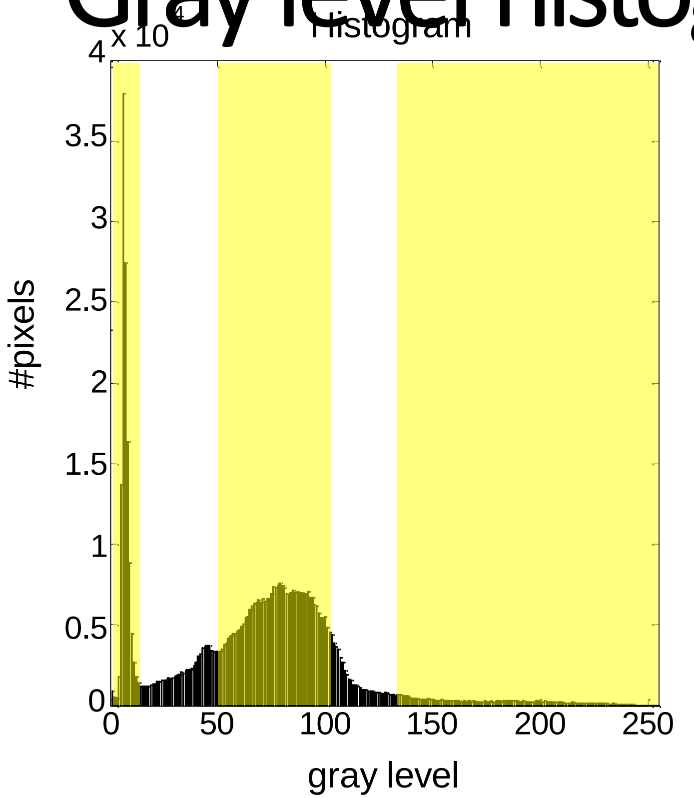
Tratamiento de Señales
Version 2024-1

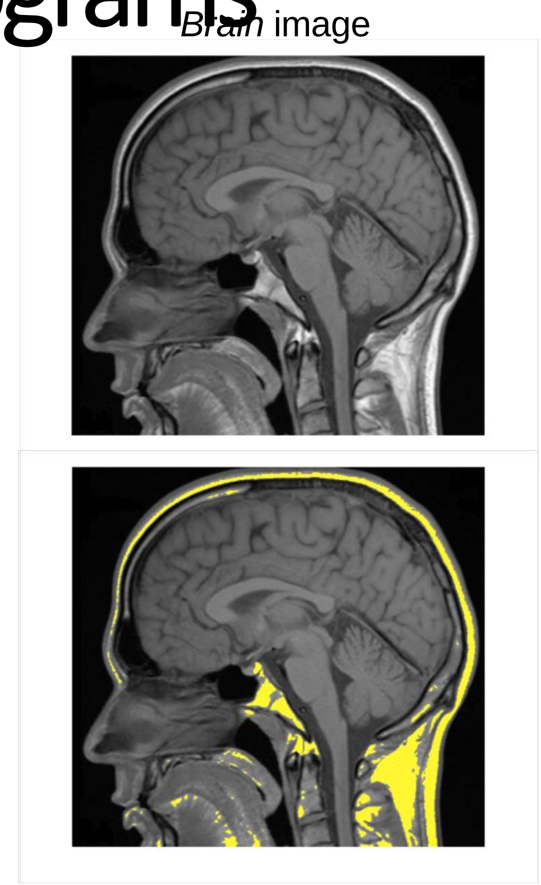
Ecualización de Histogramas

[Capítulo 3]

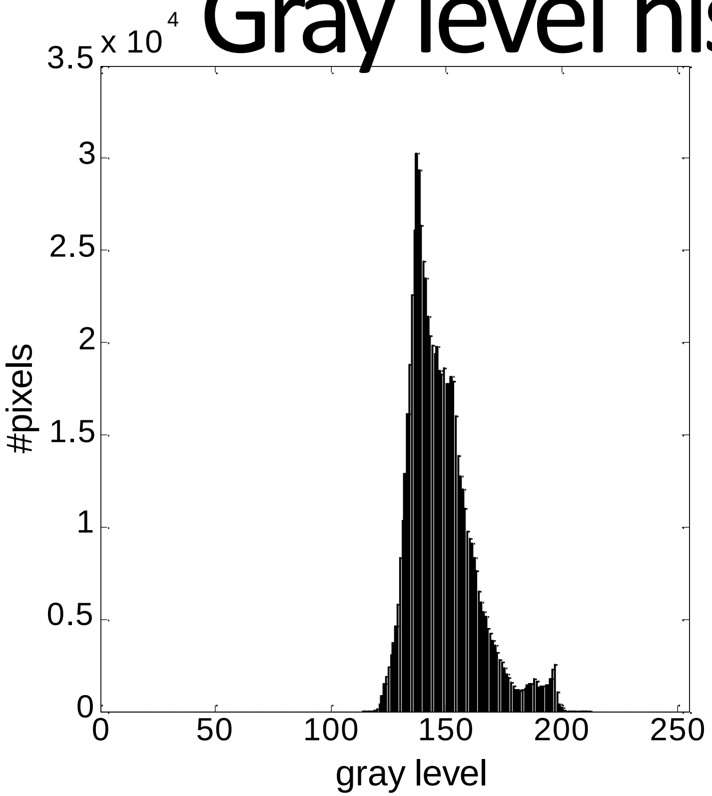
Dr. José Ramón Iglesias
DSP-ASIC BUILDER GROUP
Director Semillero TRIAC
Ingenieria Electronica
Universidad Popular del Cesar

Gray level histograms





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 $\#\iota$
- 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

$$\boxed{g = T(f)}$$

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

$$f = T^{-1}(g) \qquad 0 \le g \le 1$$

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

Histogram equalization for continuous case

- From basic probability theory

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

- Consider the transformation function

$$\boxed{g = T(f) = \int_0^f p_f(\alpha)\, d\alpha \qquad 0 \le f \le 1}$$

- Then . . .

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

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

Histogram equalization for discrete case

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

$$P_0 = \frac{n_0}{n} \qquad P_1 = \frac{n_1}{n} \qquad ! \qquad P_{L-1} = \frac{n_{L-1}}{n} \qquad \text{where } n \text{ is total number of pixels}$$

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

$$\boxed{g_k = T\left[f_k\right] = \sum_{i=0}^{k} P_i \qquad \text{for } \; k = 0,1...,L-1}$$

