

# Point Operation

## Histogram Modification

김성영교수  
금오공과대학교  
컴퓨터공학과

# 학습 내용

---

- HISTOGRAM
- HISTOGRAM MODIFICATION

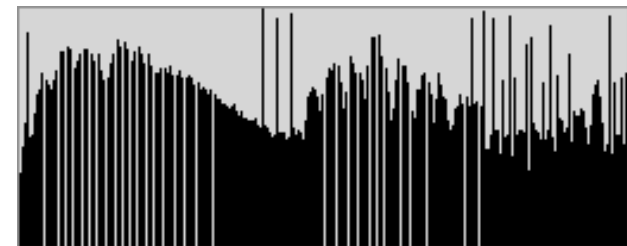
# HISTOGRAM

- A simple datum that gives **the number of pixels that a given value** in an image
- Ex) a 8bit gray-scale image

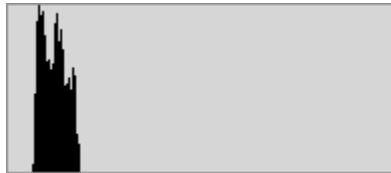
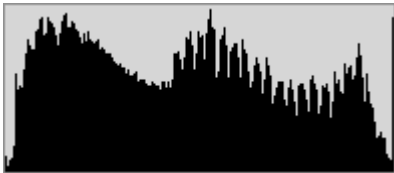


