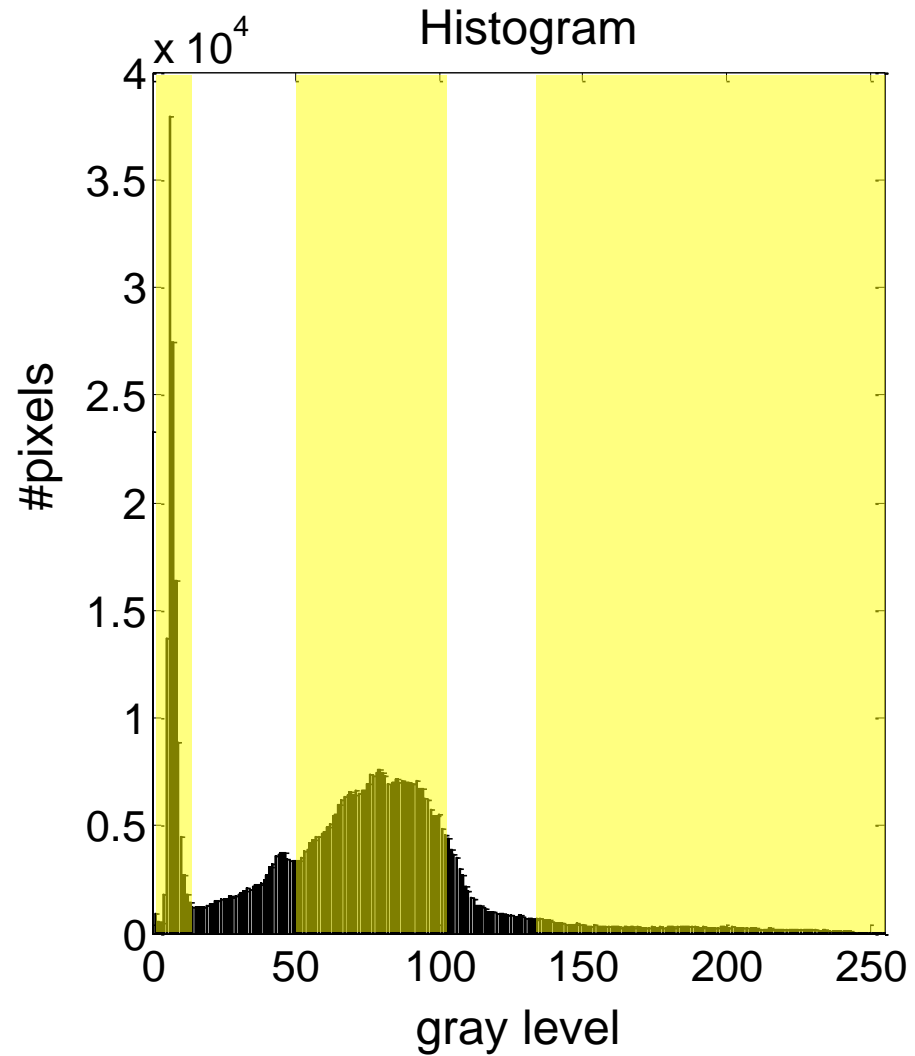
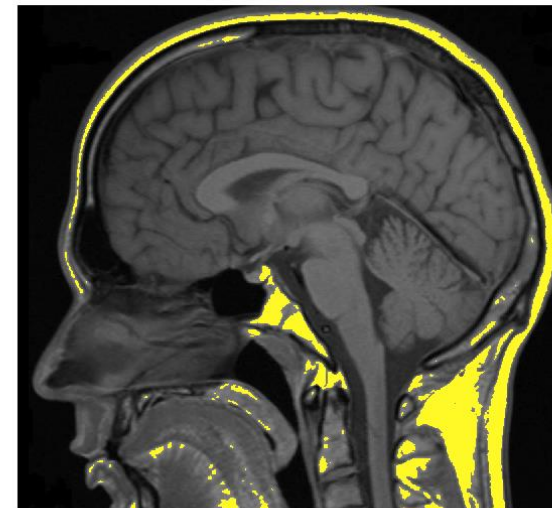


