Image Processing Lecture-3

Image Enhancement
(Pixel Operations)

Hazırlayan: M. Kemal GÜLLÜ

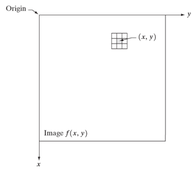
Pixel Operations

- The space consisting of pixels is called as spatial domain.
- Spatial domain operations are generally shown as

$$g(x,y)=T\left[f(x,y)\right]$$

Function/transformation

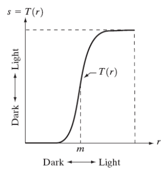
- The function T may take only the pixel located at position (x,y) but also its neighbors.

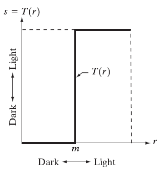


Pixel operations

- If the neighborhood dimension is 1x1 (which means only the pixel at location (x,y) is processed), in this case the T *is called as grayscale-level transformation function.*
- And this kind of operations are called as *point operations*.
- In short, it can be shown as:

$$s=T(r)$$





Brightness adjustment

$$g(x,y)=T\left[f(x,y)\right]$$
$$=f(x,y)+b$$

if b>0 the brightness increases

if b<0 the brightness decreases

$$s=r+b$$


original


b = -50


b = +50

Contrast Adjustment

$$g(x,y)=T\left[f(x,y)\right]$$
$$=af(x,y)$$

if a>1 , the contrast increases

if a<1, the contrast decreases

$$s=ar$$


original

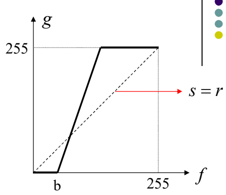

a = 0.5

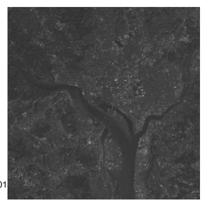

a = 2

Brightness & Contrast Adjustment

$$g(x,y)=T\left[f(x,y)\right]$$
$$=af(x,y)+b$$

$$s=ar+b$$

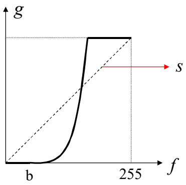




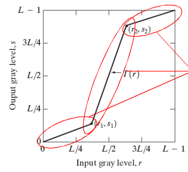
Brightness & Contrast Adjustment

$$g(x,y) = T\left[f(x,y)\right]$$

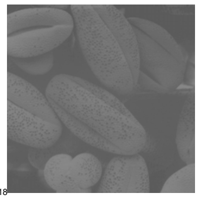
imadjust function in MATLAB

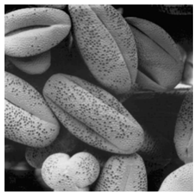


Brightness & Contrast Adjustment

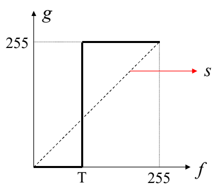


Piecewise linear transform



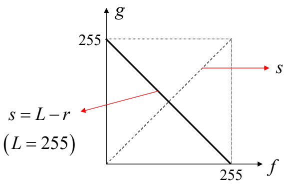


Thresholding



We have a binary image as output.

Negation



Histogram

- Show the frequency (number of occurrence) of each pixel level ([0,255]) for a given image. That is to say, histogram gives distribution of pixel levels in an image.
- Extensively used in image enhancement.

$$h(r_k) = n_k$$

r_k : *kth* grey-level

n_k : number of total pixels at gray-level k

- When normalized, Histogram gives probability occurrence of pixel levels for a given image.