Bin	Counts	Prob.
0	163	0.005
1	77	0.003
...		
255	1561	0.051

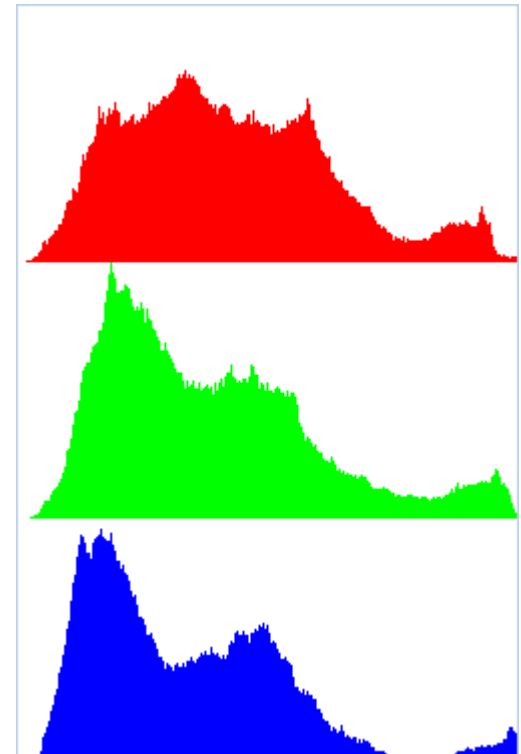
number  
of  
pixels



gray level

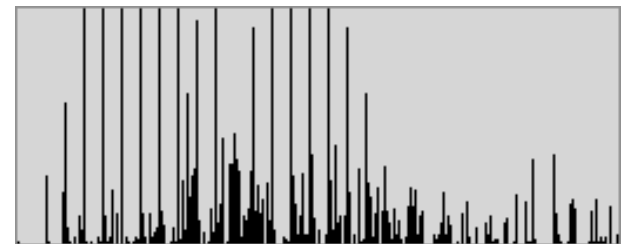


in color images





Color Quantization



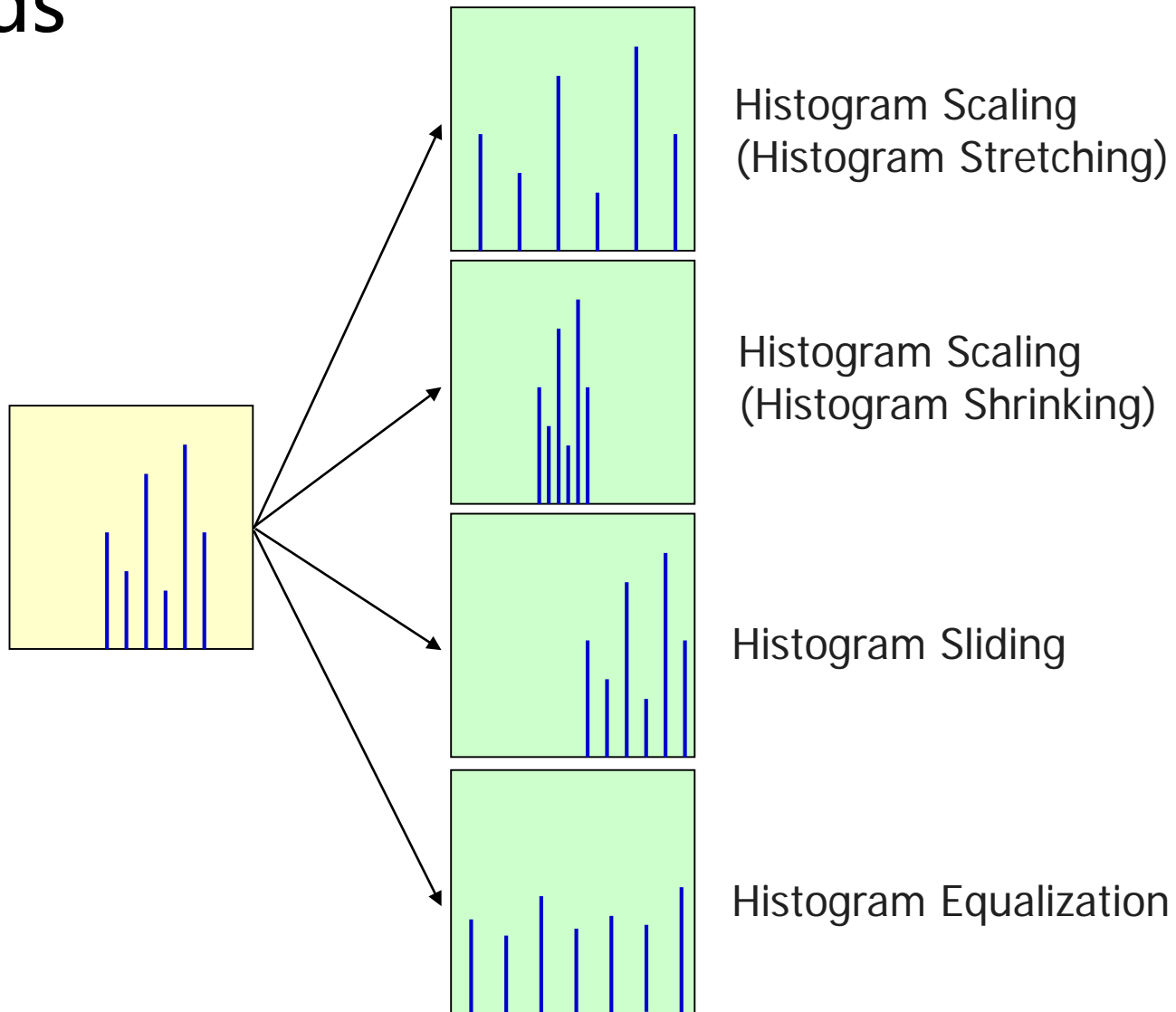
# HISTOGRAM MODIFICATIONS

---

Improving image contrast and brightness based on **histogram**

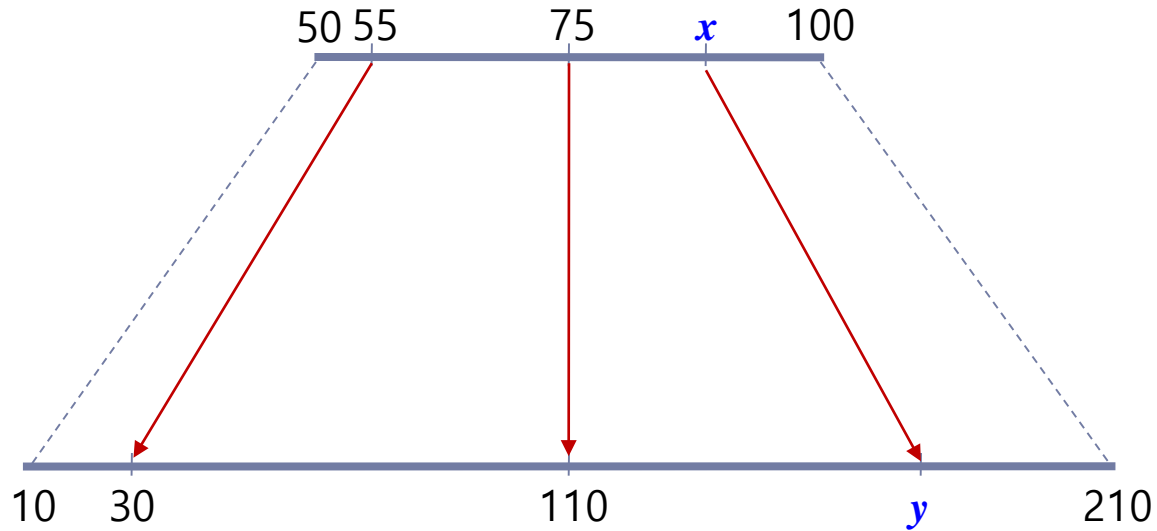
➡ Focus on the histogram **shape** and **range**