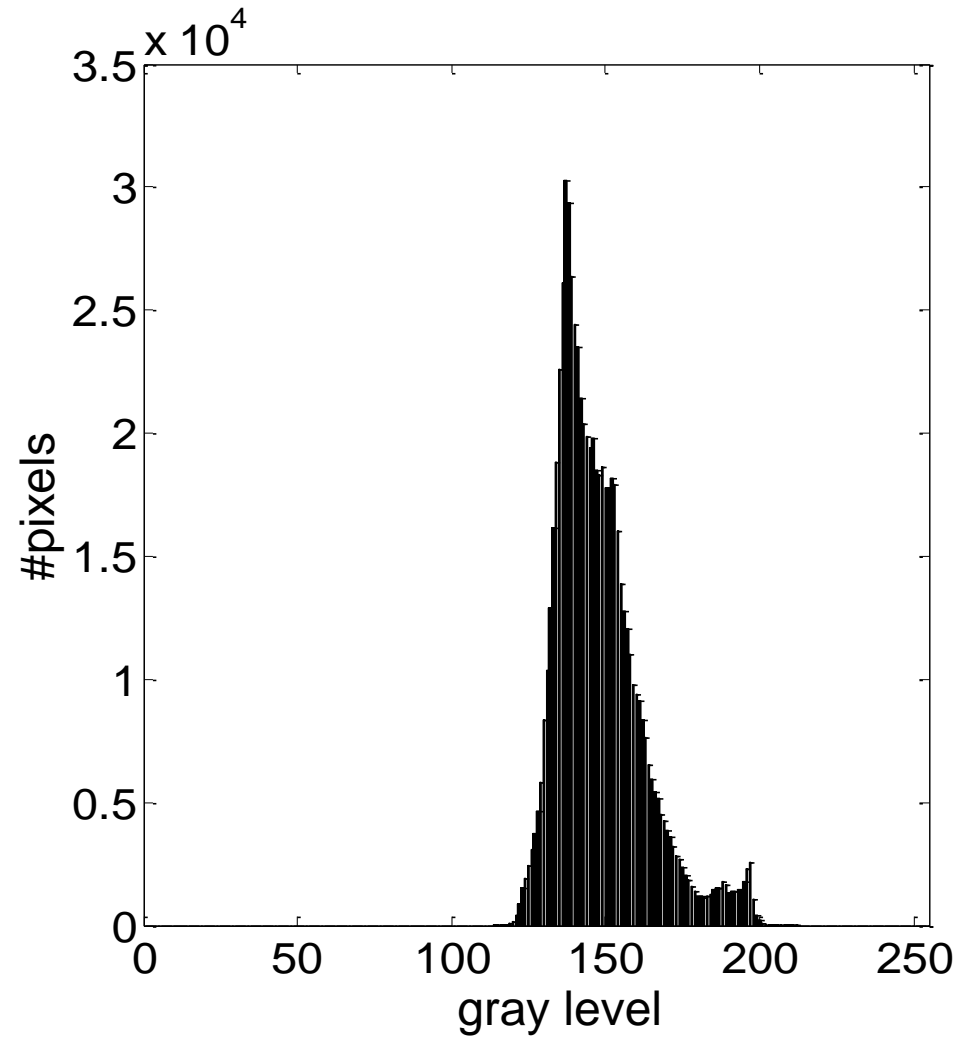
Gray level histograms



Brain image



Gray level histograms



Bay image



Gray level histogram in viewfinder



Gray level histograms

- To measure a histogram:
 - For B-bit image, initialize 2^B counters with 0
 - Loop over all pixels x,y
 - When encountering gray level $f[x,y]=i$, increment counter $\#i$
- Normalized histogram can be thought of as an estimate of the probability distribution of the continuous signal amplitude
- Use fewer, larger bins to trade off amplitude resolution against sample size.

Histogram equalization

Idea:

Find a non-linear transformation

$$g = T(f)$$

that is applied to each pixel of the input image $f[x,y]$, such that a uniform distribution of gray levels results for the output image $g[x,y]$.

Histogram equalization

Analyse ideal, continuous case first ...

