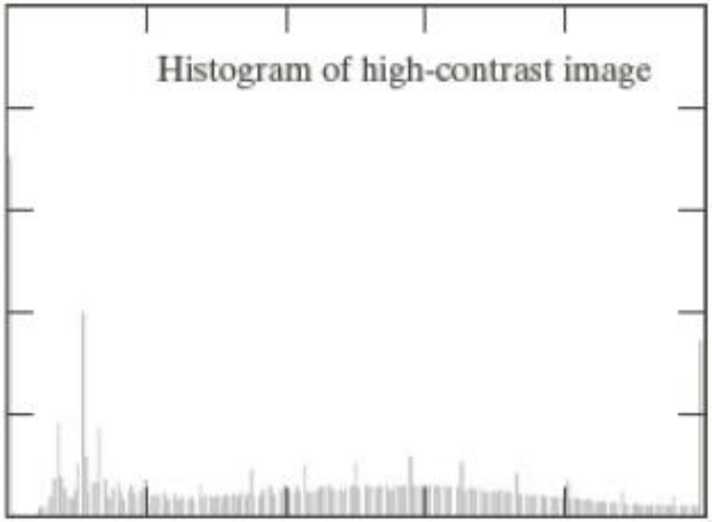
Histogram based trasformations

- The histogram provides an intuitive (visual) tool for evaluating some statistical properties of the image.
- Histogram based transformations are numerous:
  - enhancement,
  - compression,
  - segmentation;
- and can be easily implemented:
  - cheap;
  - dedicated hardware.

High contrast image



- The histogram components are distributed over all the intensity range.
- The distribution is almost uniform, with few peaks.
- If the distribution is uniform, the image tends to have a high dynamic range and the details are more easily perceived.
- This is the effect pursued by the histogram based transformations.

Automatic Contrast Adjustment

- Point operation that modifies pixel intensities such that available range of values is fully covered
- Algorithm:
  - Find high and lowest pixel intensities a_{low}, a_{high}
  - Linear stretching of intensity range

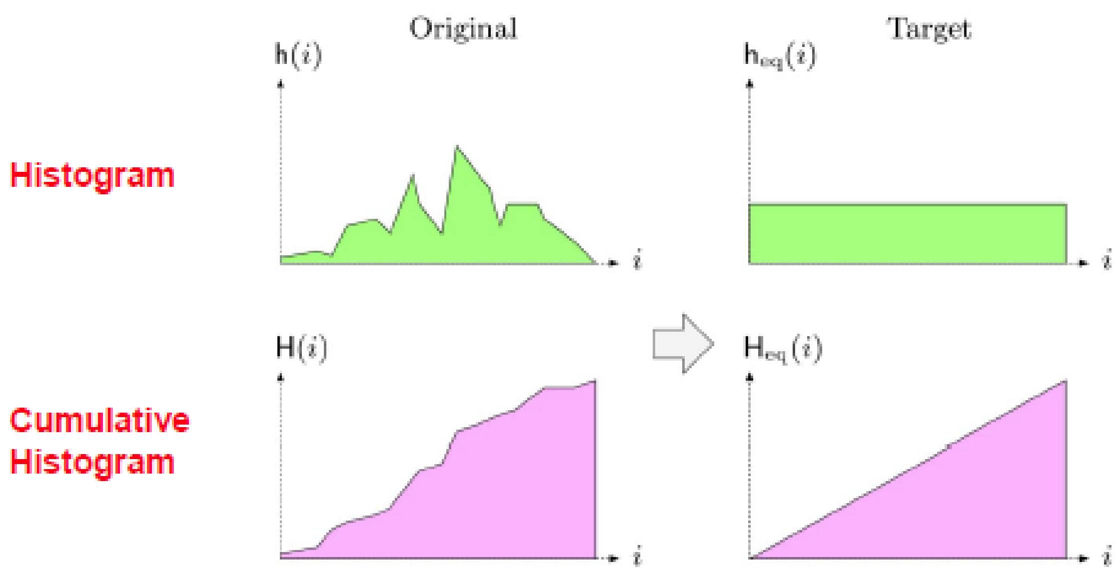
$$f_{\text{ac}}(a) = a_{\text{min}} + (a - a_{\text{low}}) \cdot \frac{a_{\text{max}} - a_{\text{min}}}{a_{\text{high}} - a_{\text{low}}}$$