# fields





# Scaling



$$(100 - 50) : (x - 50) = (210 - 10) : (y - 10)$$

$$(y - 10) * (100 - 50) = (x - 50) * (210 - 10)$$

$$y = \frac{(x - 50) * (210 - 10)}{(100 - 50)} + 10 = \frac{(210 - 10)}{(100 - 50)} (x - 50) + 10$$

$$I'(x, y) = \frac{(S_{\max} - S_{\min})}{(I_{\max} - I_{\min})} (I(x, y) - I_{\min}) + S_{\min}$$

$$\mathbf{O}(x, y) = \left[ \frac{S_{max} - S_{min}}{I_{max} - I_{min}} \right] [\mathbf{I}(x, y) - I_{min}] + S_{min}$$

$I_{max}$  : largest gray-level value in the image  $\mathbf{I}(x, y)$

$I_{min}$  : smallest gray-level value in  $\mathbf{I}(x, y)$

$S_{max}$  : maximum gray-level values possible

$S_{min}$  : minimum gray-level values possible

Low-contrast  
image



Histogram of  
low-contrast  
image

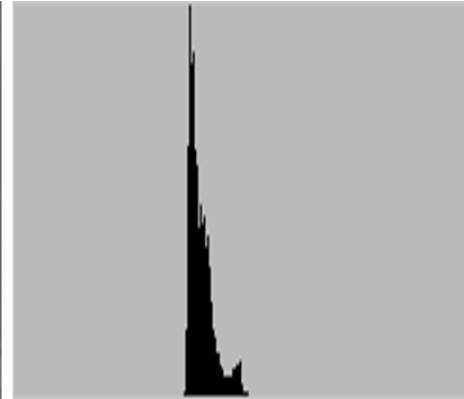
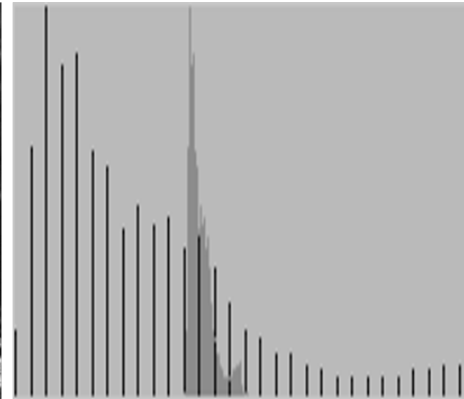
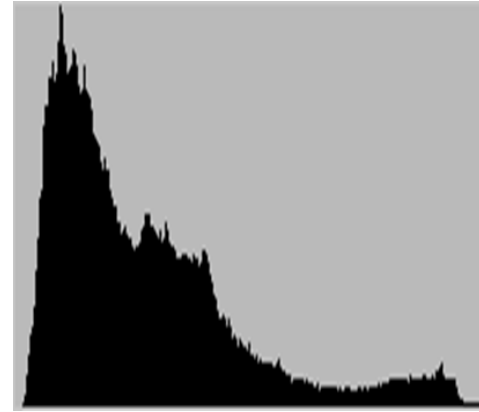