Assume

- Normalized input values $0 \leq f \leq 1$ and output values $0 \leq g \leq 1$
- $T(f)$ is differentiable, increasing, and invertible, i.e., there exists

$$f = T^{-1}(g) \quad 0 \leq g \leq 1$$

Goal: pdf $p_g(g) = 1$ over the entire range $0 \leq g \leq 1$

Histogram equalization for continuous case

- From basic probability theory

$$p_f(f) \xrightarrow{f} \boxed{T(f)} \xrightarrow{g} p_g(g) = \left[p_f(f) \frac{df}{dg} \right]_{f=T^{-1}(g)}$$

- Consider the transformation function

$$g = T(f) = \int_0^f p_f(\alpha) d\alpha \quad 0 \leq f \leq 1$$

- Then . . .

$$p_g(g) = \left[p_f(f) \frac{df}{dg} \right]_{f=T^{-1}(g)} = \left[p_f(f) \frac{1}{p_f(f)} \right]_{f=T^{-1}(g)} = 1 \quad 0 \leq g \leq 1$$

\swarrow
 $\frac{dg}{df} = p_f(f)$

Histogram equalization for discrete case

- Now, f only assumes discrete amplitude values f_0, f_1, \dots, f_{L-1} with empirical probabilities

$$P_0 = \frac{n_0}{n} \quad P_1 = \frac{n_1}{n} \quad \dots \quad P_{L-1} = \frac{n_{L-1}}{n} \quad \text{where } n = \sum_{l=0}^{L-1} n_l \quad \leftarrow \text{pixel count for amplitude } f_l$$

- Discrete approximation of $g = T(f) = \int_0^f p_f(\alpha) d\alpha$

$$g_k = T[f_k] = \sum_{i=0}^k P_i \quad \text{for } k = 0, 1, \dots, L-1$$

- The resulting values g_k are in the range $[0, 1]$ and might have to be scaled and rounded appropriately.

