UNIVERSITÀ DI PARMA

Point Operations

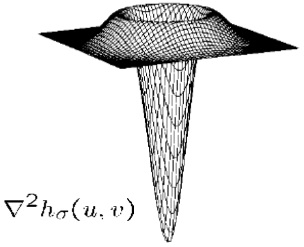
Summary

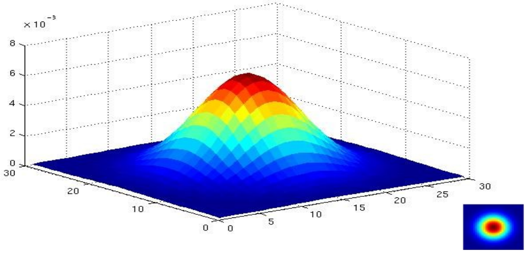
- Local operation definition
- Histogram
- Contrast/Brightness modifications
- Linear Blend
- Gamma Correction
- Histogram Equalization
  - Adaptive histogram equalization

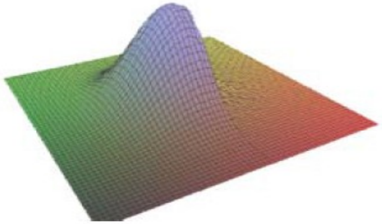
Local Operations

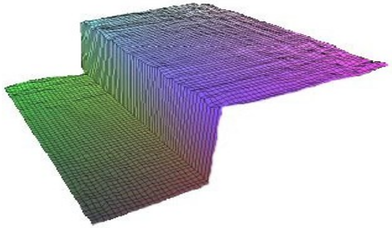
- Local Operations only depend on pixel value and neighborhood values
- We already discussed
  - Smoothing
  - Sharpening
  - Gradients
  - …