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

Histogram equalization

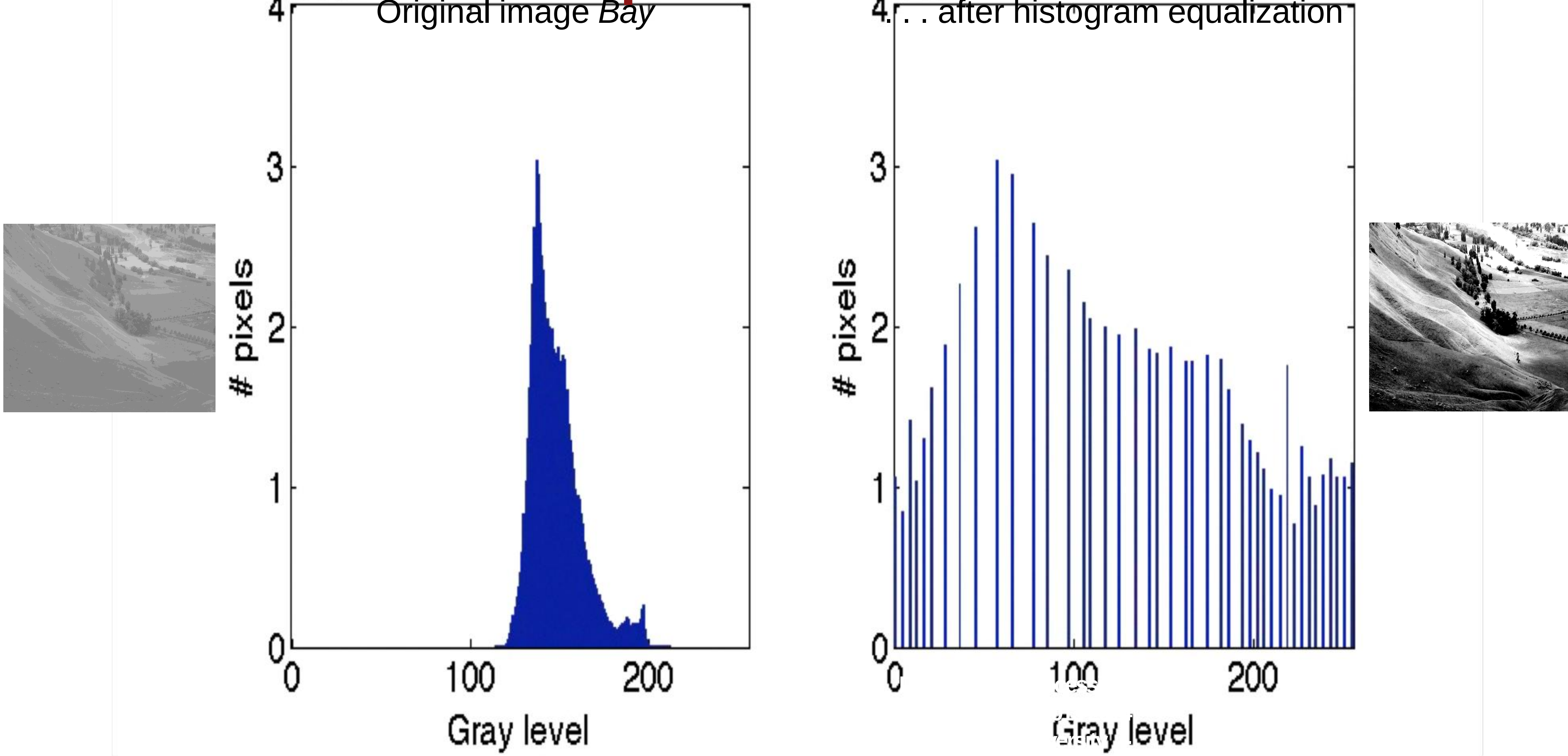
Original image *Bay*

... after histogram equalization

Histogram equalization example

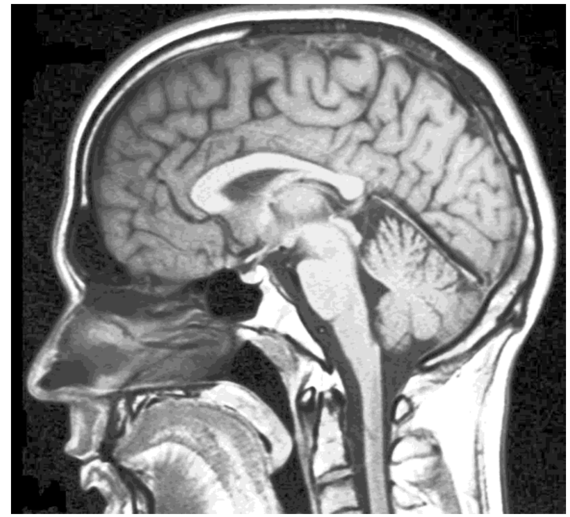
Original image *Bay*

. . . after histogram equalization



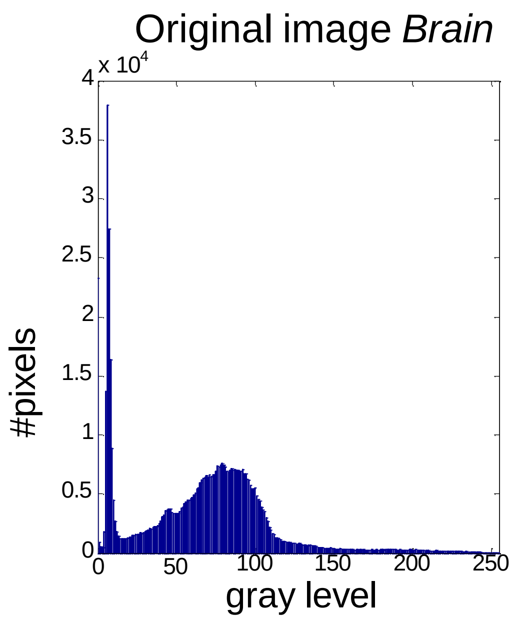
Histogram equalization example

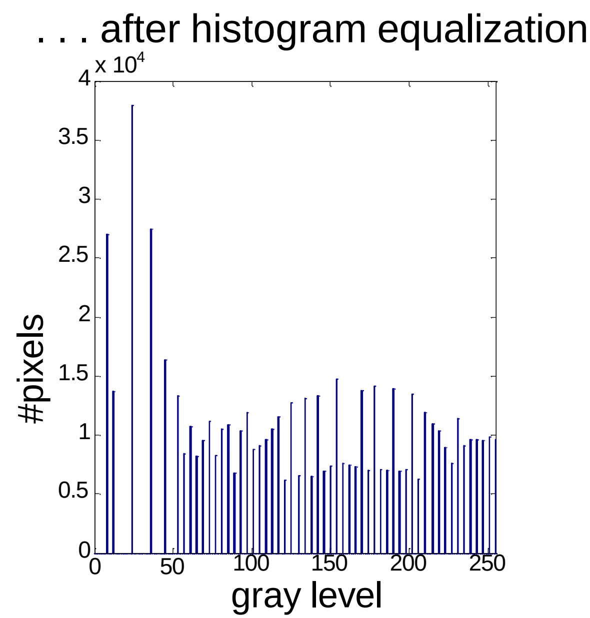
Original image *Brain*

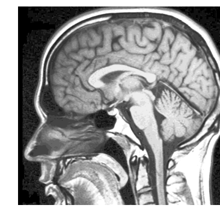


... after histogram equalization

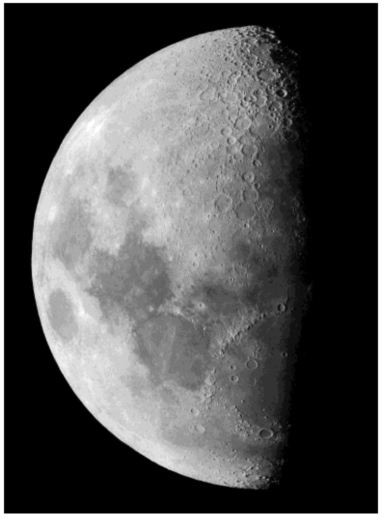
Histogram equalization example



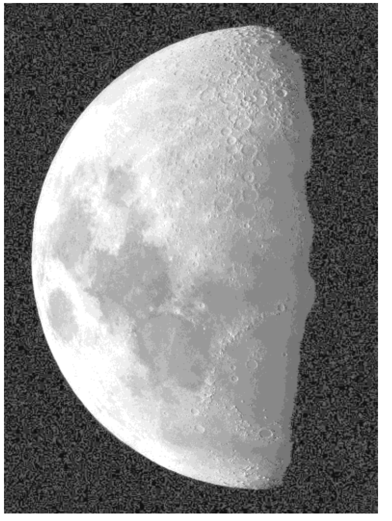




Histogram equalization example



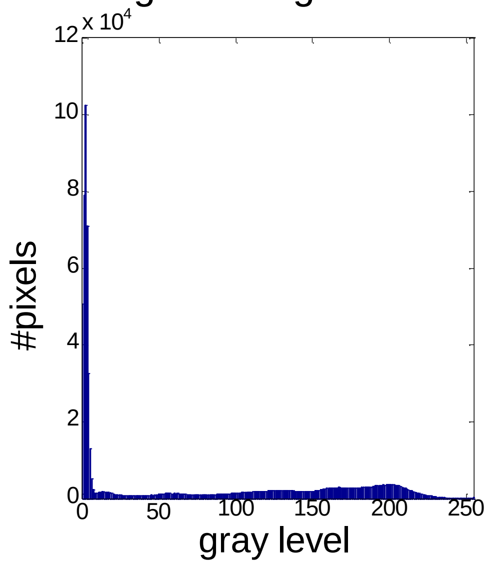
Original image *Moon*



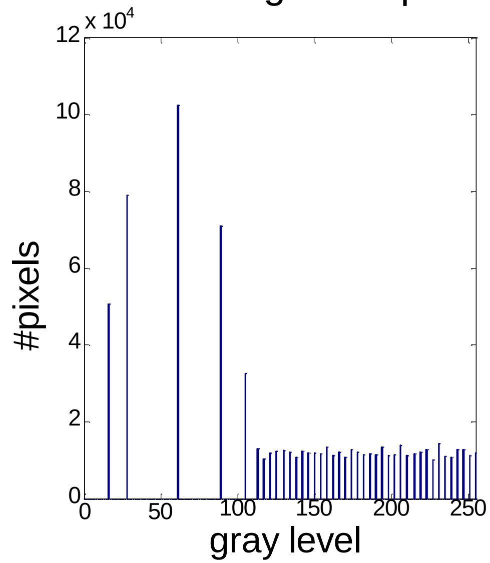
... after histogram equalization

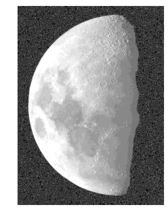
Histogram equalization example

Original image *Moon*

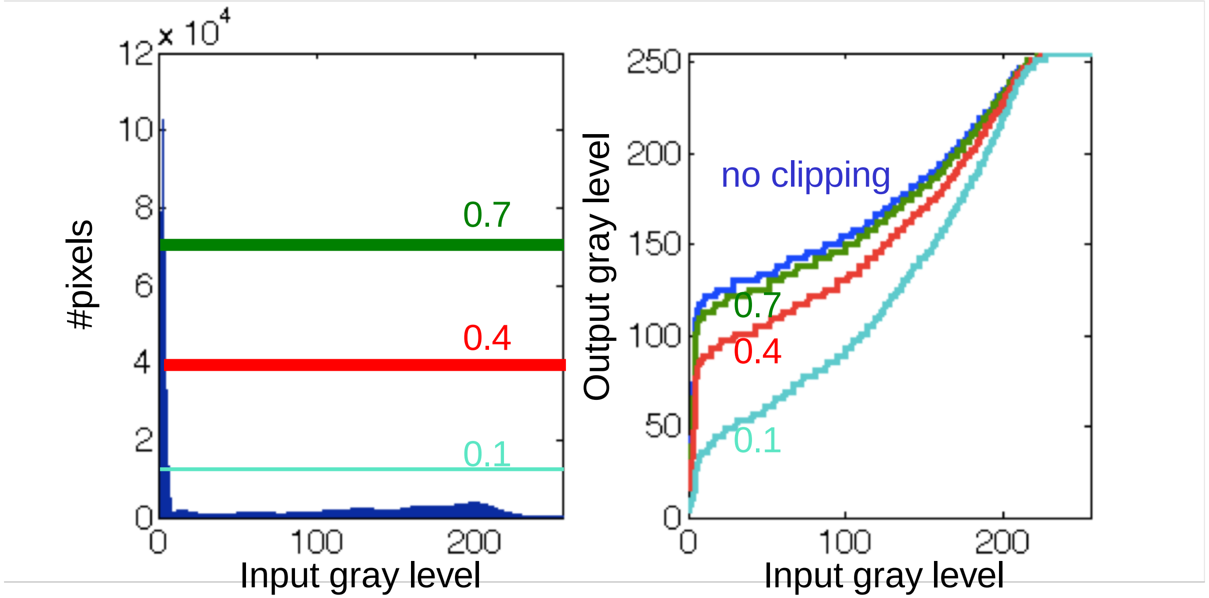


. . . after histogram equalization



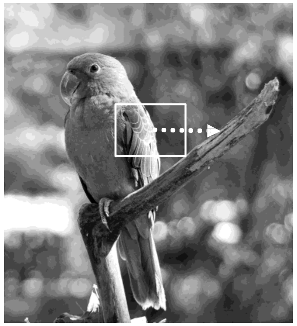


Contrast-limited histogram equalization

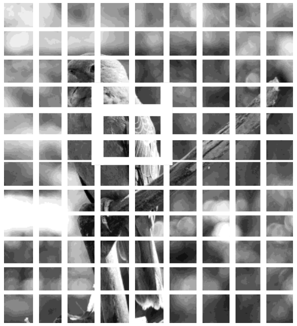


Adaptive histogram equalization

- 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

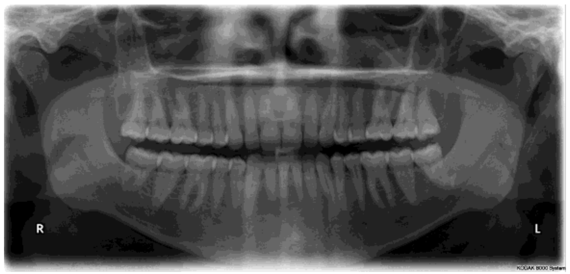
Global histogram equalization

Adaptive histogram equalization, 8x8 tiles

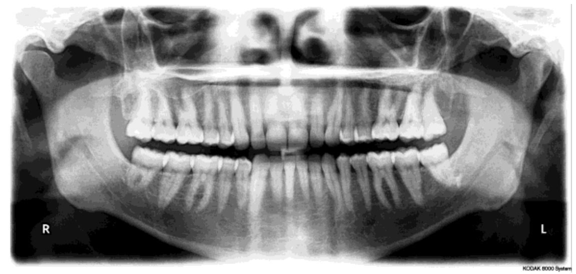
Adaptive histogram equalization, 16x16 tiles

Adaptive histogram equalization

Original image
Dental Xray



Global histogram equalization

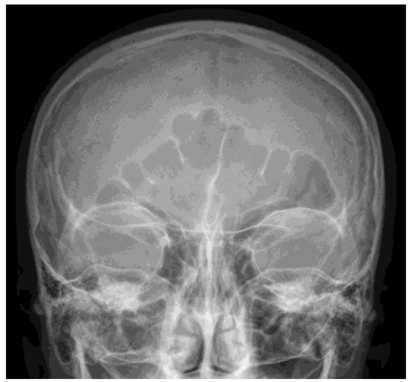


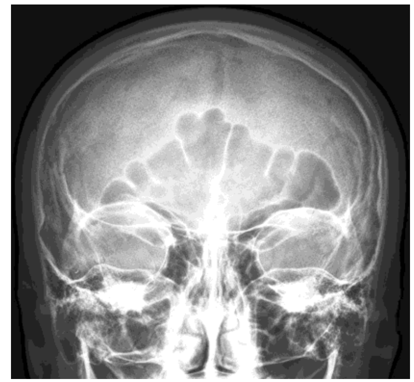
Adaptive histogram equalization, 8x8 tiles