Histogram equalization example



Original image *Bay*

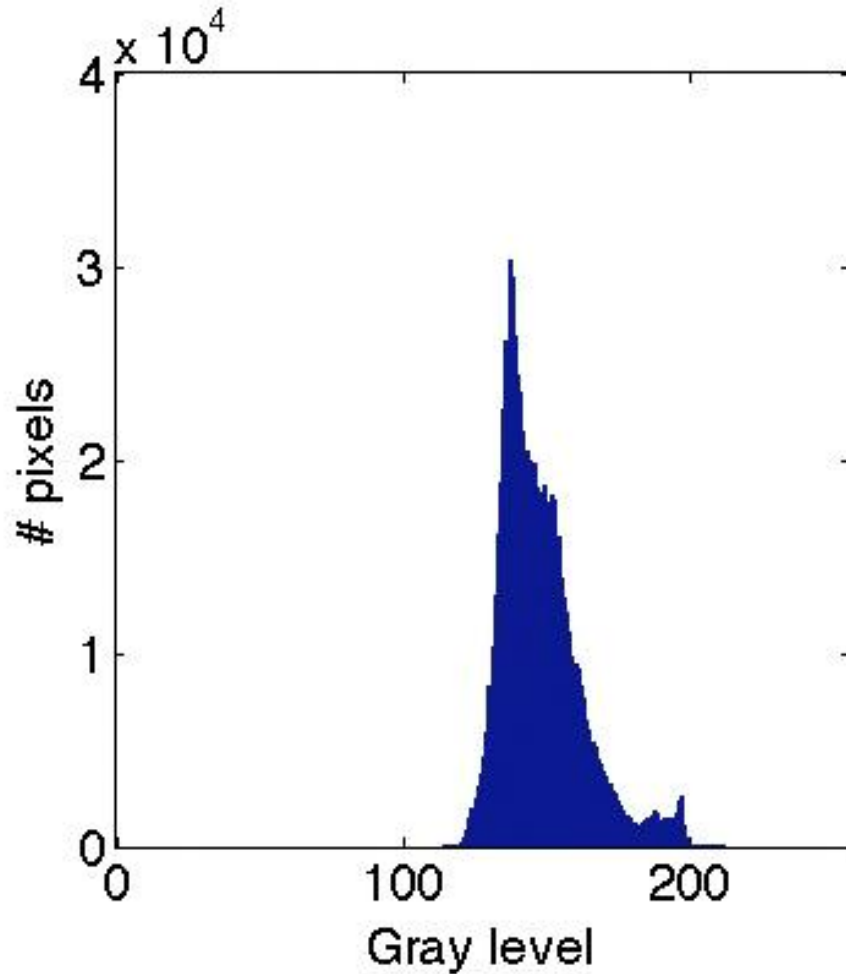


... after histogram equalization

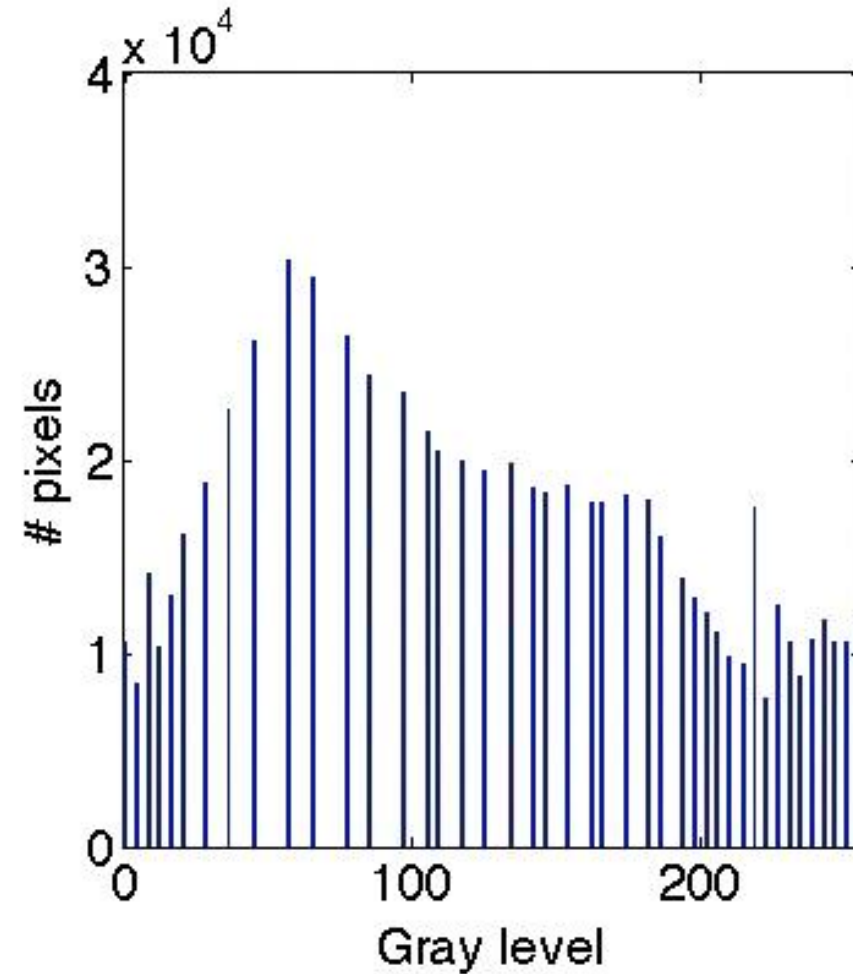


Histogram equalization example

Original image *Bay*