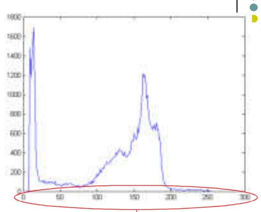
Total number of pixel in the image

Probability of a given gray-level

$$p(r_k) = n_k / n$$

$$k = 0,1,\ldots,L-1$$

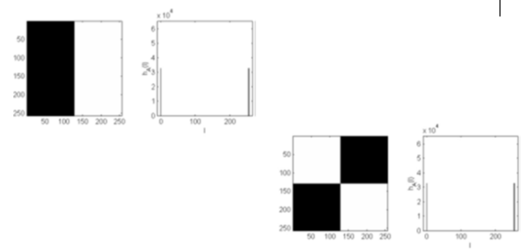
Histogram



Gray-level tones

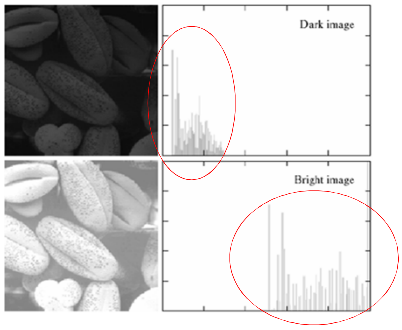
MATLAB: **imhist** function

Histogram

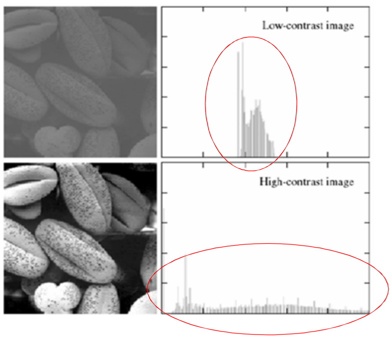


Histogram does not say anything about the spatial features!

Histogram



Histogram



Continuous amplitude random variables

- Let χ be a continuous amplitude random variable $\chi \in (-\infty, +\infty)$.

  $f_\chi(x)$: the probability density function of χ.
  $F_\chi(x)$: the probability distribution function of χ.

$$f_\chi(x)dx = \text{Probability}(x \leq \chi < x + dx)$$
$$F_\chi(x) = \text{Probability}(\chi \leq x)$$

- Properties:

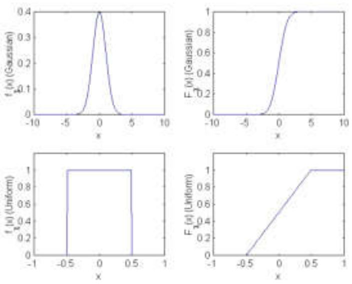
$$F_\chi(x) = \int_{-\infty}^{x} f_\chi(t)dt \Rightarrow \frac{dF_\chi(x)}{dx} = f_\chi(x)$$

$$f_\chi(x) \geq 0 \Rightarrow F_\chi(x) \geq 0, \ F_\chi(x + dx) - F_\chi(x) \geq 0$$

$F_\chi(x)$ is a non-decreasing function.

$$\int_{-\infty}^{+\infty} f_\chi(t)dt = 1 \Rightarrow f_\chi(x)|_{x=+/-\infty} = 0$$
$$F_\chi(x)|_{x=+\infty} = 1$$
$$F_\chi(x)|_{x=-\infty} = 0$$

Continuous amplitude random variables



Gaussian:
$f_\chi(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}}$

Uniform $(a < b)$:
$f_\chi(x) = \begin{cases} \frac{1}{b-a} & a < x < b \\ 0 & \text{otherwise} \end{cases}$

Mean and variance

- Mean (μ):

$$\mu = \int_{-\infty}^{+\infty} x f_\chi(x)dx$$

Analogy: Average price of apples

  - "I bought $f_\chi(x)dx$ many apples at a price of x, ..."
  - "Total price I paid: $P = \int_{-\infty}^{+\infty} x f_\chi(x)dx$.
  - "Total number of apples I purchased: $N = \int_{-\infty}^{+\infty} f_\chi(x)dx = 1$.
  - "My average price for the overall purchase: $\mu = P/N$.