Adaptive histogram equalization, 16x16 tiles

Adaptive histogram equalization

Original image
Skull Xray

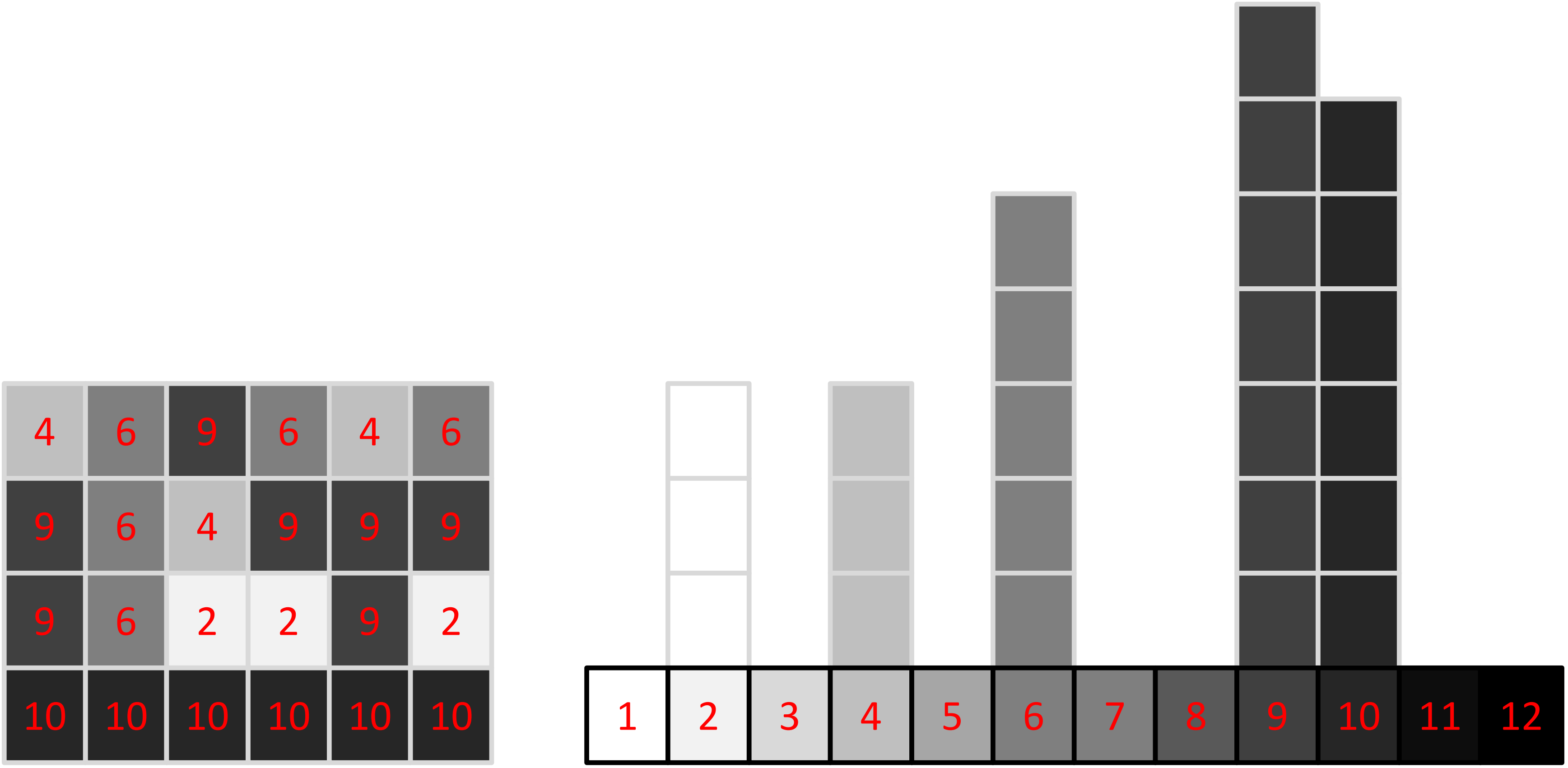


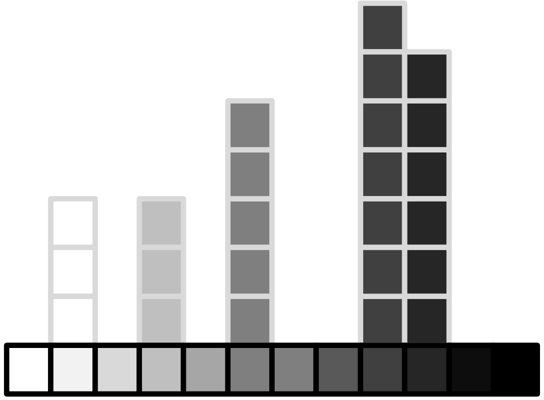


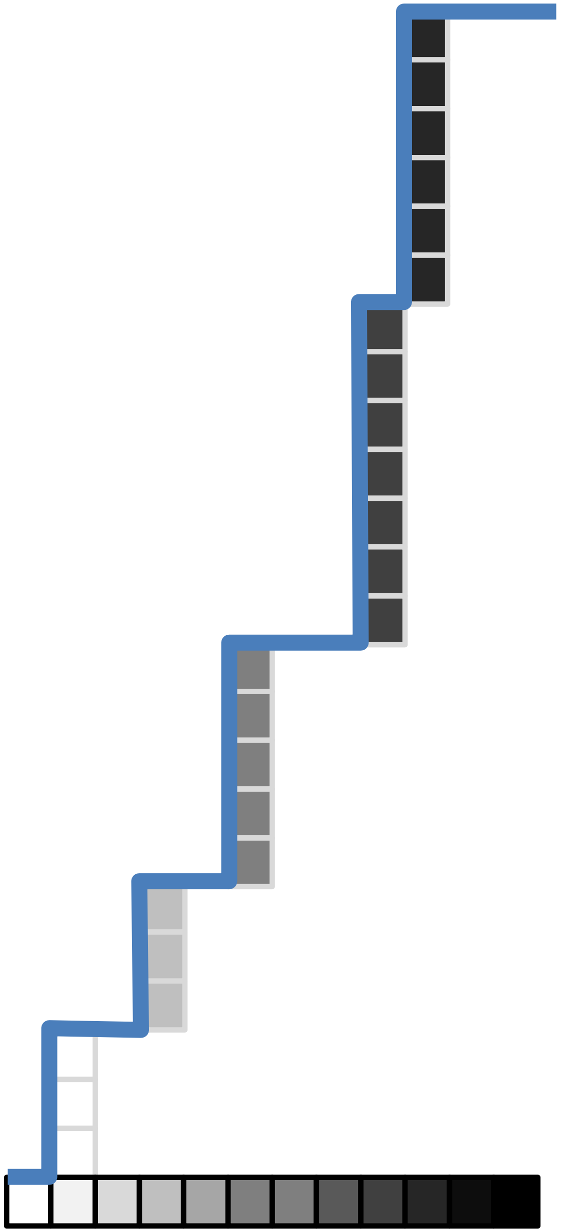
Global histogram
equalization

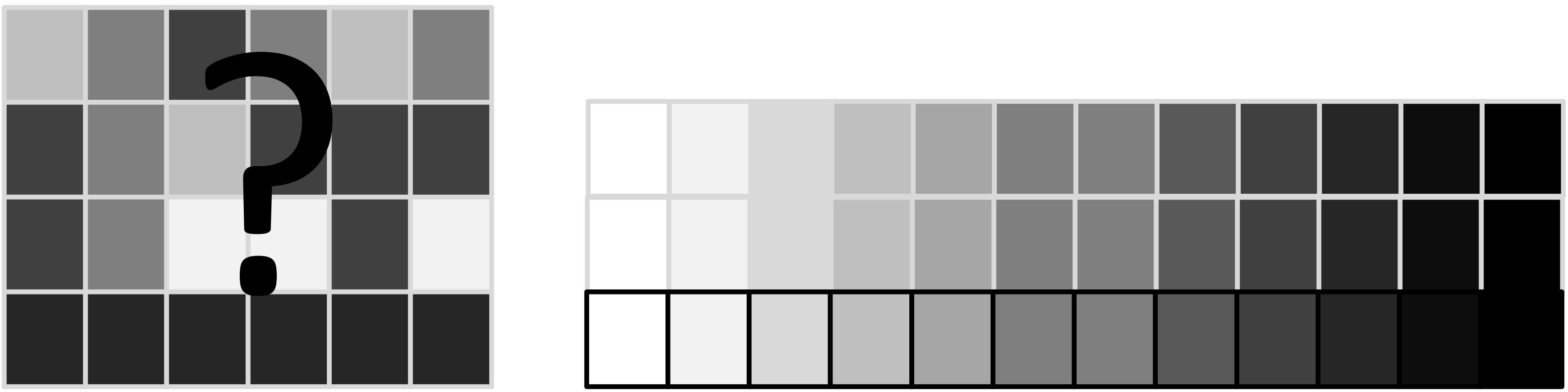
Adaptive histogram
equalization, 8x8 tiles

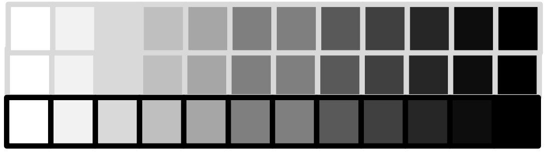
Adaptive histogram
equalization, 16x16 tiles