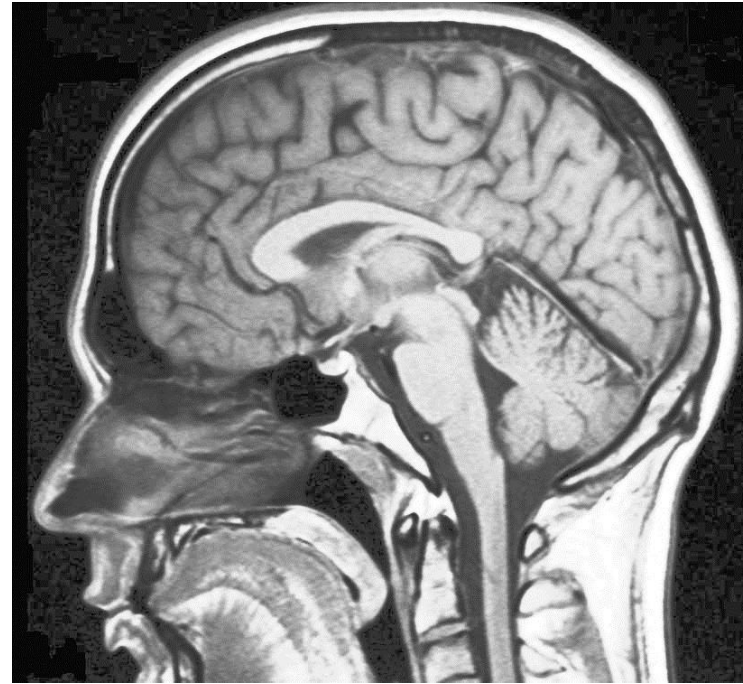
... after histogram equalization



Histogram equalization example



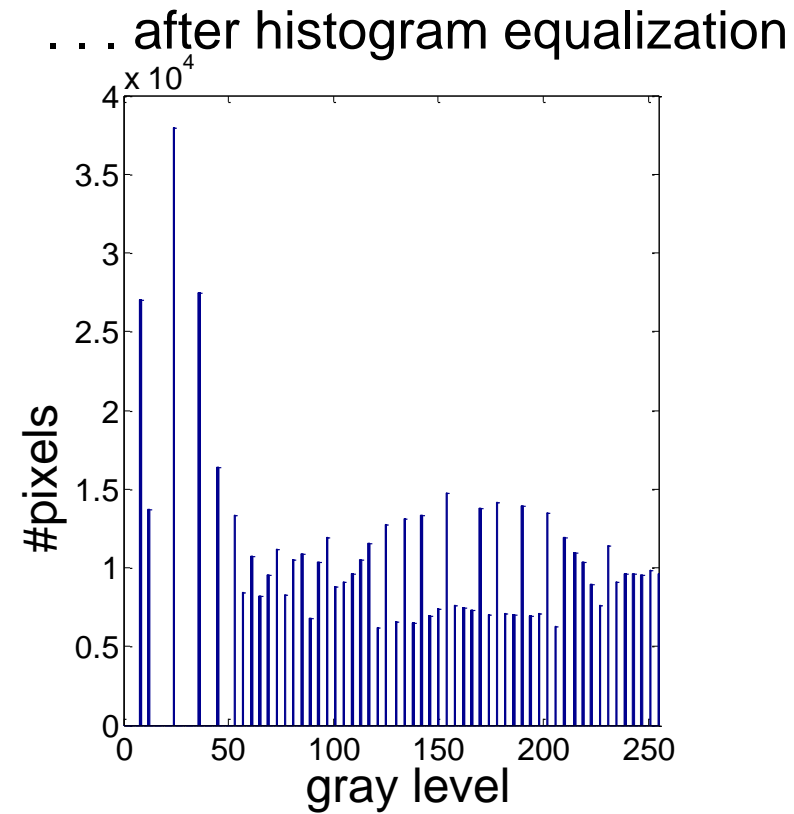
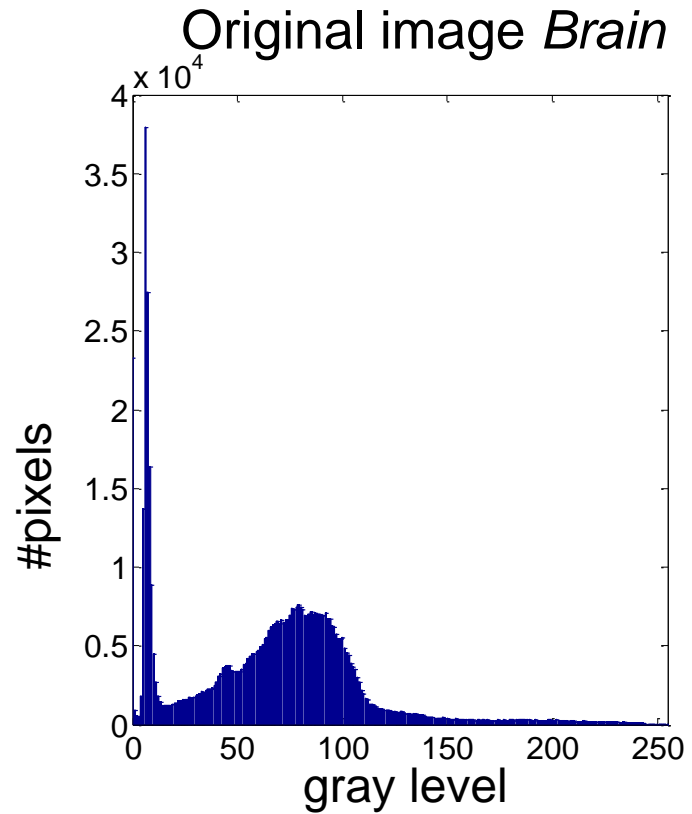
Original image *Brain*