Image after  
histogram  
stretching



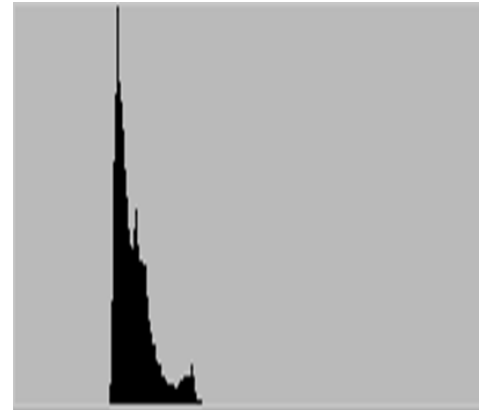
Histogram of  
image after  
stretching





Histogram  
of original  
image

Image after  
shrinking



Histogram  
of shrunk  
image

# Sliding

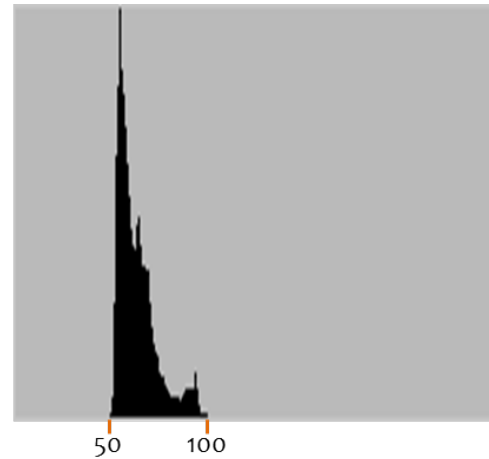
$$\mathbf{O}(x, y) = \mathbf{I}(x, y) + \textit{offset}$$