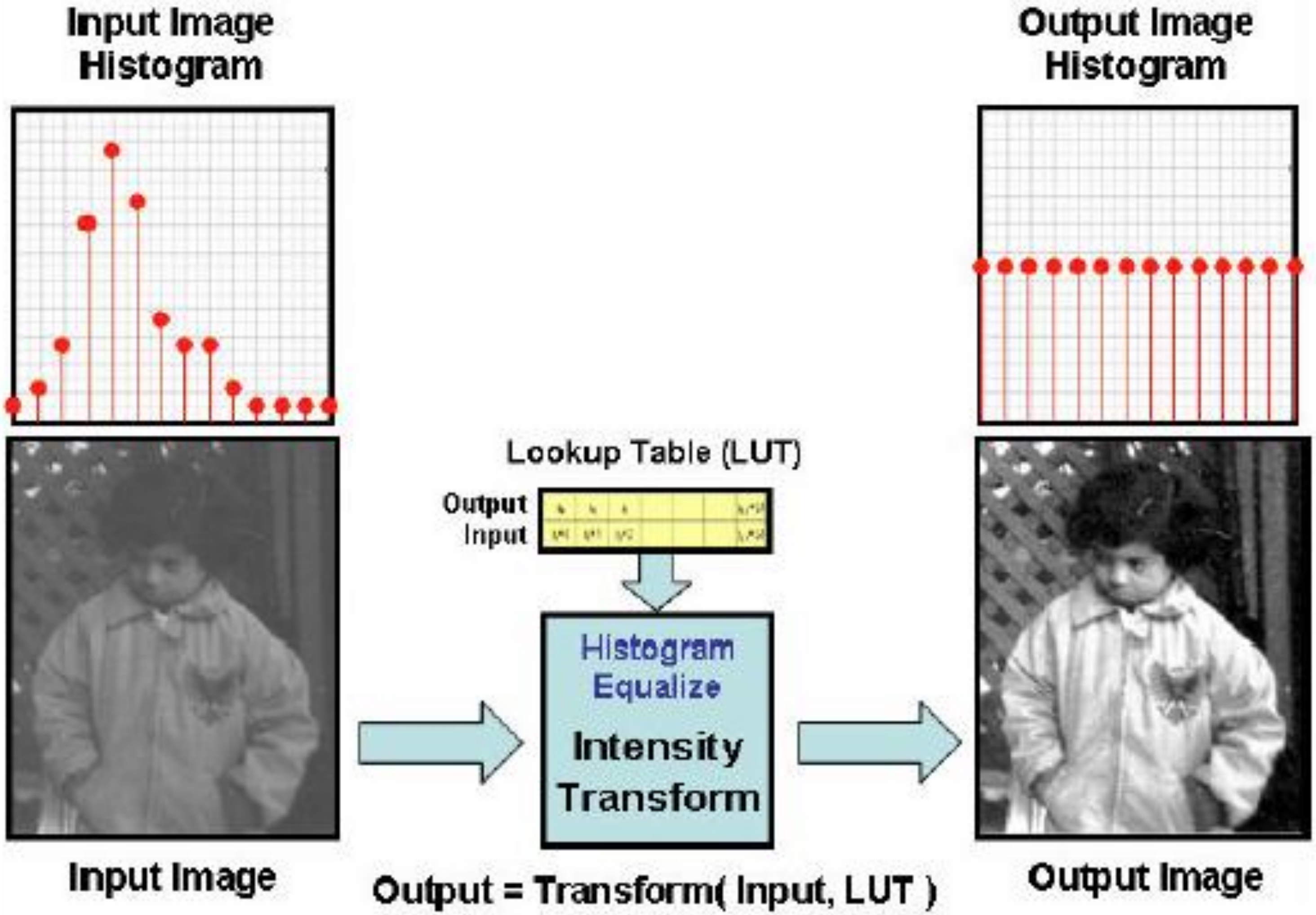
If $a_{min} = 0$ and $a_{max} = 255$

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

Histogram Equalization

- Adjust 2 different images to make their histograms (intensity distributions) similar
- Apply a point operation that changes histogram of modified image into **uniform distribution**