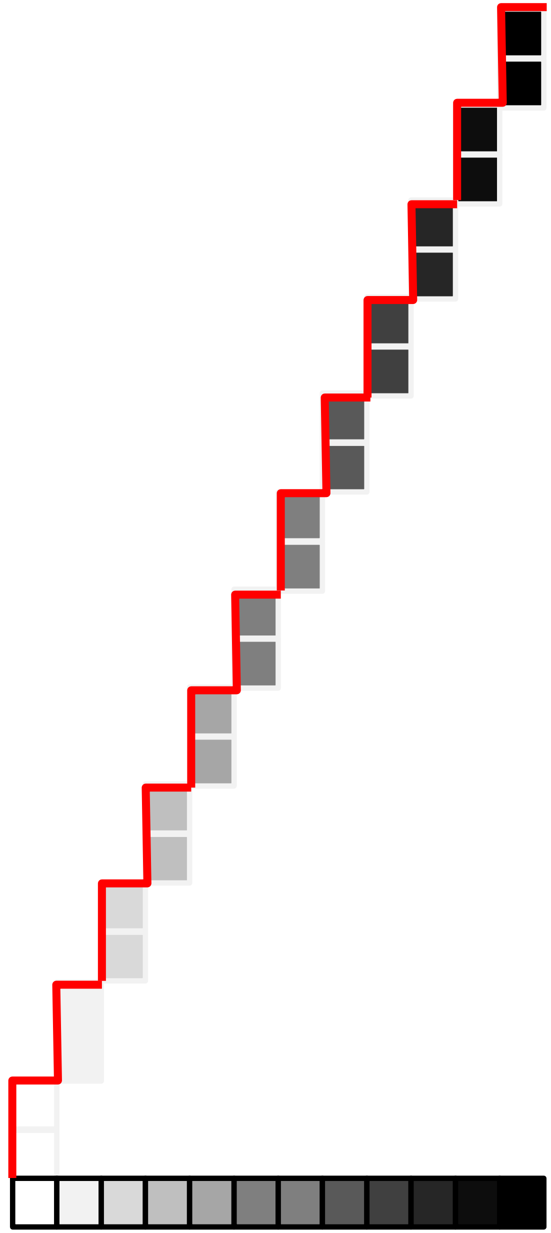

4 6 9 6 4 6
9 6 4 9 9 9
9 6 2 2 9 2
10 10 10 10 10 10
1 2 3 4 5 6 7 8 9 10 11 12

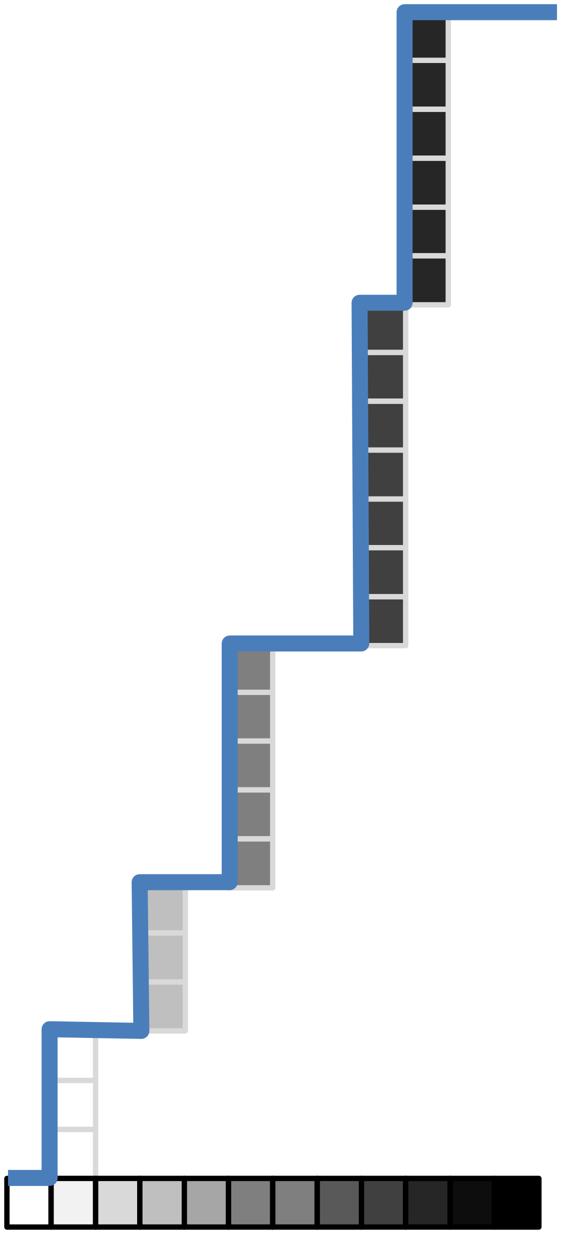


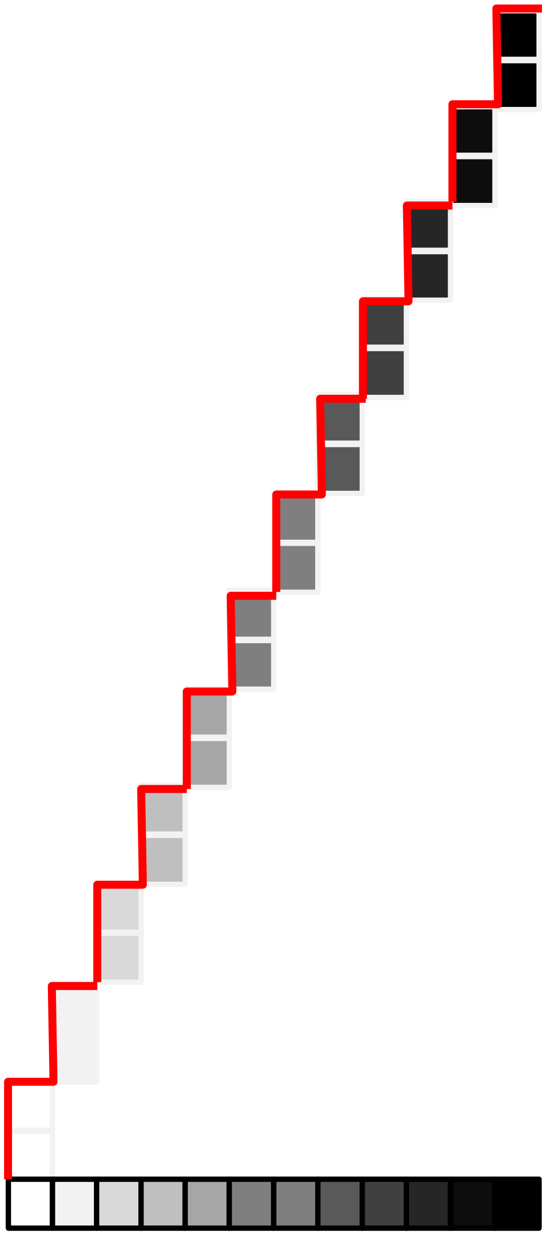


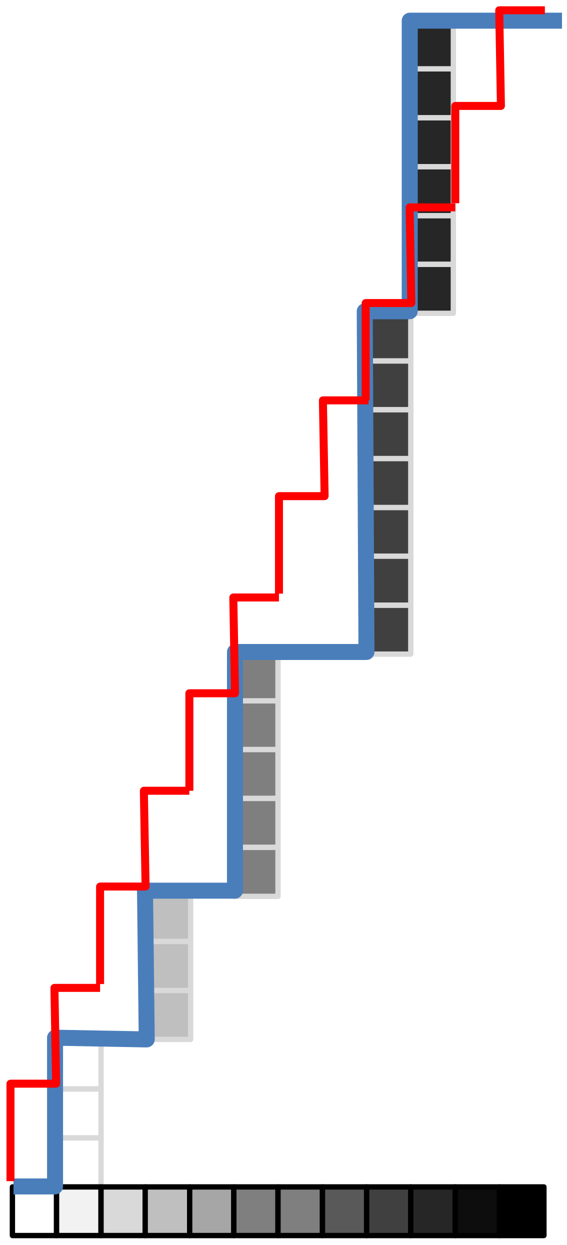




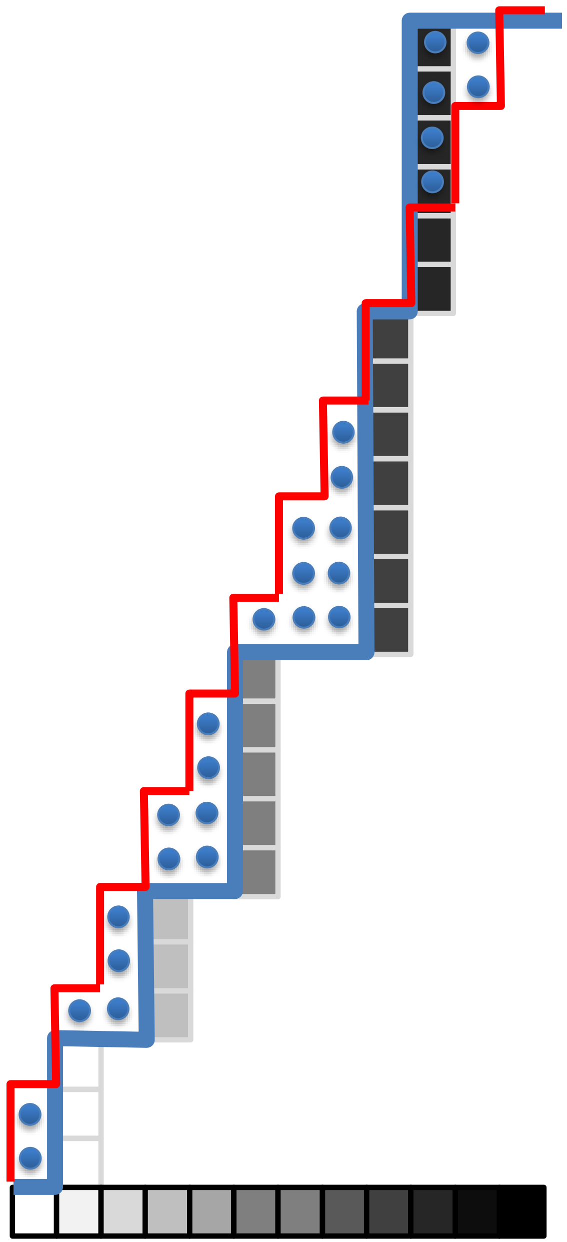






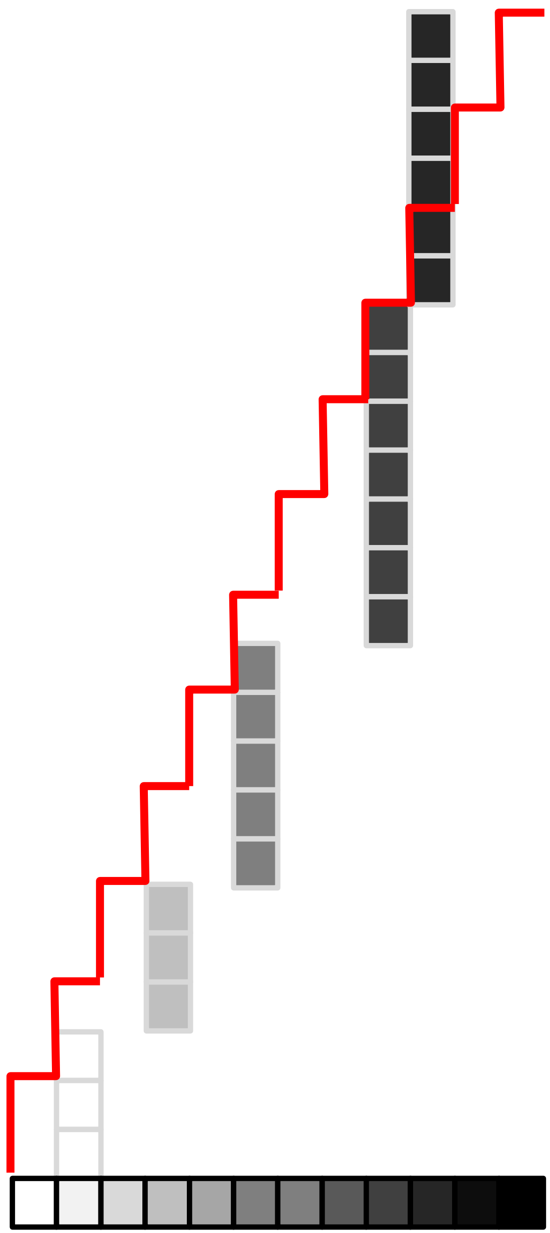


How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

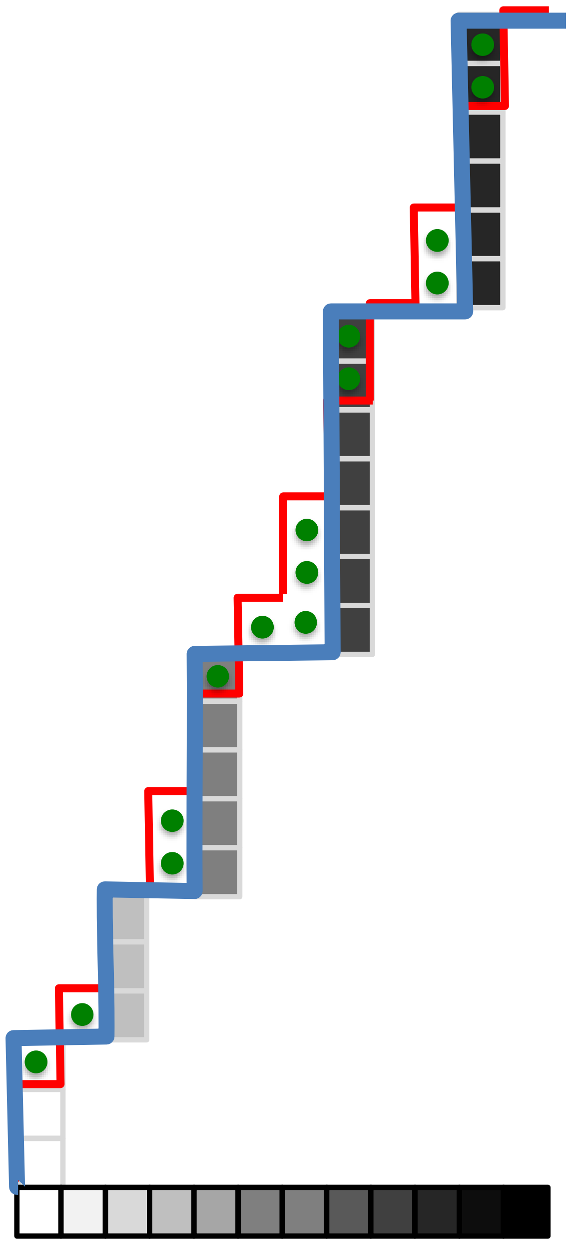


How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

Error = 27 (circles)

