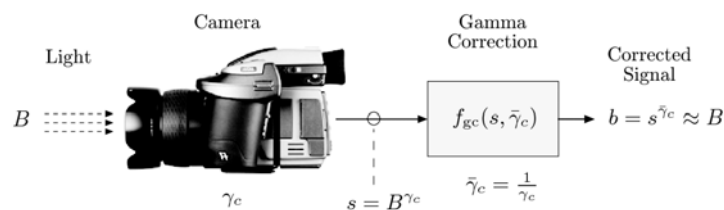$$b = f_\gamma(a) = a^\gamma \quad \text{for } a \in \mathbb{R}, \gamma > 0$$

$$a = f_\gamma^{-1}(b) = b^{1/\gamma}$$

$$f_\gamma^{-1}(b) = f_{\bar{\gamma}}(b)$$

$$\bar{\gamma} = 1/\gamma$$

Gamma Correction

- Obtain a measurement b proportional to the original light intensity B by applying the inverse gamma function
- This is important to achieve a device independent representation

$$s = B^{\gamma_c} \qquad f_{gc}(s, \bar{\gamma}_c) \qquad b = s^{\bar{\gamma}_c} \approx B$$

$$\bar{\gamma}_c = \frac{1}{\gamma_c}$$

Gamma Correction

$\gamma_c = 1.3$

Storage

$\gamma_m = 2.6$

$\bar{\gamma}_c = \frac{1}{1.3}$

$\bar{\gamma}_m = \frac{1}{2.6}$

Processing

$\bar{\gamma}_s = \frac{1}{1.9}$

$\bar{\gamma}_p = \frac{1}{3.0}$

$\gamma_s = 1.9$

$\gamma_p = 3.0$