Input Image Histogram
Output Image Histogram
Lookup Table (LUT)
Output
Input
Histogram Equalize
Intensity Transform
Input Image
Output = Transform(Input, LUT)
Output Image

Histogram Equalization

Spreading out the frequencies in an image (or equalizing the image) is a simple way to improve dark or washed out images

Can be expressed as a transformation of histogram

- r_k: input intensity
- s_k: processed intensity
- k: the intensity range (e.g 0.0 – 1.0)

processed intensity

$$s_k = T(r_k)$$

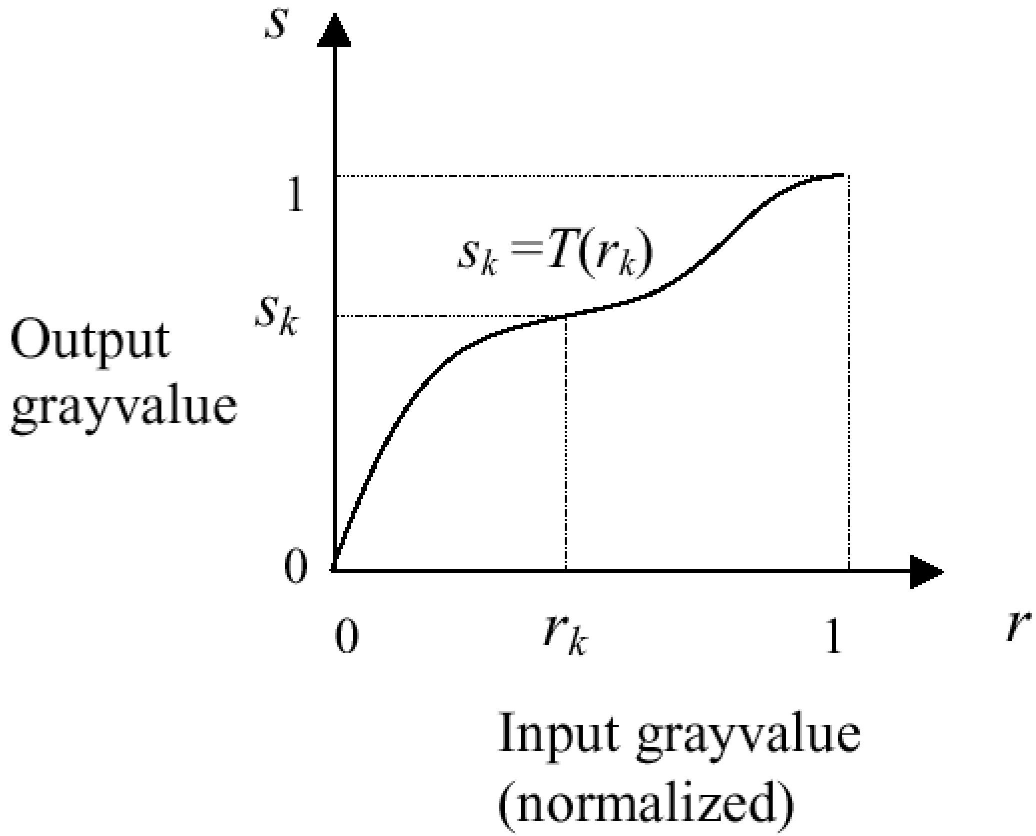
input intensity

Intensity range (e.g 0 – 255)

- As before, we assume that:

(1) *T(r)* is a monotonically increasing function for 0 $\leq r \leq 1$ (preserves order from black to white).

(2) *T(r)* maps [0,1] into [0,1] (preserves the range of allowed Gray values).