*offset* : amount to slide the histogram

Original  
image



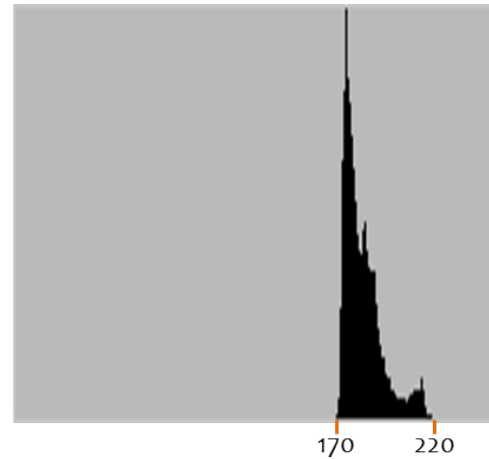
Histogram of  
original image



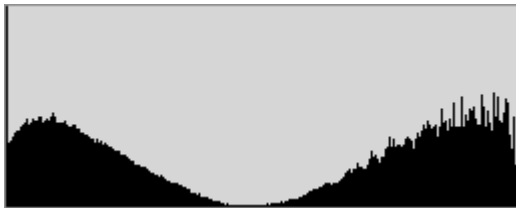
positive-  
value  
histogram  
sliding



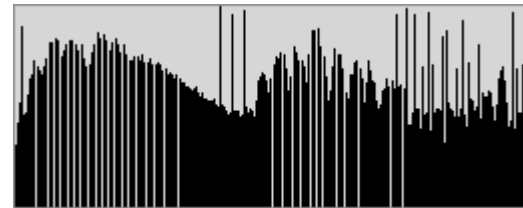
Histogram of  
image after  
sliding



# Equalization

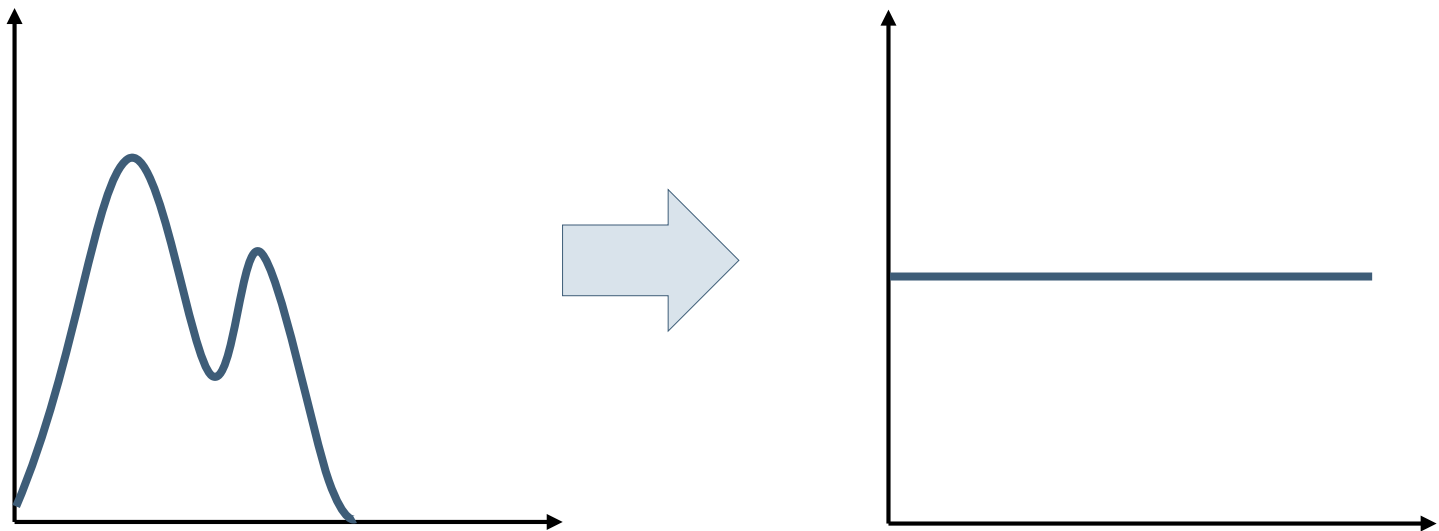


높은 contrast



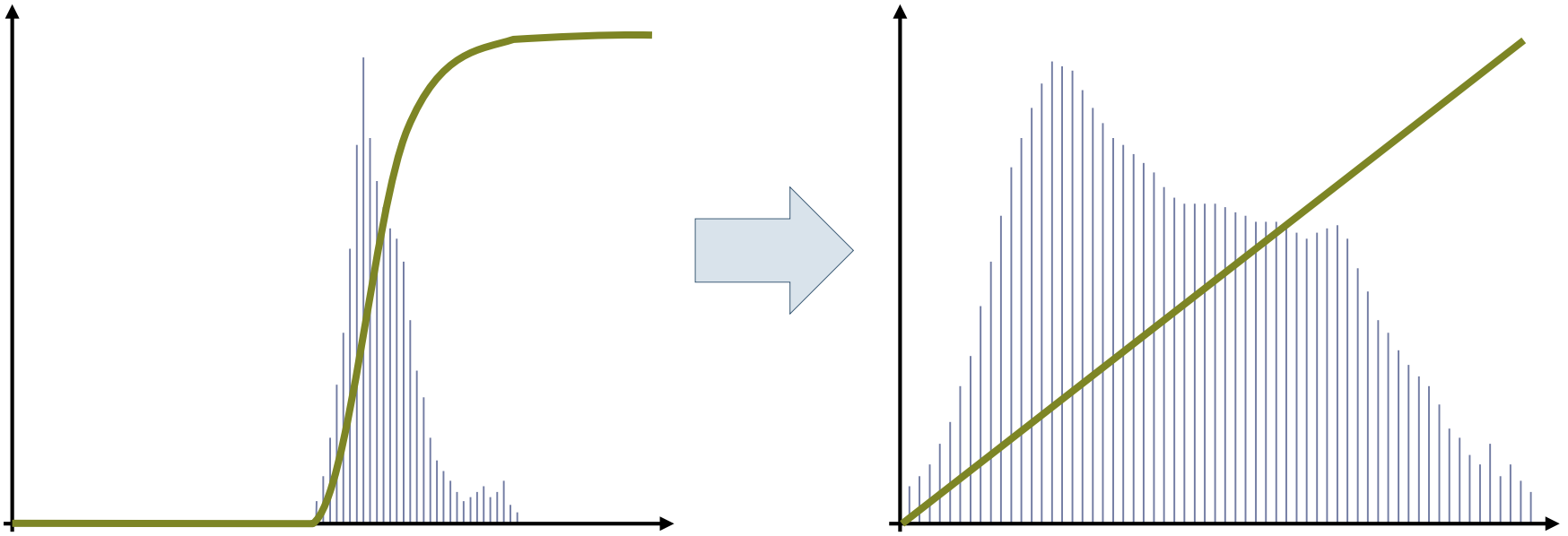
좋은 contrast

Aim to change a picture in such a way as to produce a picture with *flatter* histogram, where all levels are equiprobable