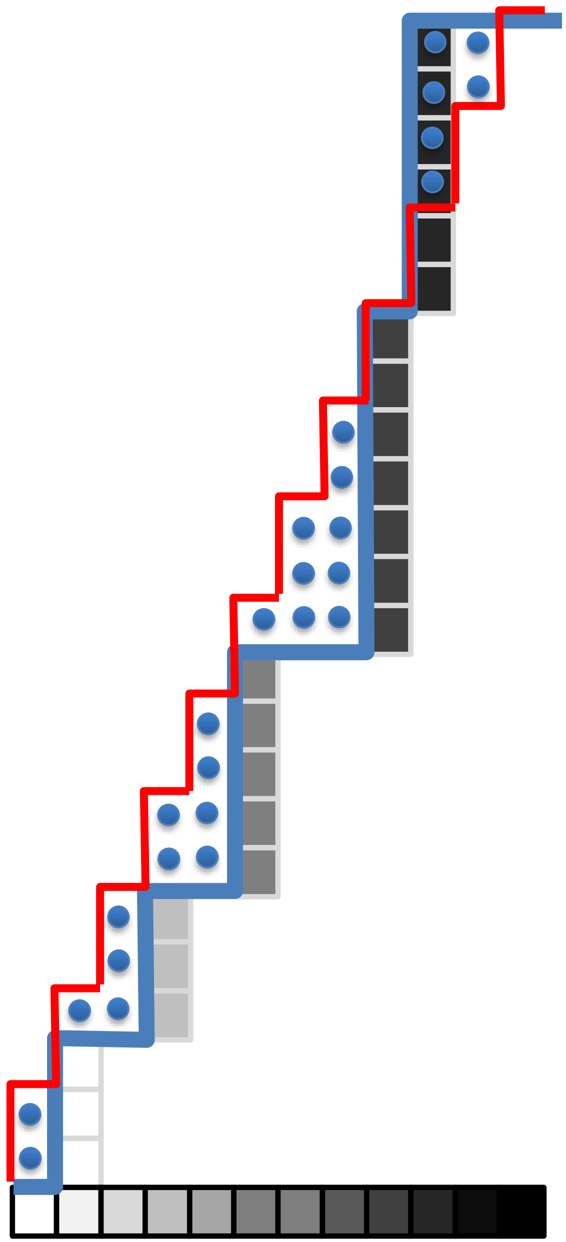


How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?



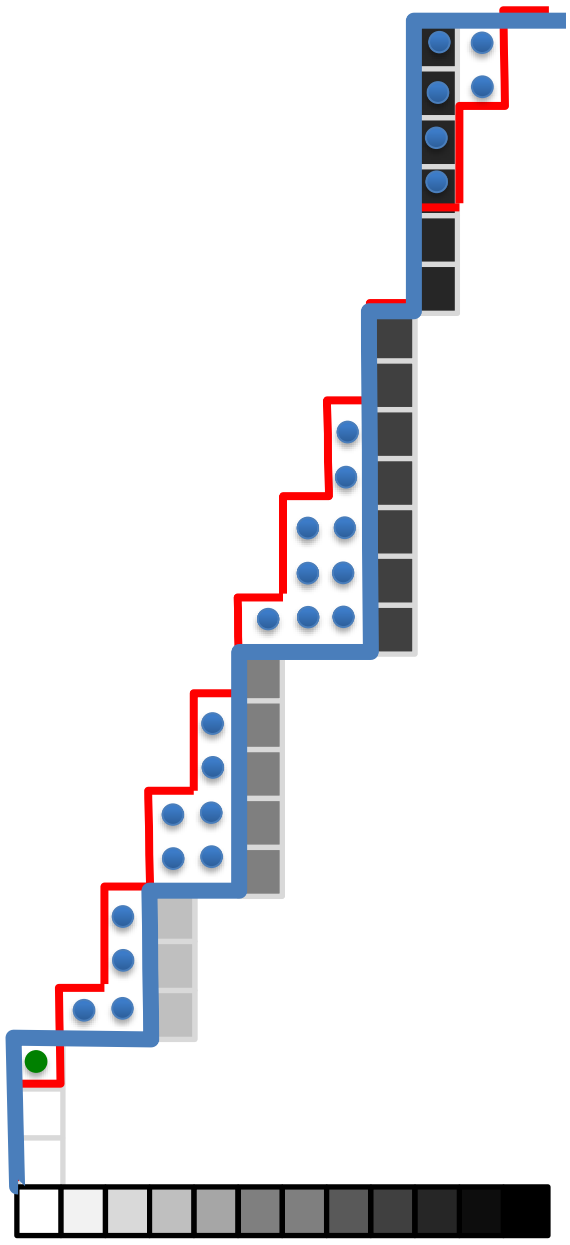
How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

Error = 15



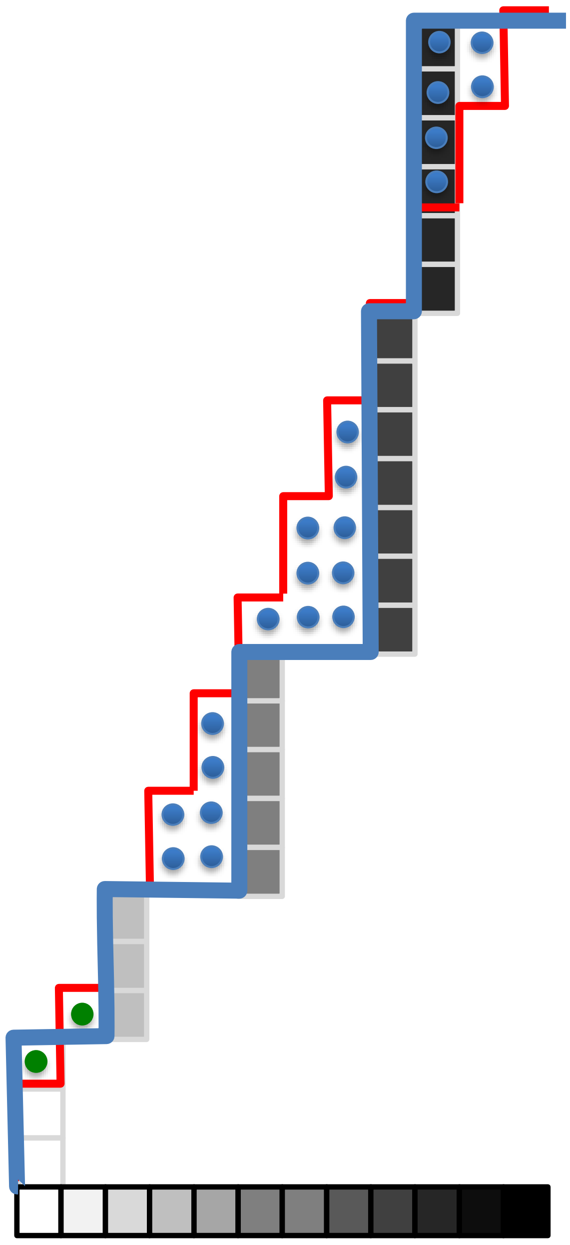
How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

Error = 27 (circles)



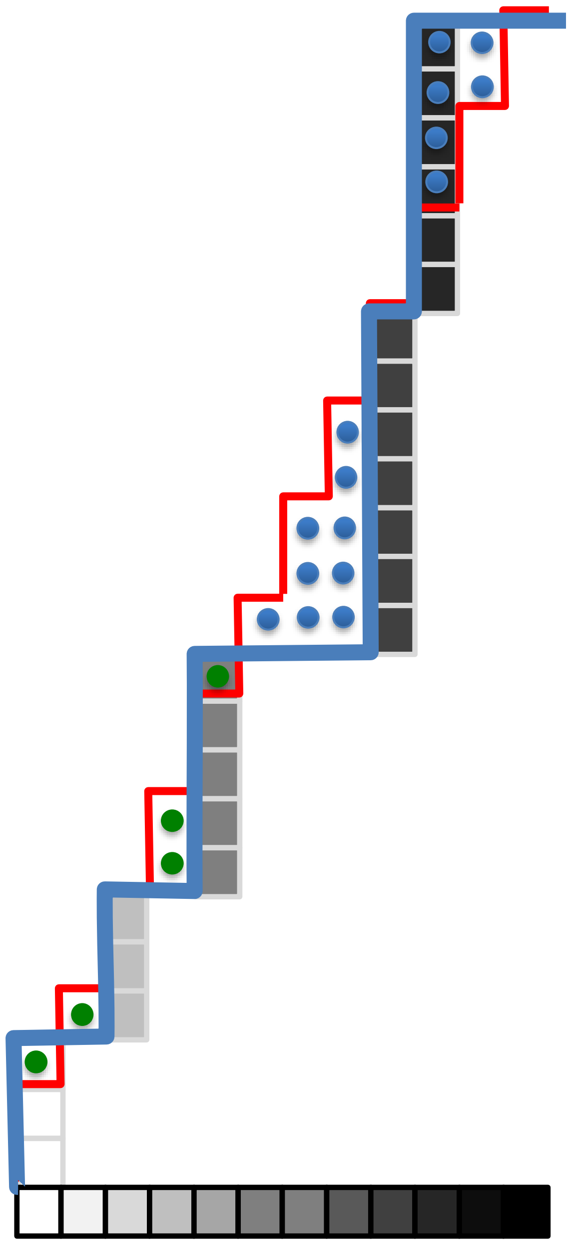
How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

Error = 26 (circles)



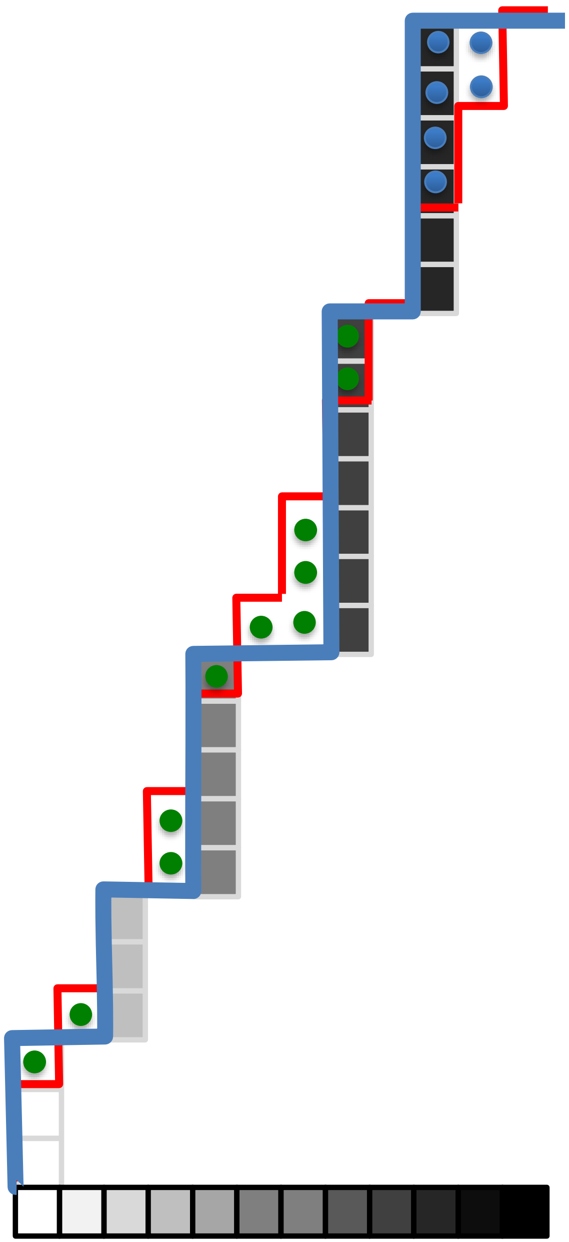
How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