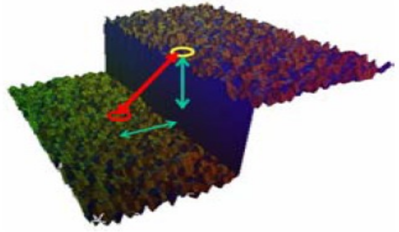
Laplacian of Gaussian
or LoG filterf











Point Operations

- Point Operations only depends on pixel value
  - No neighborhood is considered

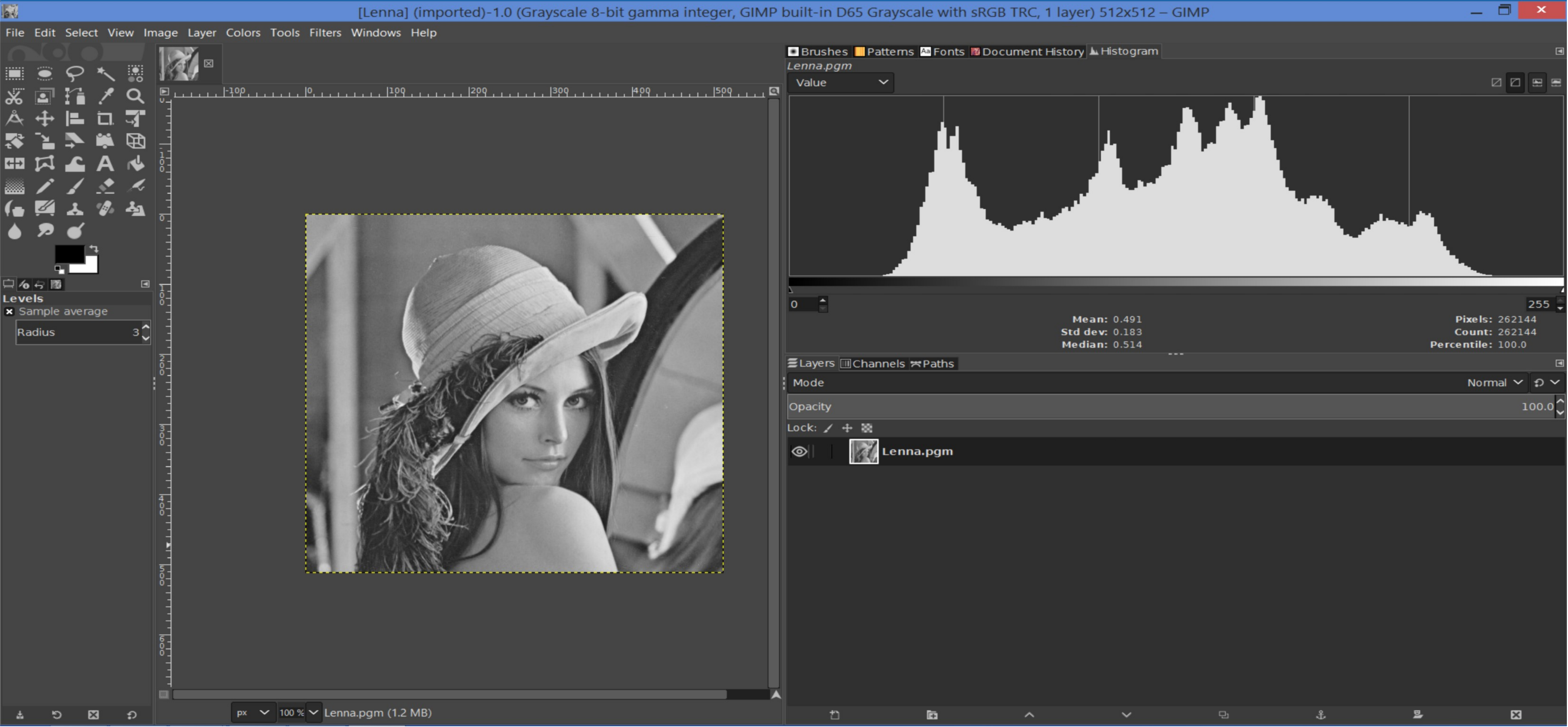
$$g(r,c) = h(f(r,c))$$

- Also known as *Point Processes*

Histogram

- Result is not an image
  - Vector of integers
  - Size of that vector is the same as possible pixel values
  - i.e. it does not depend on image content
    - Grey level image 8 bit/pixel $\longrightarrow$ 256 values
    - Grey level image 10 bit/pixel $\longrightarrow$ 1024 values
    - Color image 8 bit/pixel/channel $\longrightarrow$ 3 histograms for 256 values each
  - Each vector element "counts" the pixels of the image that have the same value of the element index
    - i.e. for a grey level image the element at index X contains the number of pixels that feature a X grey level