... after histogram equalization



Histogram equalization example



Histogram equalization example



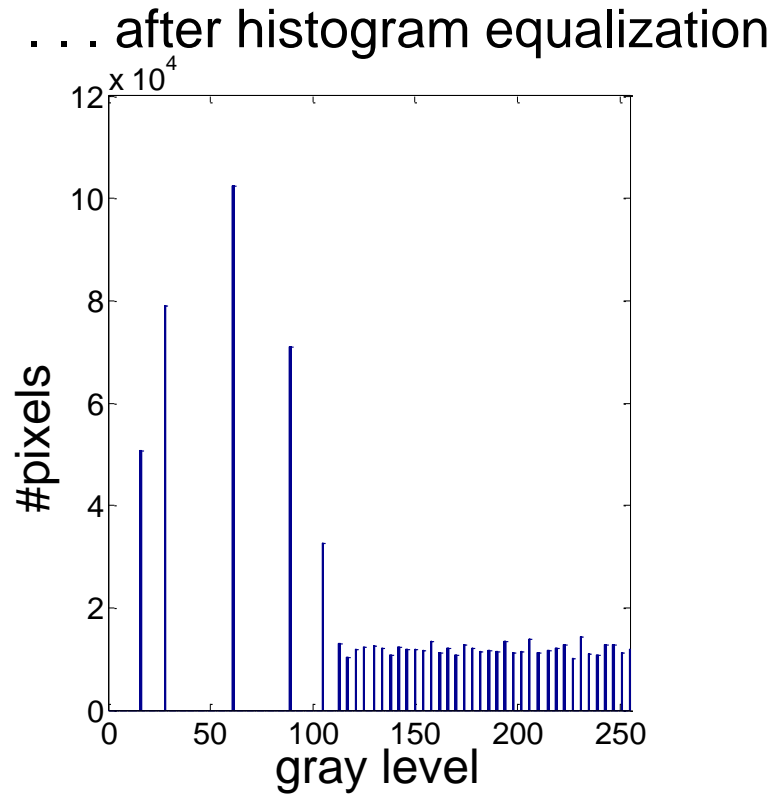
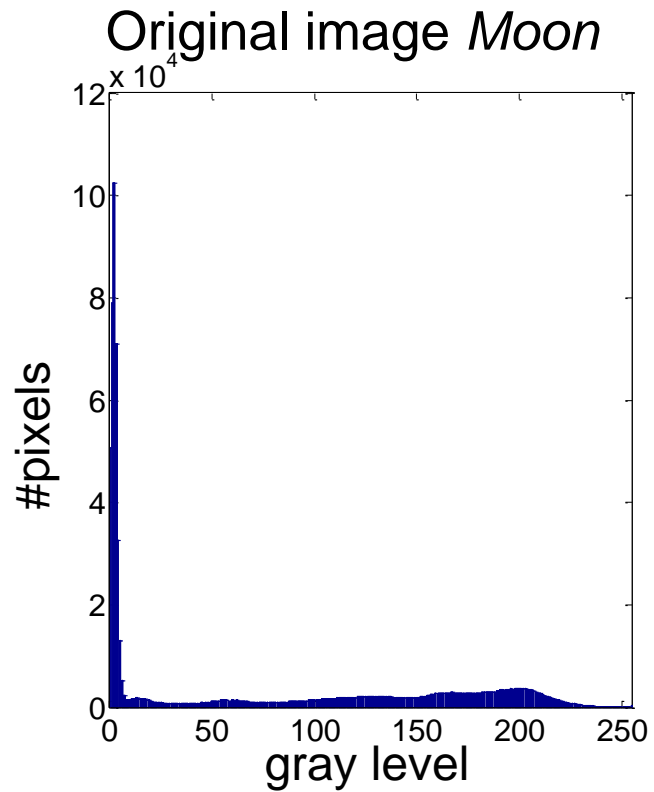
Original image *Moon*