Cumulative Histogram

- Useful for certain operations (e.g. histogram equalization) later
- Analogous to the Cumulative Density Function (CDF)
- Definition:

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

- Recursive definition

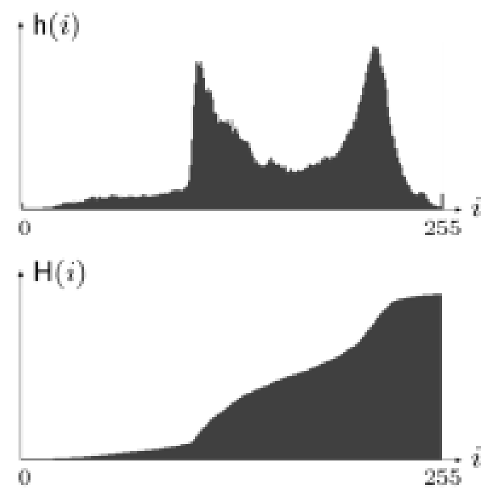
$$\mathsf{H}(i) = \begin{cases} \mathsf{h}(0) & \text{for} \;\; i = 0 \\ \mathsf{H}(i-1) + \mathsf{h}(i) & \text{for} \;\; 0 < i < K \end{cases}$$

- Monotonically increasing

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

Last entry of
Cum. histogram

Total number of
pixels in image



Histogram Equalization

$$s_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n} = \sum_{j=0}^{k} p_r(r_j)$$

- Histogram equalization (HE) results are similar to contrast stretching but offer the advantage of full automation, since HE automatically determines a transformation function to produce a new image with a uniform histogram.

How to implement histogram equalization?

Step 1:For images with discrete gray values, compute:

$$p_{in}(r_k) = \frac{n_k}{n} \qquad 0 \le r_k \le 1 \qquad 0 \le k \le L-1$$

L: Total number of gray levels

n_k: Number of pixels with gray value r_k

n: Total number of pixels in the image

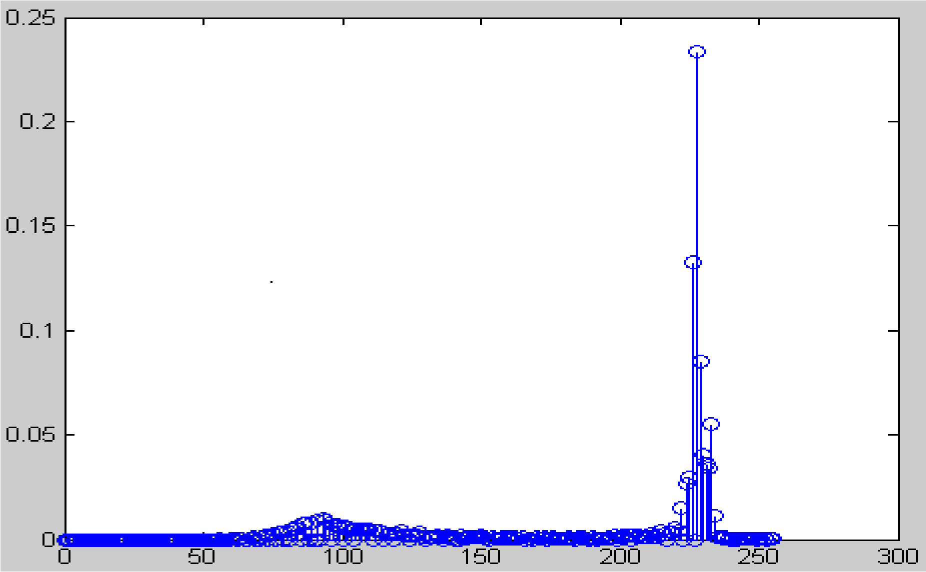
Step 2: Based on CDF, compute the discrete version of the previous transformation :