Error = 23



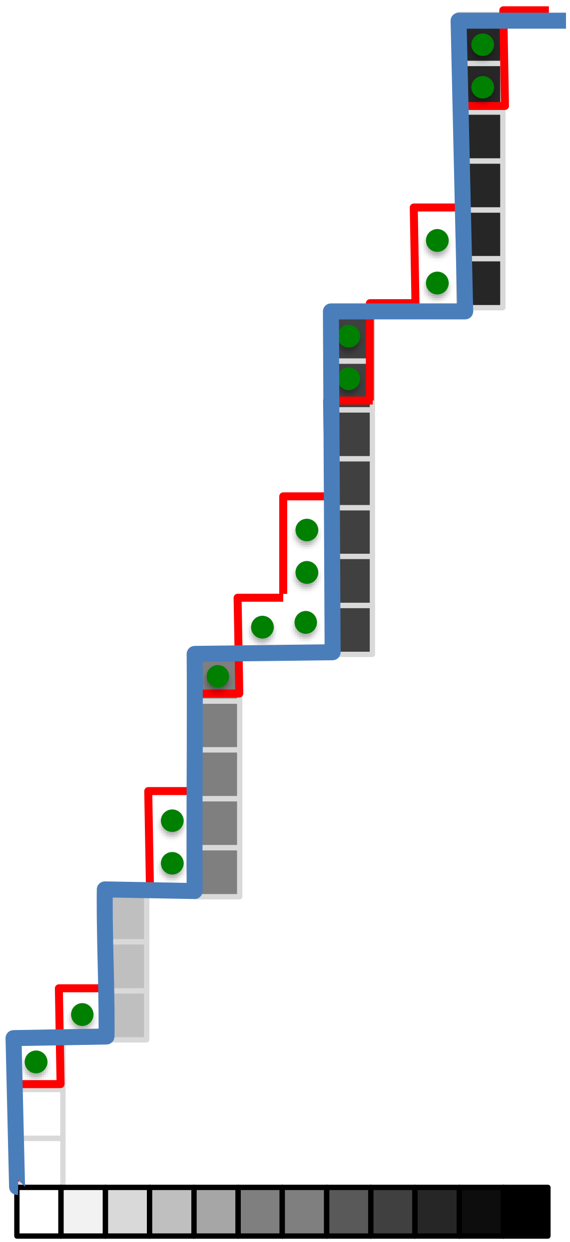
How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

Error = 20



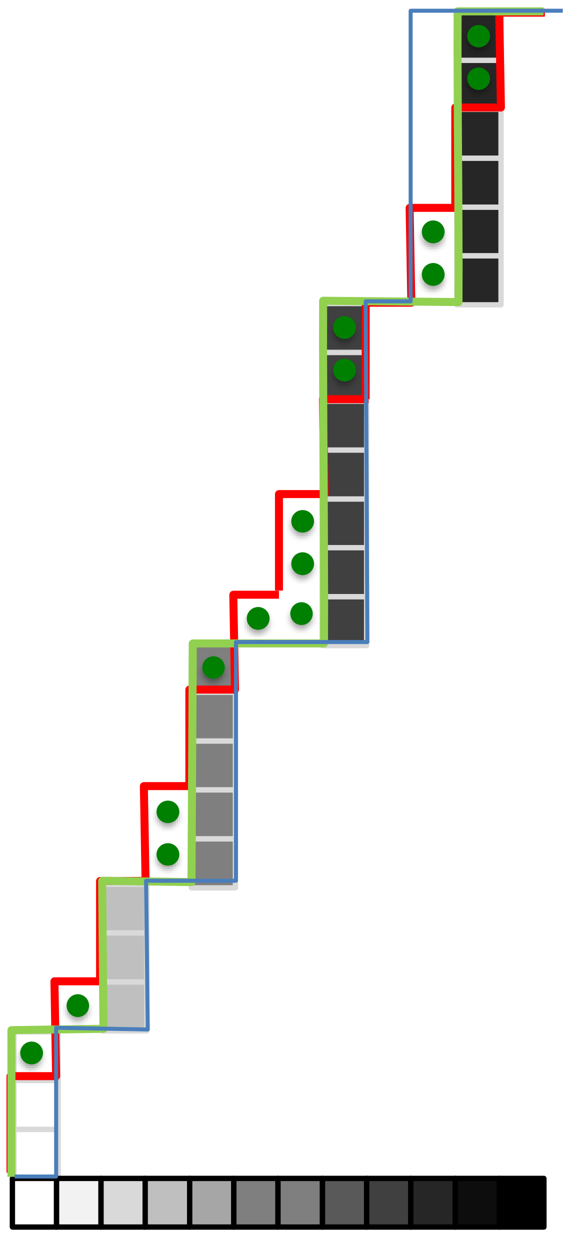
How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

Error = 17



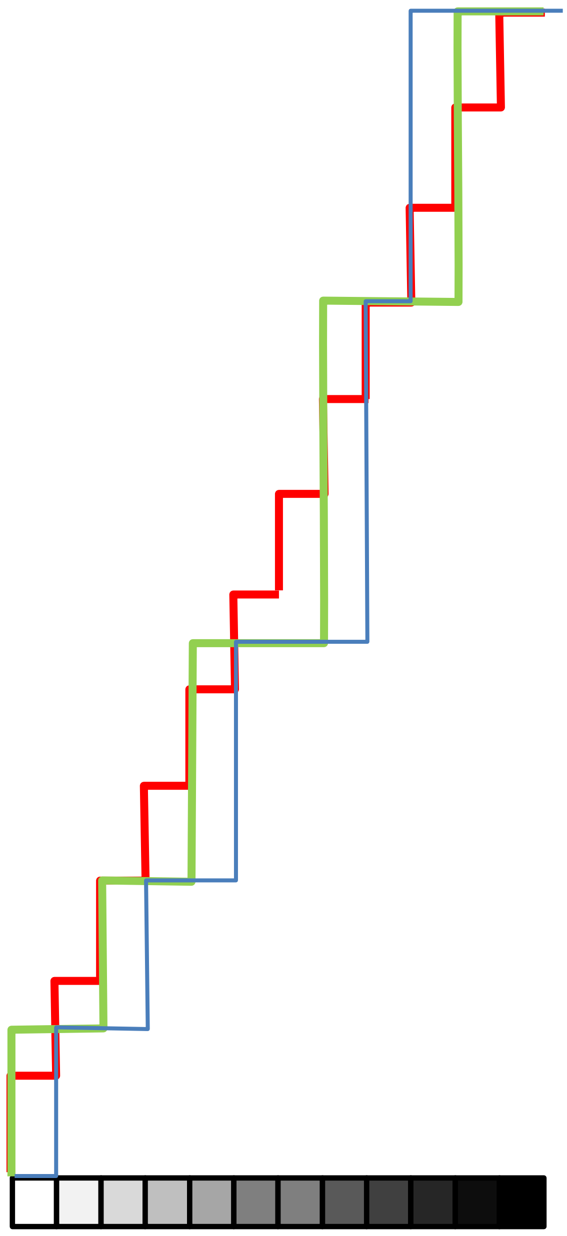
How to shift the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?

Error = 15

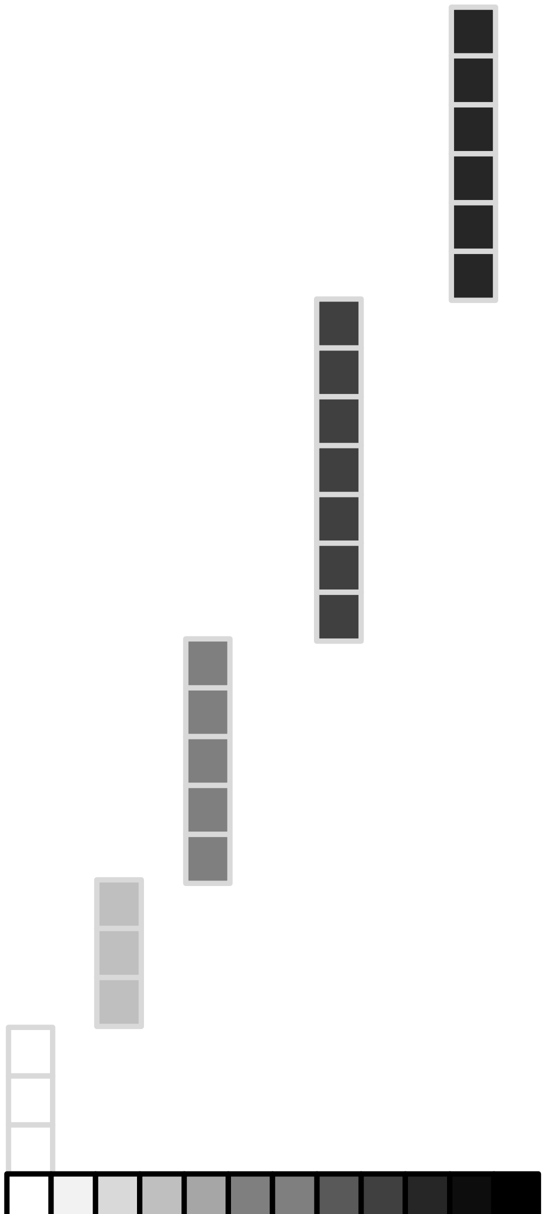


ideal accumulative function (red)
[old] real accumulative function (blue)
[new] real accumulative function (green)

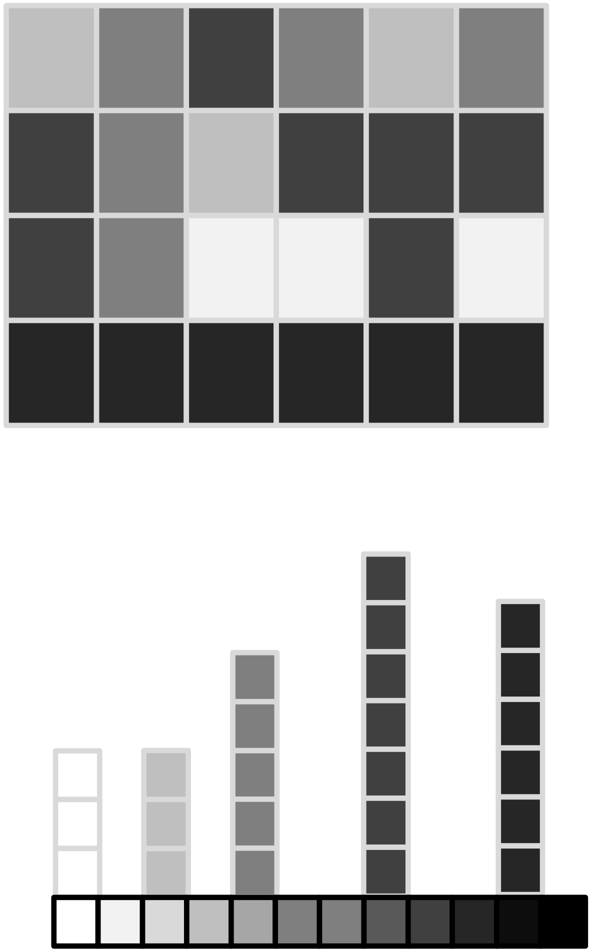
Error = 15 (green circles)



ideal accumulative function (red)
[old] real accumulative function (blue)
[new] real accumulative function (green)

