EECS 332 Digital Image Analysis

Histogram Techniques

Ying Wu

Dept. Electrical Engineering and Computer Science
Northwestern University
Evanston, IL 60208

http://www.ece.northwestern.edu/~yingwu
yingwu@ece.northwestern.edu

What we've learnt …

- Binary image analysis
  - Attributes of a region (area, centroid, orientation)
  - CCL
  - Morphological operators
  - Boundary

Questions

- How do we get those binary images?

Image segmentation

gray-level images

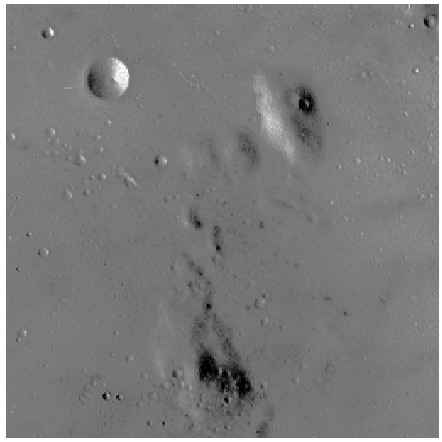
color images

- We'll introduce some basic concepts today.

Outline

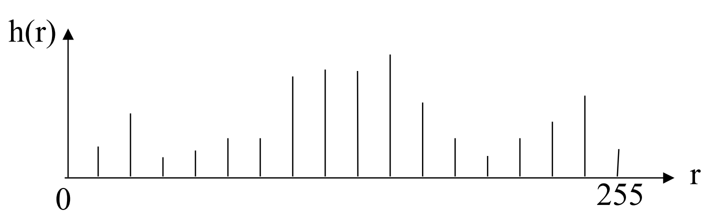
- Histogram
- Histogram equalization
- Fitting
- Lighting correction

Let's view this picture



- Why doesn't this picture look good?
- Can we make it look good? If so, how?

Histogram



- Histogram is a discrete representation of the distribution on the quantized pixel attributes (e.g., pixel intensities).
- Histogram bin
  - # of bins is the levels of quantized attributes
  - The value of each bin is the frequency that the corresponding attribute appears in the image
- E.g. $h(r_k) = n_k$, where
  - r_k is the k-th gray scale of the intensity
  - n_k is the # of pixels with intensity r_k.

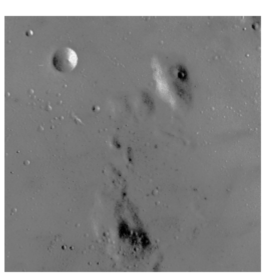
How to construct a histogram

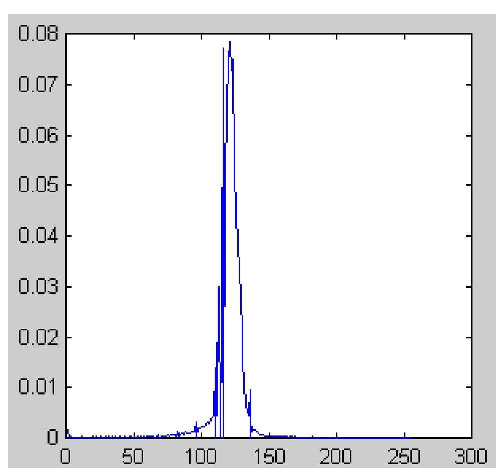
- As easy as ABC …

```
for r = 1:R
      for c = 1:C
            h(I(r,c)) ++ ;
      end
end
```

Note: This is just conceptual code. You need to consider bin quantization before writing the actual code.

Aha! I know the reason …



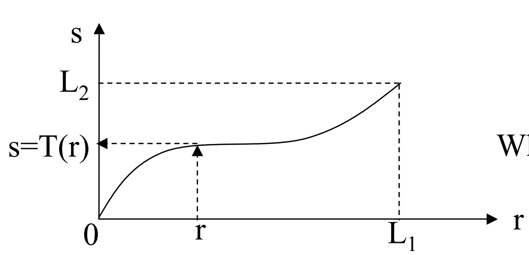


Idea?

- We'd better make the histogram more spread and more even.
- Why?
  - Human vision system can only distinguish about 20 grayscales
- In this particular case, we need to stretch the histogram.
- But, how … ?