- Variance (σ^2):

$$\sigma^2 = \int_{-\infty}^{+\infty} (x - \mu)^2 f_\chi(x)dx$$

Discrete amplitude random variables

- Let Θ be a discrete amplitude random variable.
  $\Theta = x_i$ for some i, $\ldots, -1, 0, 1, \ldots$.
  x_i are a sequence of possible values for Θ.

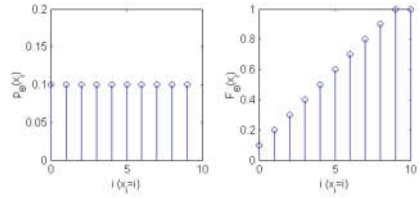
  $p_\Theta(x_i)$: the probability mass function of Θ,
  $F_\Theta(x_i)$: the probability distribution function of Θ.

$$\begin{aligned} p_\Theta(x_i) &= \text{Probability}(\Theta = x_i) \\ F_\Theta(x_i) &= \text{Probability}(\Theta \leq x_i) \end{aligned}$$

- Properties:

$$\begin{aligned} F_\Theta(x_i) &= \sum_{j=-\infty}^{j=i} p_\Theta(x_j) \\ p_\Theta(x_i) &= F_\Theta(x_i) - F_\Theta(x_{i-1}) \geq 0 \\ \sum_{j=-\infty}^{j=+\infty} p_\Theta(x_j) &= 1 \end{aligned}$$

Discrete amplitude random variables



The probability mass and distribution functions for a uniform, discrete amplitude random variable.

Histogram as a probability density function

- For a given image $\mathbf{A}$, consider the image pixels as the realizations of a discrete amplitude random variable "A".
  - For example suppose we toss a coin (Heads=255 and Tails=0) $N \times M$ times and record the results as an N by M image matrix.
- Define the sample probability mass function $p_A(l)$ as the probability of a randomly chosen pixel having the value l.

$$p_A(l) = \frac{h_A(l)}{NM}$$

- Note that the sample mean and variance we talked about in Lecture 2 can be calculated as:

$$\begin{aligned} m_A &= \sum_{l=0}^{255} l p_A(l) \\ \sigma_A^2 &= \sum_{l=0}^{255} (l - m_A)^2 p_A(l) \end{aligned}$$

Histogram Equalization

- This method usually increases the global contrast of many images, especially when the usable data of the image is represented by close contrast values.
- Through this adjustment, the intensities can be better distributed on the histogram. This allows for areas of lower local contrast to gain a higher contrast.
- Histogram equalization accomplishes this by effectively spreading out the most frequent intensity values.

$$cdf(v) = round\left(\frac{cdf(v) - cdf_{\min}}{(M \times N) - cdf_{\min}} \times (L-1)\right)$$

Histogram Equalization

$$\begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix}$$

blok

Value	Count	Value	Count	Value	Count	Value	Count	Value	Count
52	1	64	2	72	1	85	2	113	1
55	3	65	3	73	2	87	1	122	1
58	2	66	2	75	1	88	1	126	1
59	3	67	1	76	1	90	1	144	1
60	1	68	5	77	1	94	1	154	1
61	4	69	3	78	1	104	2		
62	1	70	4	79	2	106	1		
63	2	71	2	83	1	109	1		

histogram

Value	cdf	Value	cdf	Value	cdf	Value	cdf	Value	cdf
52	1	64	19	72	40	85	51	113	60
55	4	65	22	73	42	87	52	122	61
58	6	66	24	75	43	88	53	126	62
59	9	67	25	76	44	90	54	144	63
60	10	68	30	77	45	94	55	154	64
61	14	69	33	78	46	104	57		
62	15	70	37	79	48	106	58		
63	17	71	39	83	49	109	59		

cdf

$$cdf(v) = round\left(\frac{cdf(v) - cdf_{\min}}{(M \times N) - cdf_{\min}} \times (L-1)\right)$$