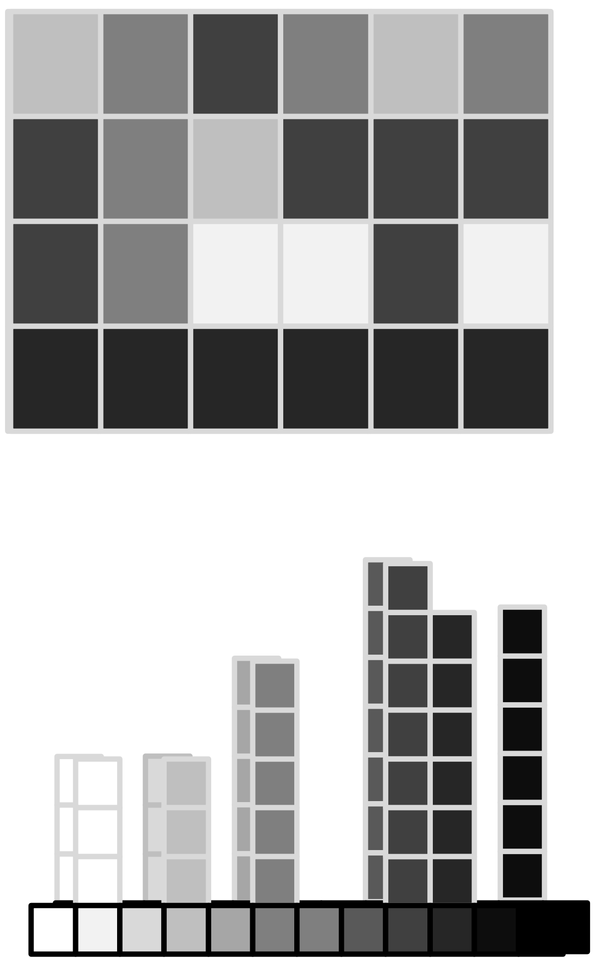


Before

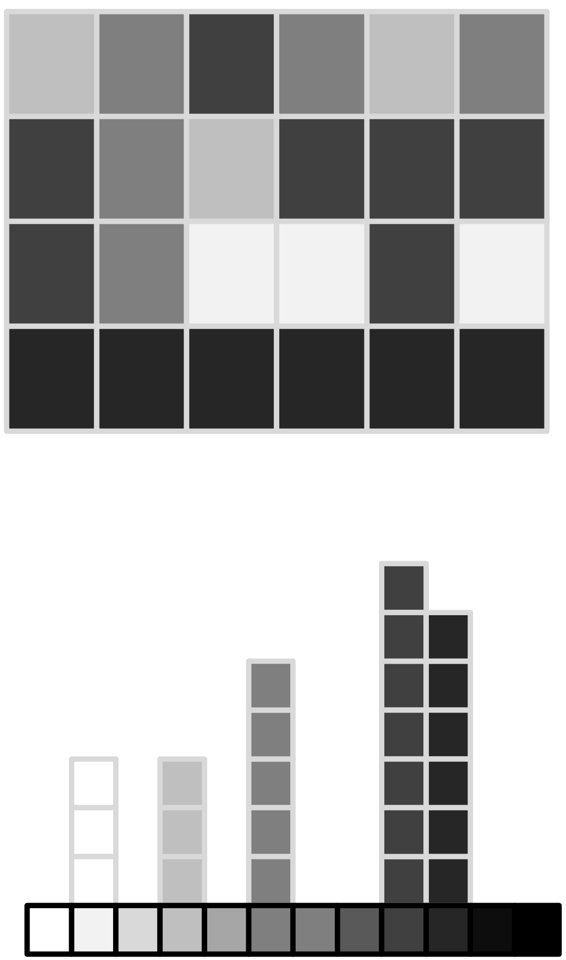


Before

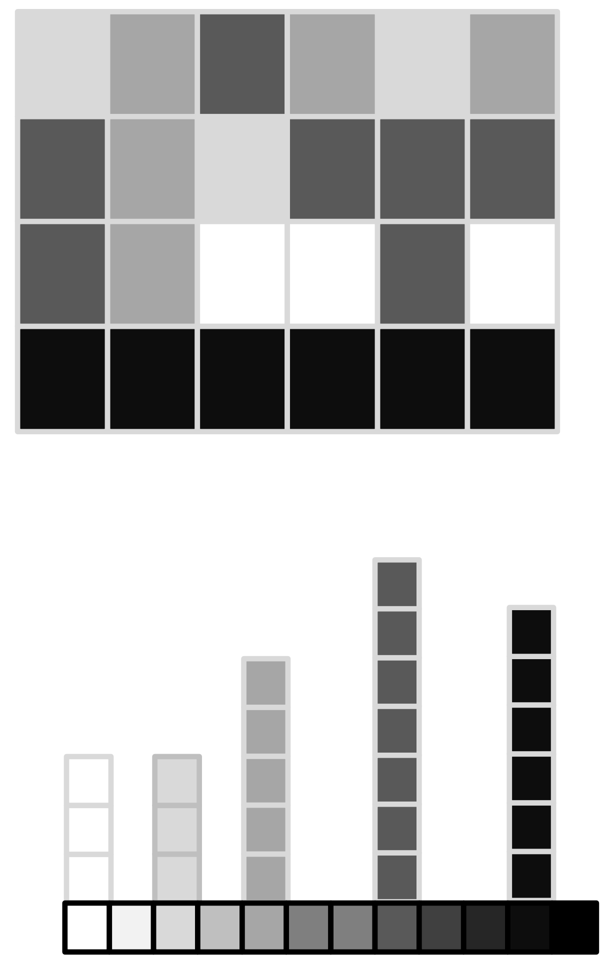
After



Before

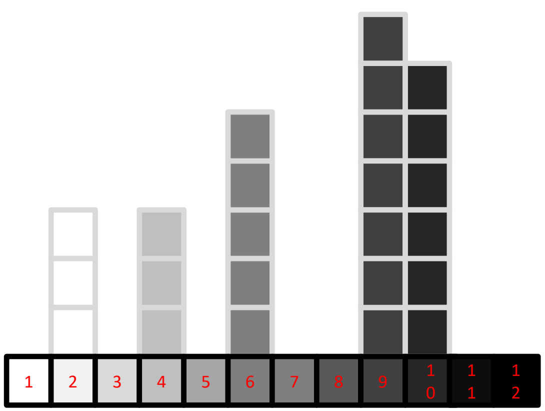


After



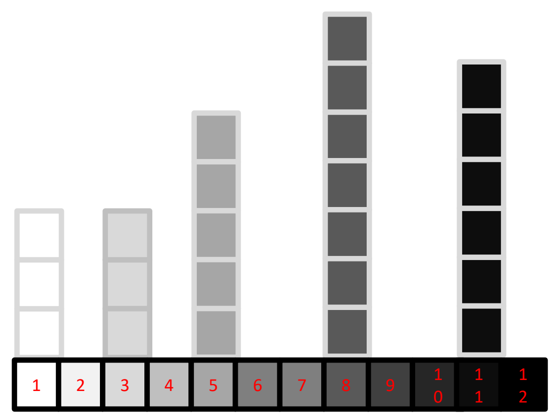
Before

4	6	9	6	4	6
9	6	4	9	9	9
9	6	2	2	9	2
10	10	10	10	10	10



After

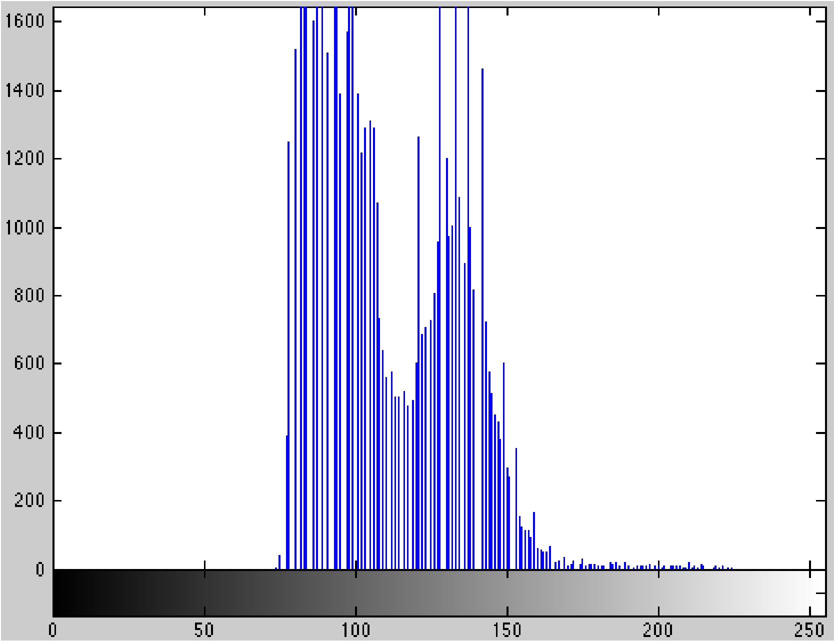
3	5	8	5	3	5
8	5	3	8	8	8
8	5	1	1	8	1
11	11	11	11	11	11