CDF  
Cumulative Distribution Function



# Deriving Algorithm (1)

$\mathbf{I}(l)$  and  $\mathbf{O}(l)$  ( $0 \leq l < L$ ):

for the input and output image, the number of pixels per level

$$\sum_{l=0}^{L-1} \mathbf{I}(l) = \sum_{l=0}^{L-1} \mathbf{O}(l)$$

for an arbitrarily chosen level  $p$  in the input image

$$\sum_{l=0}^p \mathbf{I}(l) = \sum_{l=0}^q \mathbf{O}(l)$$

## Deriving Algorithm (2)

Since the output histogram is uniformly flat  
( $T$ : total number of pixels in the image)

$$\mathbf{o}(l) = \frac{T}{N_{\max} - N_{\min}}$$

So the cumulative histogram of the output image

$$\sum_{l=0}^q \mathbf{o}(l) = q \times \frac{T}{N_{\max} - N_{\min}} = \sum_{l=0}^p \mathbf{i}(l)$$

## Deriving Algorithm (3)

Output pixels at level  $q$  is given by

$$E(q, \mathbf{I}) = q = \frac{N_{\max} - N_{\min}}{T} \times \sum_{l=0}^p \mathbf{I}(l)$$

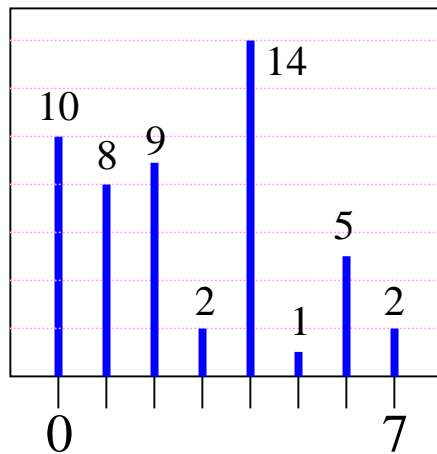
Equalizing function ( $E$ ) of the level ( $q$ ) and the image ( $I$ )

The output image is then

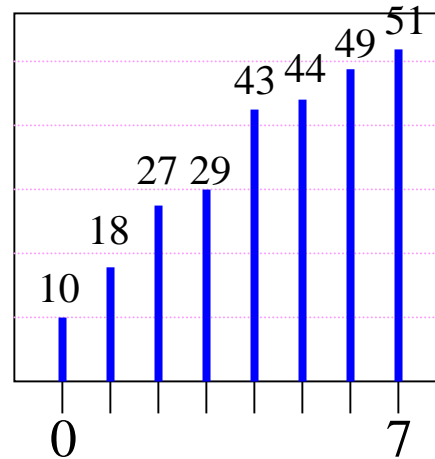
$$\mathbf{O}_{x,y} = E(\mathbf{I}_{x,y}, \mathbf{I})$$

# algorithm

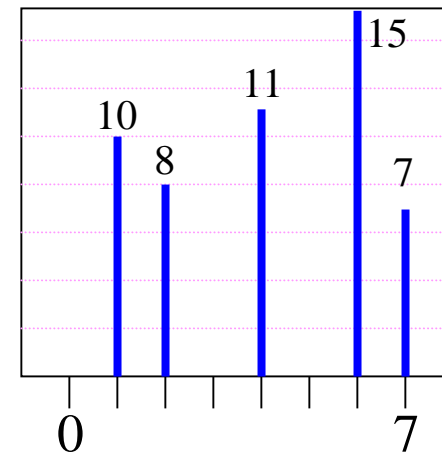
- ① 입력 영상의 히스토그램의 값을 누적시켜 각 레벨에서의 히스토그램 누적 합 계산
- ② 히스토그램의 누적 합을 전체 픽셀의 개수로 나누어 값을 정규화함
- ③ 정규화된 값에 최대 gray level 값을 곱한 후 반올림을 수행
- ④ 입력 영상의 각 gray level에 대해 변환 값으로 대응