$$s_k = T(r_k) = \sum_{j=0}^{k} p_{in}(r_j) \qquad 0 \le k \le L-1$$

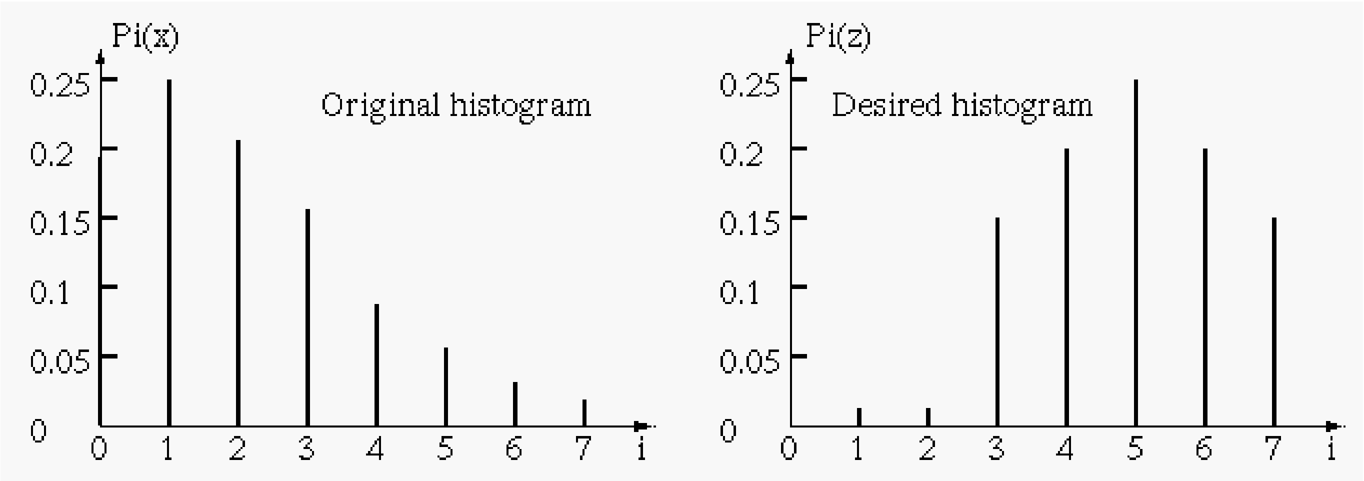
- Applying the transformation, $s_k = T(r_k) = \sum_{j=0}^{k} p_{in}(r_j)$ we have

- Comments:

Histogram equalization may not always produce desirable results, particularly if the given histogram is very narrow. It can produce false edges and regions. It can also increase image “graininess” and “patchiness.”



Histogram matching



x_i	n_j	h_x	$y = H_x$
0	790	0.19	0.19
1	1023	0.25	0.44
2	850	0.21	0.65
3	656	0.16	0.81
4	329	0.08	0.89
5	245	0.06	0.95
6	122	0.03	0.98
7	81	0.02	1.00

z_i	p_z	$\underline{y' = H_z}$
0	0.0	0.0
1	0.0	0.0
2	0.0	0.0
3	0.15	0.15
4	0.20	0.35
5	0.30	0.65
6	0.20	0.85
7	0.15	1.0

$x_i = i$	$y_j = H_x$	$\underline{y'_j = H_z}$	$z_j = j$
0	0.19	0.0	3
1	0.44	0.0	4
2	0.65	0.0	5
3	0.81	0.15	6
4	0.89	0.35	6
5	0.95	0.65	7
6	0.98	0.85	7
7	1.0	1.0	7