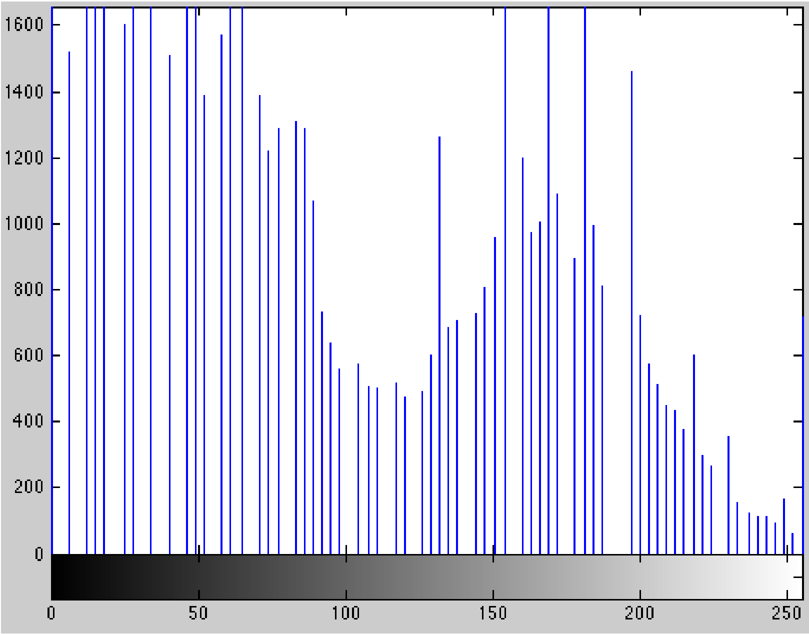
Before

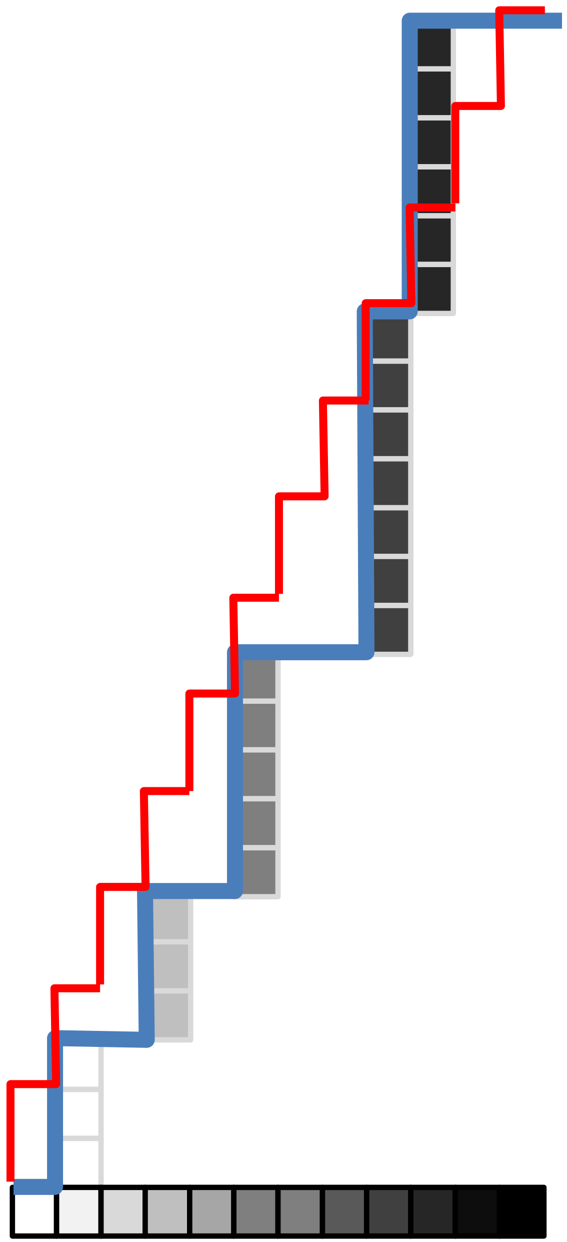
After

Before

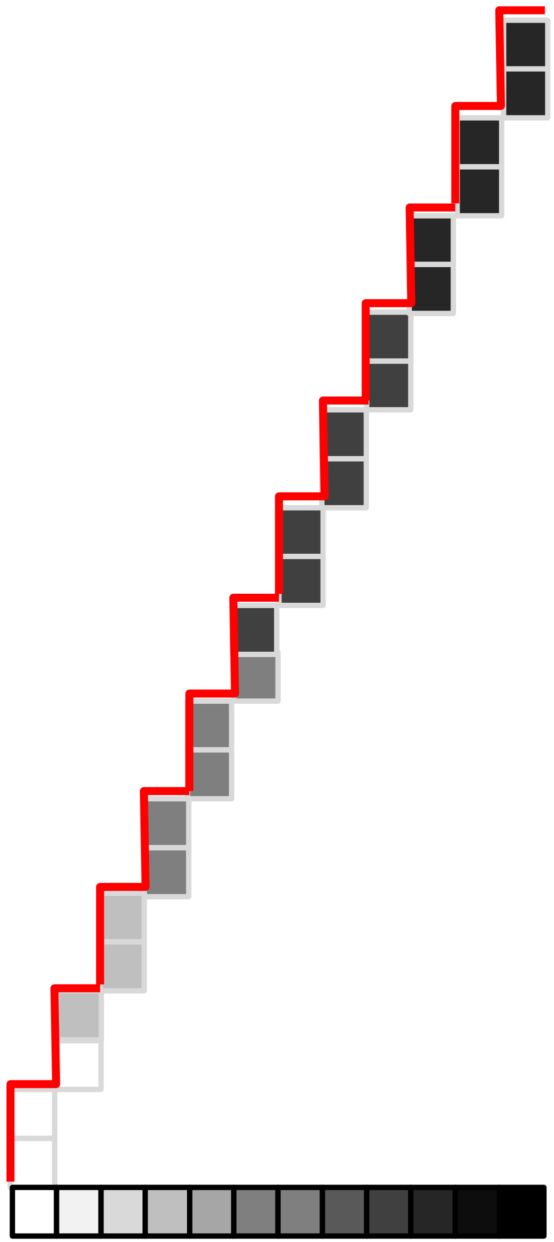


After

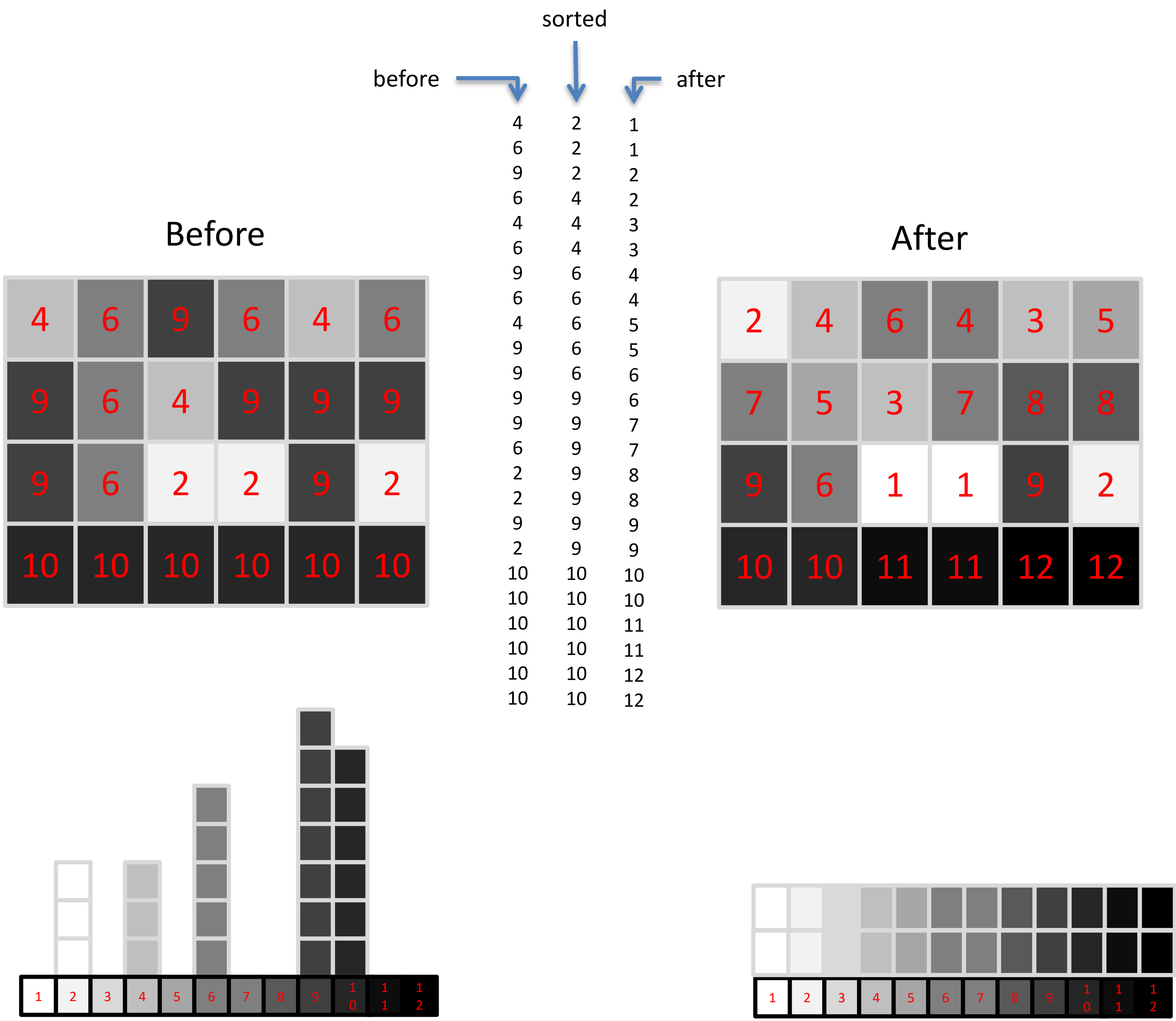




How to split the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?



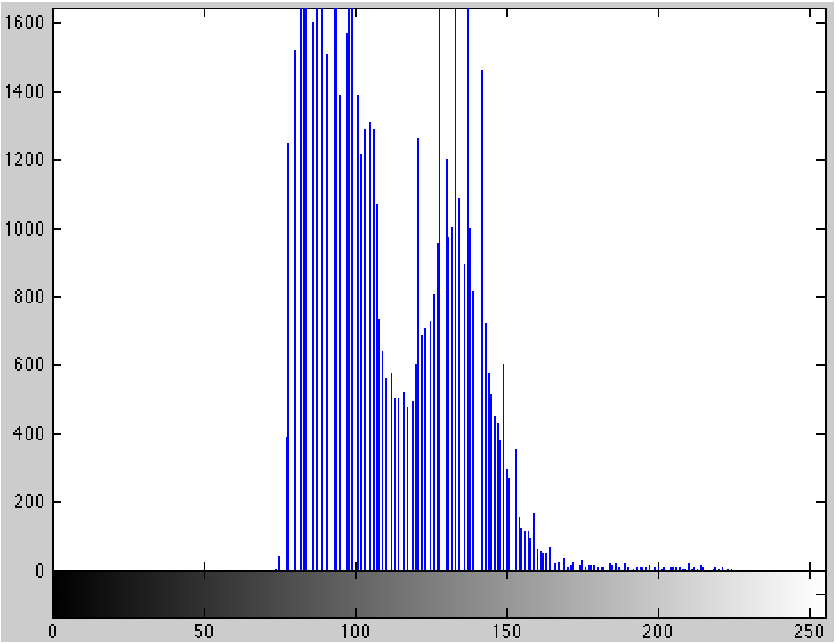
How to split the bars, so that the difference between ideal accumulative function (red) and real accumulative function (blue) is minimal?


sorted
before
after
4 6 9 6 4 6 9 6 4 9 9 9 9 6 2 2 9 2 10 10 10 10 10 10
2 2 2 4 4 4 6 6 6 6 6 9 9 9 9 9 9 9 10 10 10 10 10 10
1 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10 11 11 12 12
Before
4 6 9 6 4 6
9 6 4 9 9 9
9 6 2 2 9 2
10 10 10 10 10 10
After
2 4 6 4 3 5
7 5 3 7 8 8
9 6 1 1 9 2
10 10 11 11 12 12
1 2 3 4 5 6 7 8 9 10 11 12
1 2 3 4 5 6 7 8 9 10 11 12

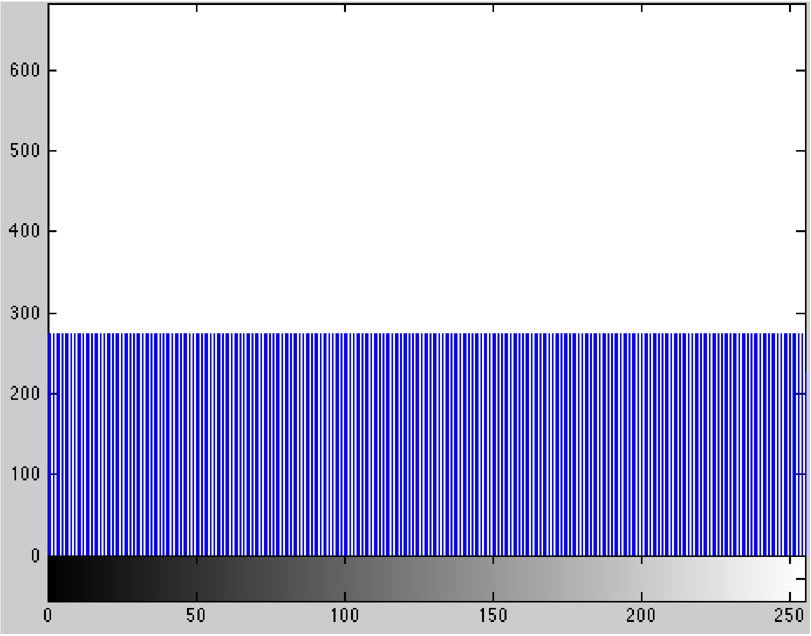
Before

After

Before



After



Method 1

Method 2