Histogram Equalization

- Problem statement:
  - Input image gray level $r \in [0, L_1]$
  - Output image gray level $s \in [0, L_2]$
  - We need to find a transformation $s = T(r)$
    s.t., s tends to be distributed uniformly
  - Requirement: T(r) monotonically non-decreasing



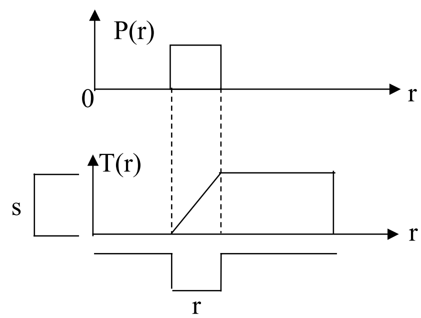
What is the best T(r)?

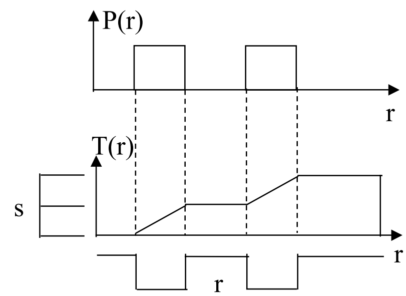
The Solution is simple

$$T(r_k) = \sum_{j=0}^{k} P(r_j) \cdot L_2$$

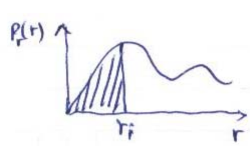
$P(r_j)$: The mass distribution of r_j

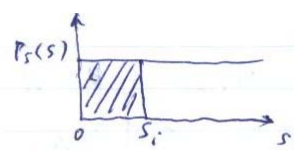
Cumulative mass distribution of r_j (or just the normalized cumulative histogram!)





Why does it work?





we want $\int_0^{r_i} P_r(t)dt = \int_0^{s_i} P_s(t)dt$.

$= \frac{MN}{S_{max}} s_i$ MxN # of pixels

So: $s_i = \frac{S_{max}}{MN} \int_0^{r_i} P_r(t)dt$

$\doteq \frac{S_{max}}{MN} \sum_{i=0}^{r_i} H(i)$

A formal explanation

gray levels in an image may be viewed as r.v. in the interval [0,1]. $r \sim p_r(r)$ where $s = T(r)$

$s \sim p_s(s)$

probability theory: $r \xrightarrow{T(r)} s$

$$p_s(s) = p_r(r)\left|\frac{dr}{ds}\right| = \frac{p_r(r)}{|J|} \quad \text{where } J = \frac{ds}{dr} = \frac{dT(r)}{dr}$$

we choose

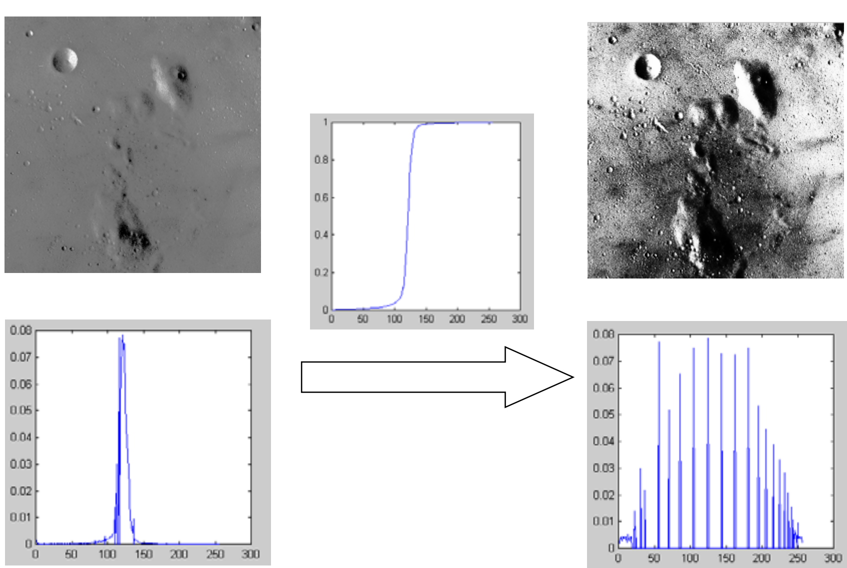
$$s = T(r) = \int_0^r p_r(w)\,dw$$

so $\dfrac{ds}{dr} = \dfrac{dT(r)}{dr} = p_r(r)$

$$\Rightarrow p_s(s) = p_r(r)\cdot\left|\frac{1}{p_r(r)}\right| = 1 \qquad 0 \le s \le 1.$$

If you have not taken ECE302, skip this.