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

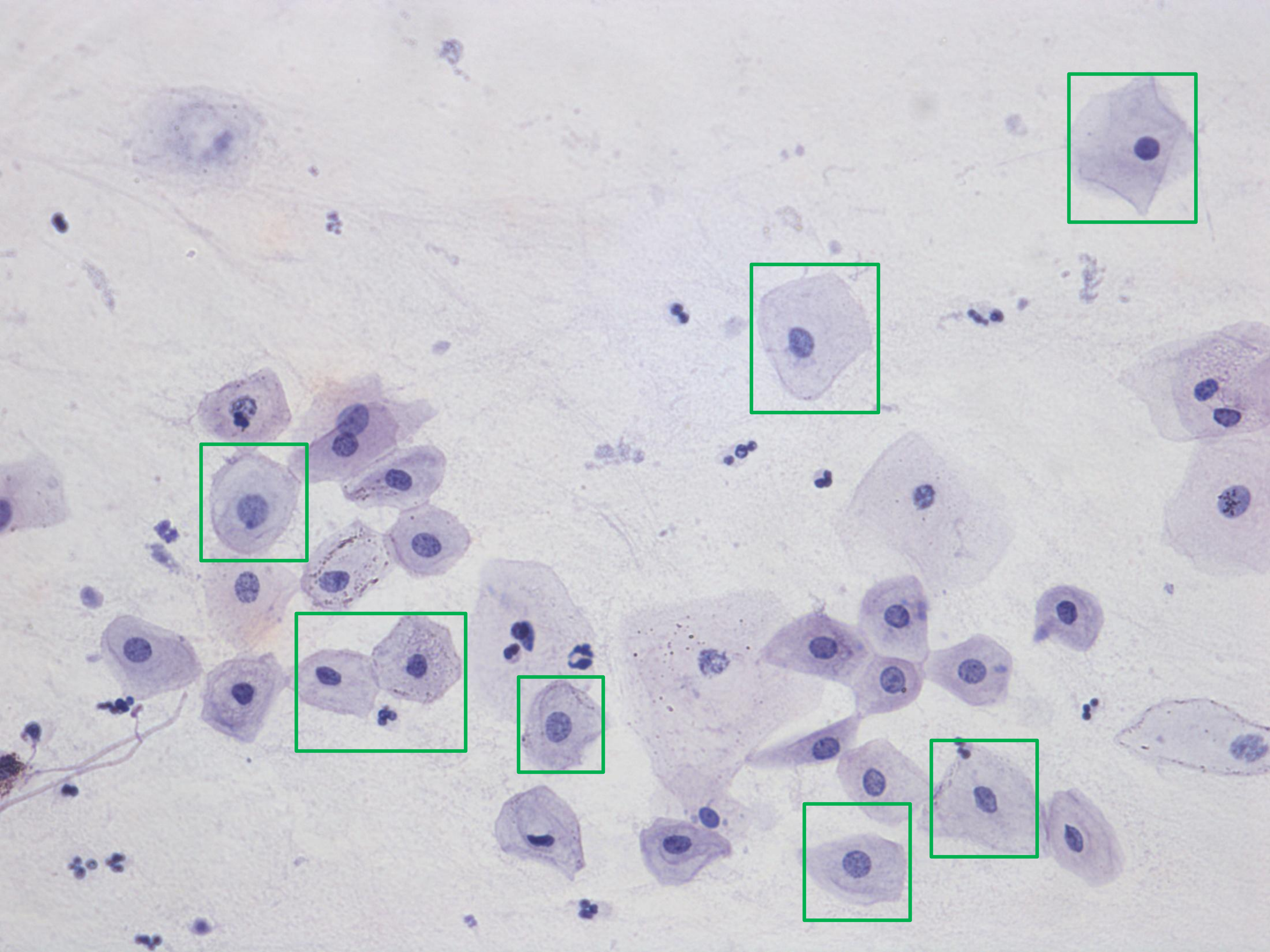
Image Entropy

Entropy is a concept which originally arose from the study of the physics of heat engines. It can be described as a measure of the amount of disorder in a system.