$$cdf(v) = round\left(\frac{cdf(v) - 1}{64 - 1} \times 255\right)$$

$$cdf(78) = round\left(\frac{46 - 1}{63} \times 255\right) = 182$$

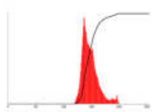
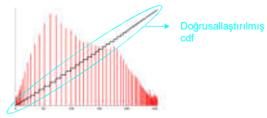
$$cdf(154) = round\left(\frac{64 - 1}{63} \times 255\right) = 255$$

$$\begin{bmatrix} 0 & 12 & 53 & 93 & 146 & 53 & 73 & 166 \\ 65 & 32 & 12 & 215 & 235 & 202 & 130 & 158 \\ 57 & 32 & 117 & 239 & 251 & 227 & 93 & 166 \\ 65 & 20 & 154 & 243 & 255 & 231 & 146 & 130 \\ 97 & 53 & 117 & 227 & 247 & 210 & 117 & 146 \\ 190 & 85 & 36 & 146 & 178 & 117 & 20 & 170 \\ 202 & 154 & 73 & 32 & 12 & 53 & 85 & 194 \\ 206 & 190 & 130 & 117 & 85 & 174 & 182 & 219 \end{bmatrix}$$

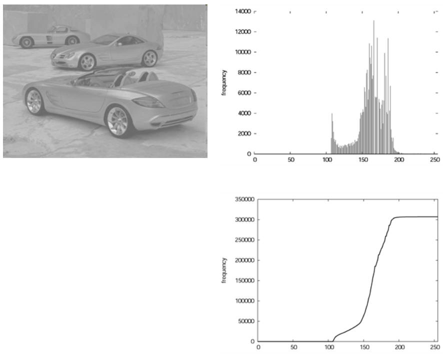
Histogram Equalization

- It makes the probability distribution function linear.

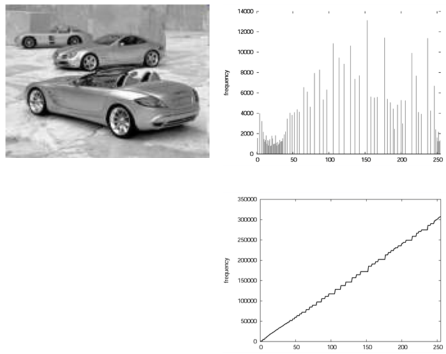
 

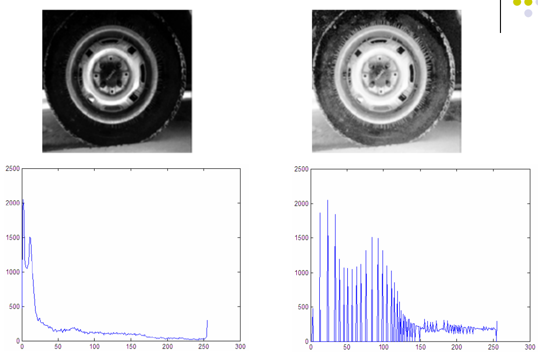
Histogram Equalization



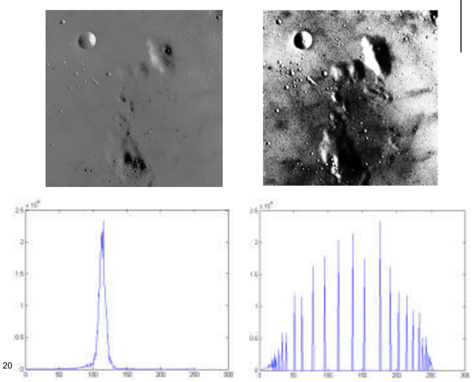
Histogram Equalization



Histogram Equalization



Histogram Equalization



Quantization

- Let $t_n \in \{0, 1, \ldots, 255\}$ denote a sequence of thresholds $(n = 0, \ldots, P-1)$.
- Consider the P "half-open, discrete intervals" $R_n = [t_n, t_{n+1})$ $(t_0 = 0,\ t_P = 256)$.
- Let $r_n \in R_n$ be the reproduction level of the interval R_n.
- Define the quantizing point function or the P-level quantizer $Q(l)$ in terms of the $R_n,\ r_n$ (or equivalently in terms of $t_n,\ r_n$) as follows:

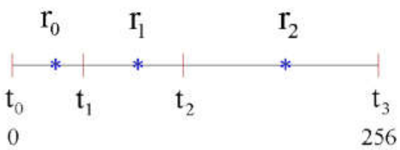
$$Q(l) = \{r_k | l \in R_k,\ k = 0, \ldots, P-1\}$$

i.e., $l \in R_k \Leftrightarrow Q(l) = r_k$.

- Quantizing an image $\mathbf{A}$ in matlab:

```
>> Q   = zeros(256,1);  x = (0:255)';
>> for i = 1:P
       Q = Q + r(i)*((x >= t(i))&(x < t(i+1)));
       end;      % t(P+1) = 256
>> B   = Q(A+1);
```

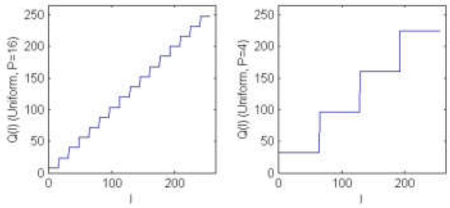

Quantization



- $R_n = [t_n, t_{n+1})$.

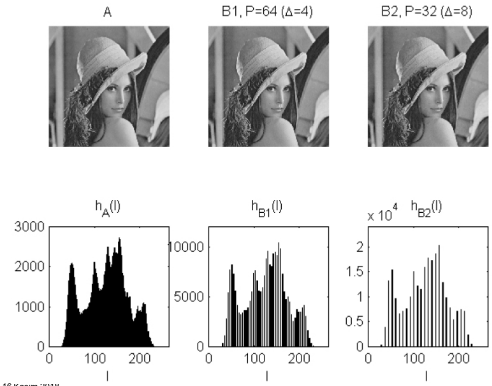
Quantization

- In uniform quantization $P = 256/\Delta$, $t_{n+1} - t_n = \Delta$, $\forall n$ and $r_n = \frac{t_n + t_{n+1}}{2}$.
- Δ is the stepsize of the uniform quantizer $(r_n = n\Delta + \Delta/2)$.

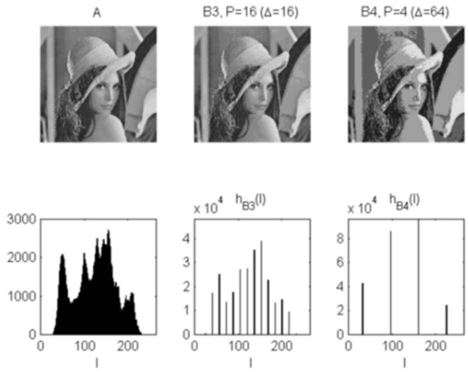


Easy uniform quantization: >> $B = delta * \text{floor}(A/delta) + delta/2;$

Quantization



Quantization



Quantization

- The quantization error matrix is defined as $\mathbf{E} = \mathbf{A} - Q(\mathbf{A})$.
- The sample mean squared quantization error (MSQE) is:

$$\begin{aligned} \text{MSQE} &= \frac{\Sigma_{i=0}^{N-1} \Sigma_{j=0}^{M-1} (E(i,j))^2}{NM} \\ &= \frac{\Sigma_{i=0}^{N-1} \Sigma_{j=0}^{M-1} (A(i,j) - Q(A(i,j)))^2}{NM} \\ &= \sum_{l=0}^{255} (l - Q(l))^2 p_A(l) \end{aligned}$$

Example

For the earlier example:

Δ	Quantized Image	MSQE
4	B1	1.50
8	B2	5.49
16	B3	22.18
64	B4	334.77