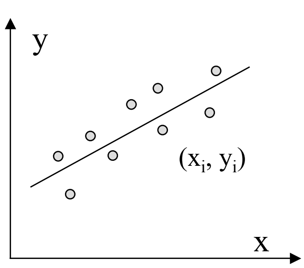
It works!



But not perfect

- The lighting is uneven. Some parts are brighter than other parts.
- The shadow also affects the imaging.
- Idea?
  - Seems the lighting is from the upper-right corner
  - Can we correct the lighting by tilting the image?
  - Then, we need to find a tilted 3D plan before compensation.

Line fitting



to find a line $y = ax + b$

s.t., $\sum_{i=1}^{N} (ax_i + b - y_i)^2$ is minmized

Let's derive on-line!

Note: details can be found in class notes.

Generalized LS fitting

$$\begin{cases} ax_1 + b = y_1 \\ ax_2 + b = y_2 \\ \vdots \\ ax_N + b = y_N \end{cases} \Longleftrightarrow \begin{bmatrix} x_1 & 1 \\ x_2 & 2 \\ \vdots & \vdots \\ x_N & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$

$$\Longleftrightarrow \quad \mathrm{A} = \begin{bmatrix} x_1 & 1 \\ x_2 & 2 \\ \vdots & \vdots \\ x_N & 1 \end{bmatrix}; \quad \mathrm{x} = \begin{bmatrix} a \\ b \end{bmatrix}; \quad \mathrm{t} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix}$$

An over-determined linear system.We need to find the least squares solution.

$$\mathrm{Ax} = \mathrm{t}$$

$$\Rightarrow \quad \mathrm{x} = (\mathrm{A}^T\mathrm{A})^{-1}\mathrm{A}^T\mathrm{t}$$

Least squares solution

$$\mathrm{A}^{\dagger} = (\mathrm{A}^T\mathrm{A})^{-1}\mathrm{A}^T$$

$$\mathrm{x} = \mathrm{A}^{\dagger}\mathrm{t}$$

Pseudo-inverse

Note: is the the same as the last slide?

Plane fitting

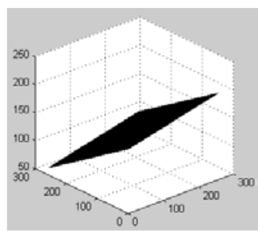
$$\begin{cases} a_1u_1 + a_2v_1 + a_3 = I(u_1, v_1) \\ a_1u_2 + a_2v_2 + a_3 = I(u_2, v_2) \\ \vdots \\ a_1u_N + a_2v_N + a_3 = I(u_N, v_N) \end{cases}$$

$$\begin{bmatrix} u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \\ \vdots & \vdots & \vdots \\ u_N & v_N & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} I(u_1, v_1) \\ I(u_2, v_2) \\ \vdots \\ I(u_N, v_N) \end{bmatrix}$$

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \\ \vdots & \vdots & \vdots \\ u_N & v_N & 1 \end{bmatrix}^{\dagger} \begin{bmatrix} I(u_1, v_1) \\ I(u_2, v_2) \\ \vdots \\ I(u_N, v_N) \end{bmatrix}$$

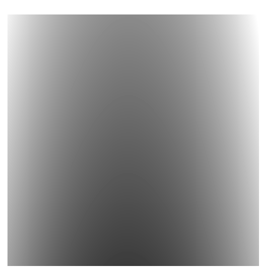
Lighting correction – linear

Truncated version

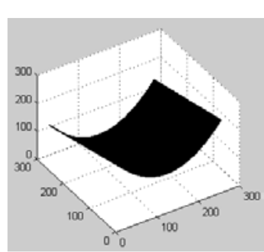


Scaled version

Lighting correction – quadratic



Truncated version



Scaled version