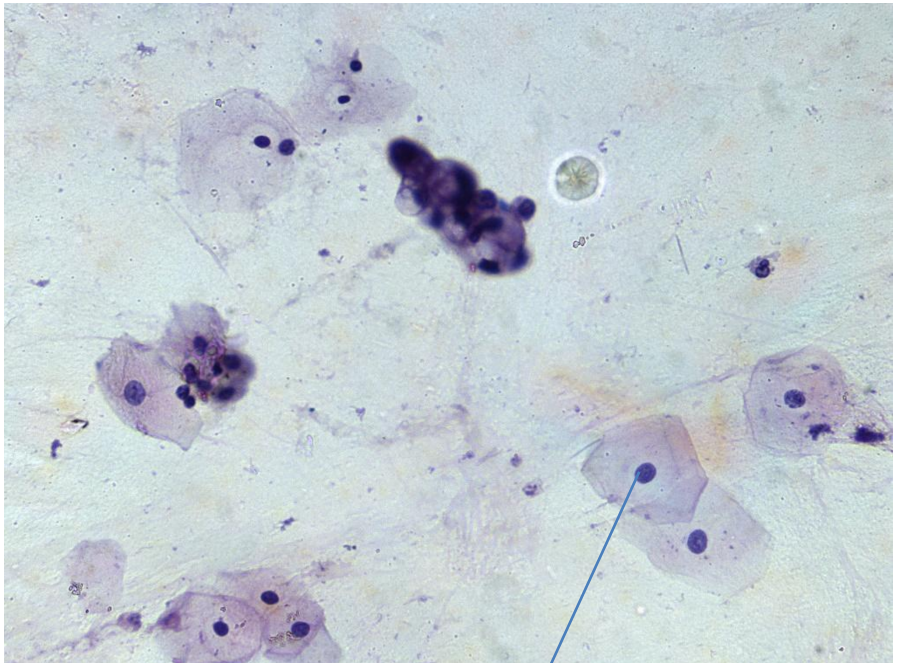
In the case of an image, these states correspond to the gray levels which the individual pixels can adopt.

$$H = -\sum_{k=0}^{M-1} p_k \log_2(p_k)$$

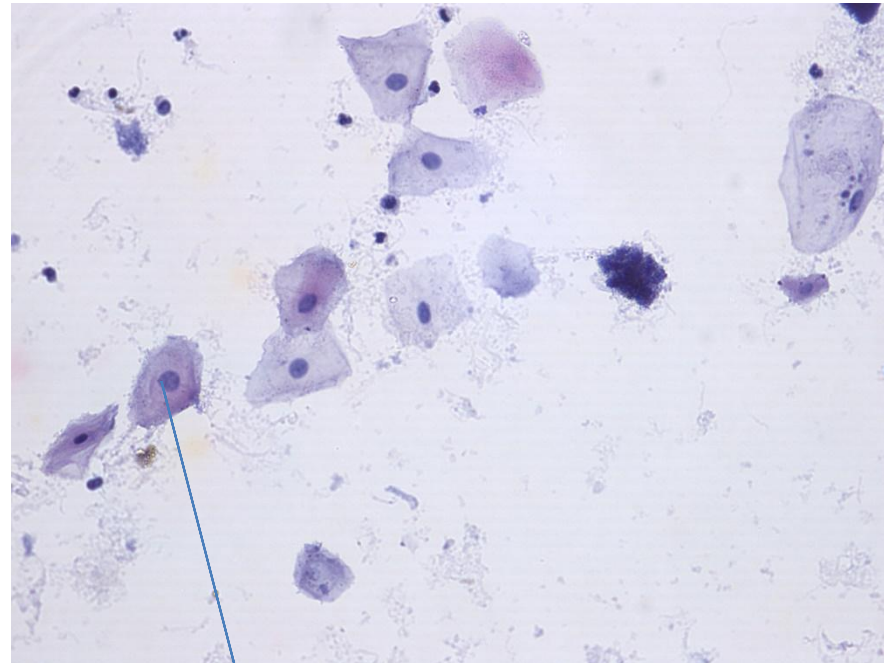
For example, in an 8-bit pixel there are 256 such states. If all such states are equally occupied, as they are in the case of an image which has been perfectly histogram equalized, the spread of states is a maximum, as is the entropy of the image.

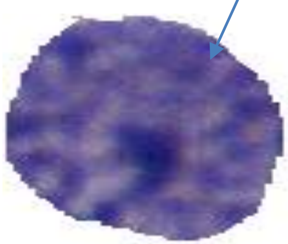


Normal appearing cells (patient with cancer)



Normal cells(patient without cancer)





Cells tested for malignancy associated changes