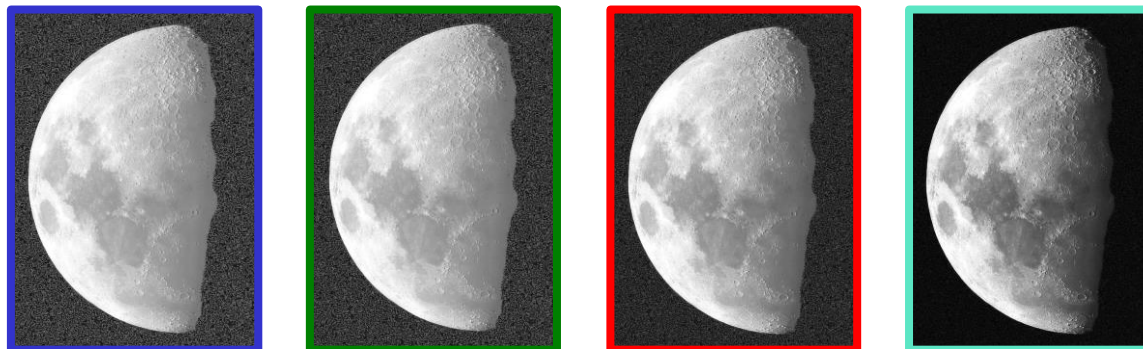
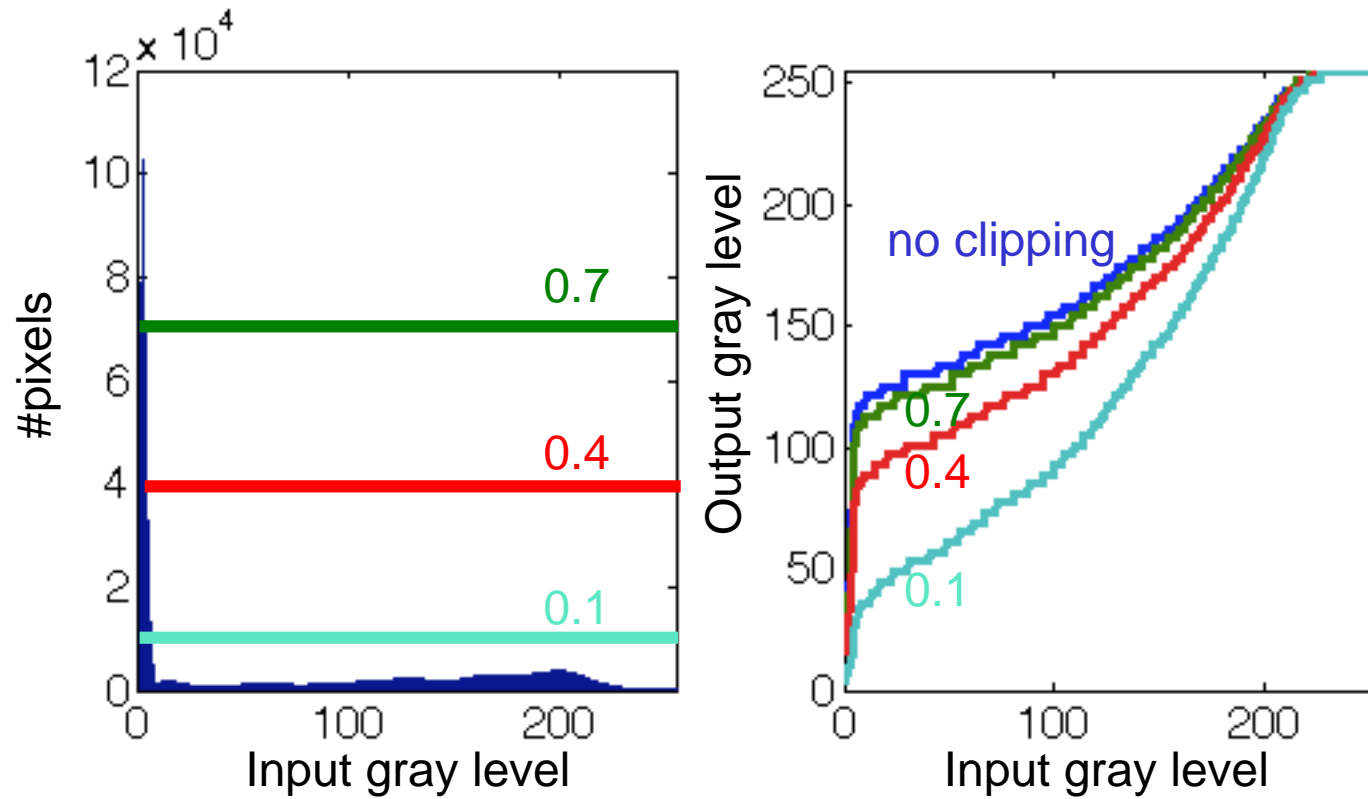
... after histogram equalization