히스토그램



누적값



균일화 결과

$$\frac{(10, 18, 27, 29, 43, 44, 49, 51)}{51} \times 7$$

$$\approx (1.37, 2.47, 3.71, 3.98, 5.90, 6.04, 6.73, 7.00)$$

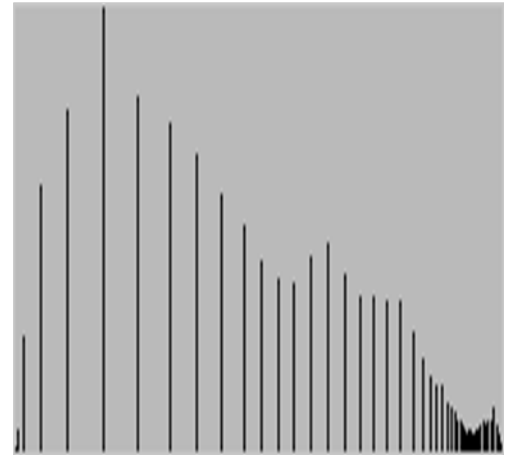
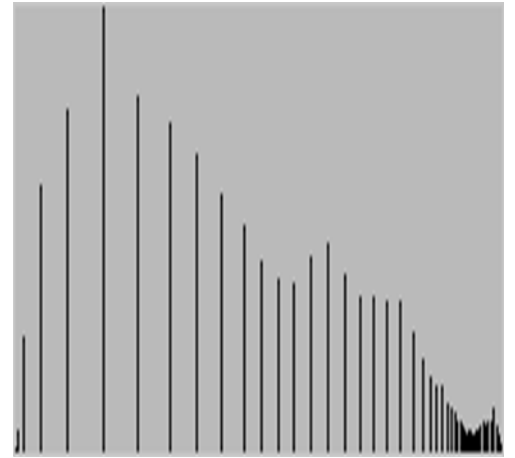
$$\approx (1, 2, 4, 4, 6, 6, 7, 7)$$

# LUT

4	5	3	6	7
4	2	2	4	6
0	2	2	5	7
0	0	2	3	5
0	1	2	4	4

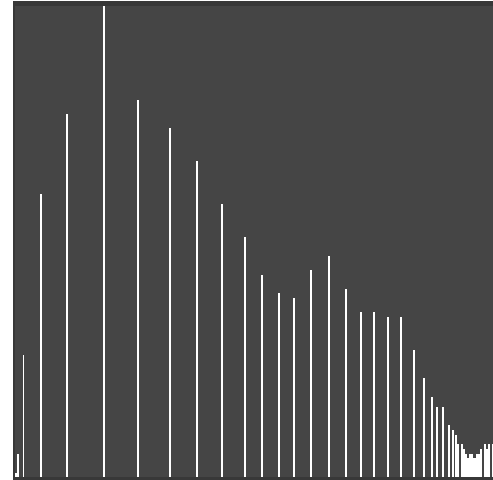
입력레벨	개수	누적값	균일화	결과레벨
0	10	10	1.37	1
1	8	18	2.47	2
2	9	27	3.71	4
3	2	29	3.98	4
4	14	43	5.90	6
5	1	44	6.04	6
6	5	49	6.73	7
7	2	51	7.00	7

6	6	4	7	7
6	4	4	6	7
1	4	4	6	7
1	1	4	4	6
1	2	4	6	6

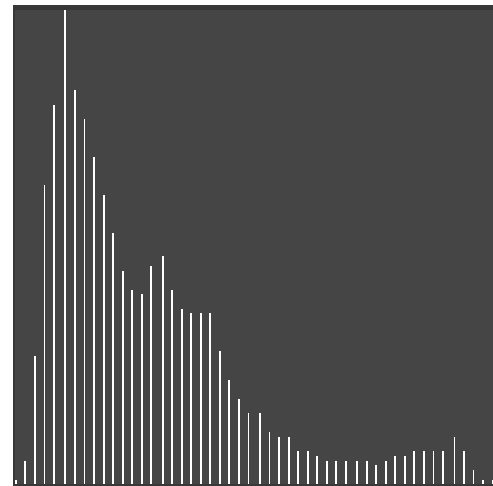




# Histogram normalization vs. equalization



equalization



normalization

# DISADVANTAGE

---

- Background noise can be increased
- The image quality in a near-constant region may be degraded

# REFERENCE

---

- R. Gonzalez, R. Woods, **Digital Image Processing (2nd Edition)**, Prentice Hall, 2002
- Scott E Umbaugh, **Computer Imaging**, CRC Press, 2005
- Mark Nixon and Alberto Aguado, **Feature Extraction & Image Processing**, ELSEVIER, 2008
- Frank SHIH, **Image Processing and Pattern Recognition**, IEEE Press, 2010