Histogram equalization example

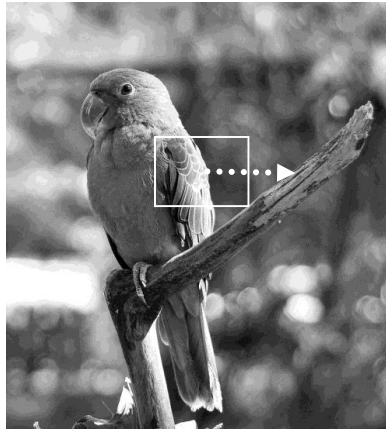


Contrast-limited histogram equalization

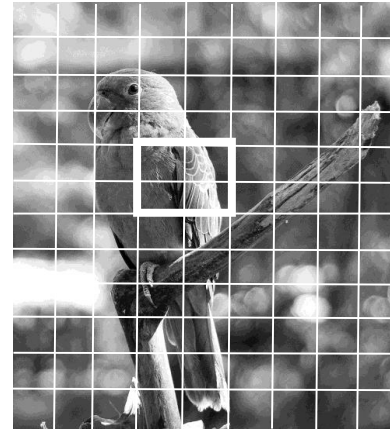


Adaptive histogram equalization

- Histogram equalization based on a histogram obtained from a portion of the image



Sliding window approach:
different histogram (and
mapping) for every pixel



Tiling approach:
subdivide into overlapping
regions, mitigate blocking
effect by smooth blending
between neighboring tiles

- Limit contrast expansion in flat regions of the image, e.g., by clipping histogram values.
("Contrast-limited adaptive histogram equalization")

[Pizer, Amburn et al. 1987]

Adaptive histogram equalization

Original image
Parrot



Global histogram
equalization



Adaptive histogram
equalization, 8x8 tiles

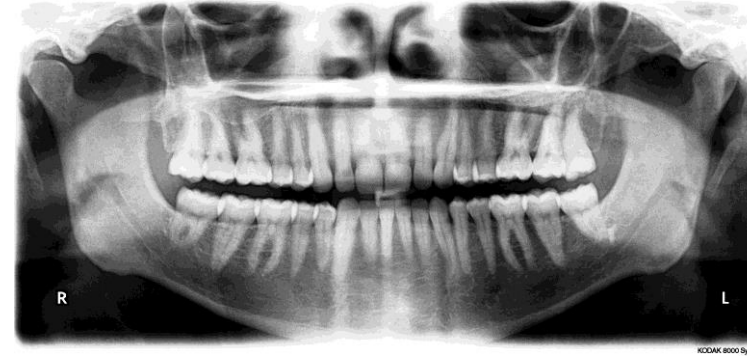
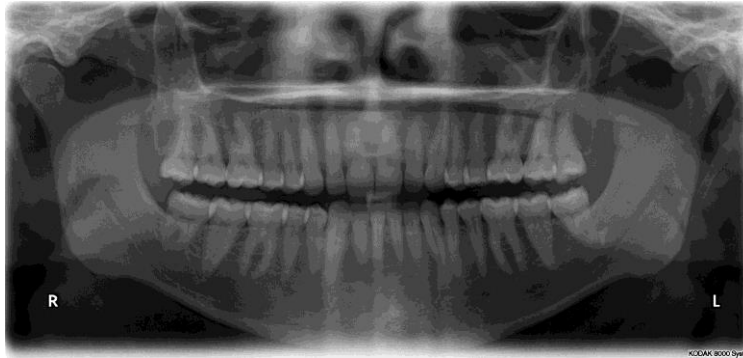


Adaptive histogram
equalization, 16x16 tiles



Adaptive histogram equalization

Original image
Dental Xray



Global histogram
equalization

Adaptive histogram
equalization, 8x8 tiles

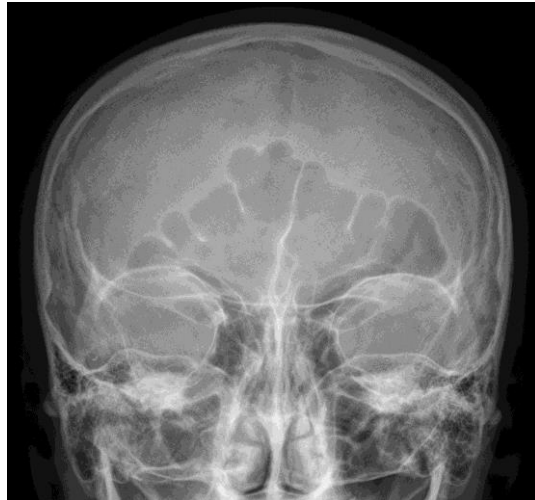


Adaptive histogram
equalization, 16x16 tiles



Adaptive histogram equalization

Original image
Skull Xray



Global histogram
equalization



Adaptive histogram
equalization, 8x8 tiles



Adaptive histogram
equalization, 16x16 tiles