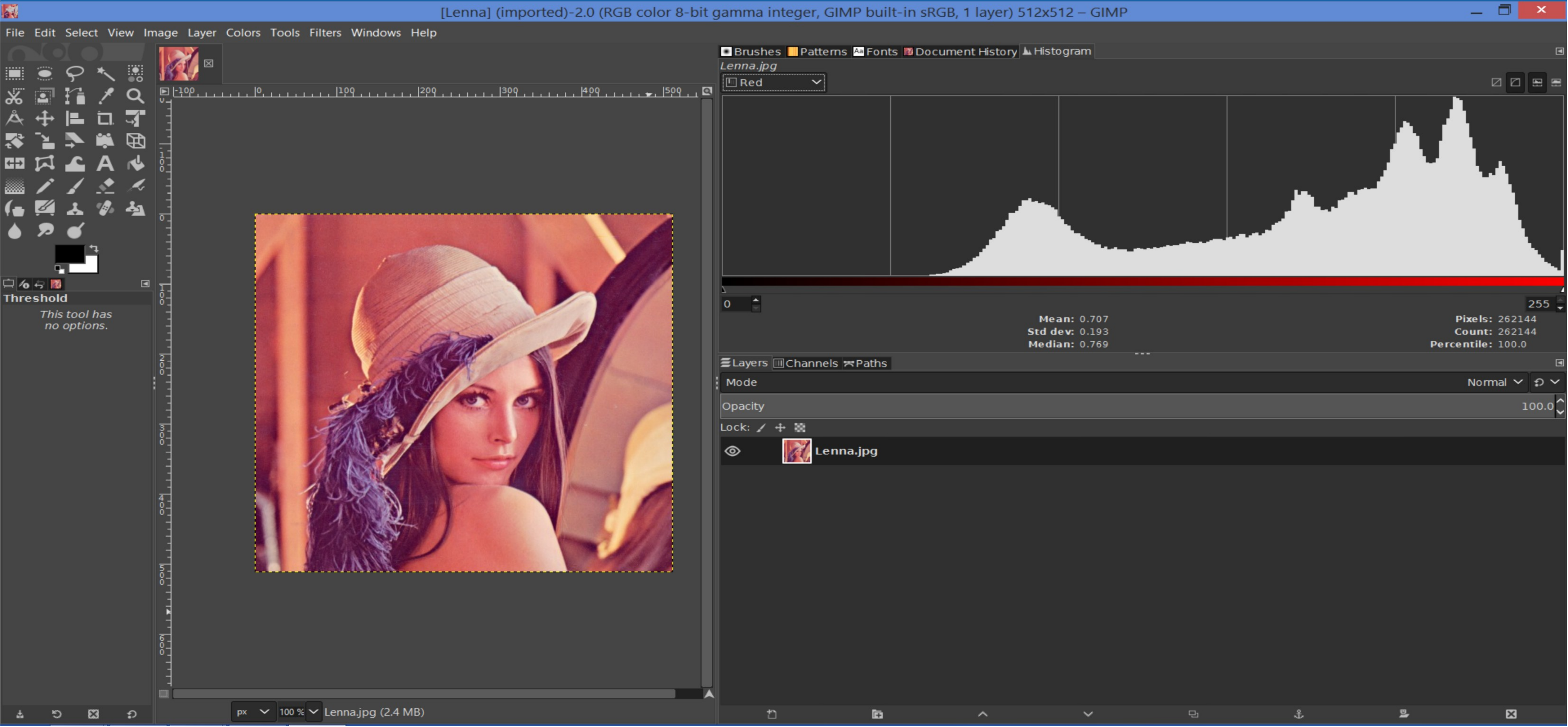
Histogram



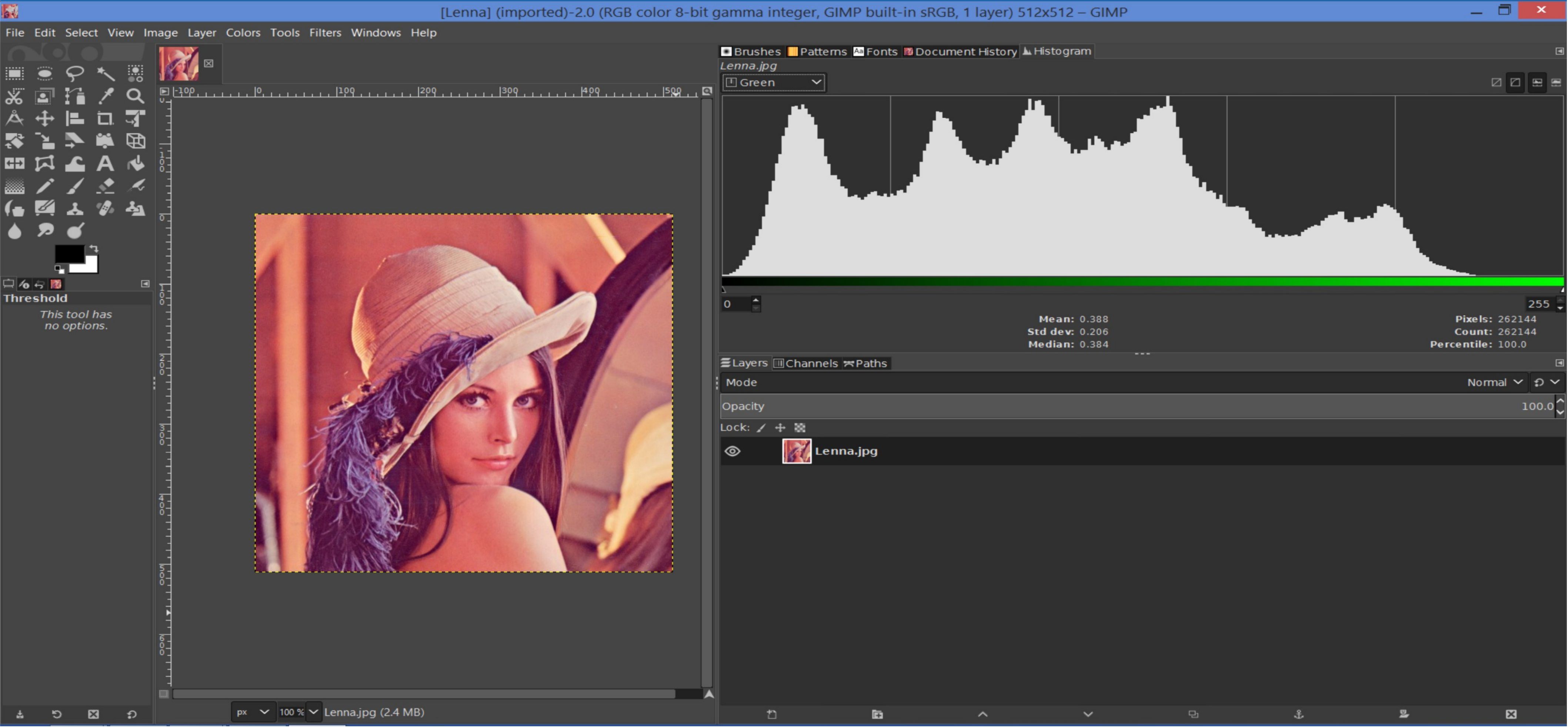
[Lenna] (imported)-1.0 (Grayscale 8-bit gamma integer, GIMP built-in D65 Grayscale with sRGB TRC, 1 layer) 512x512 – GIMP
File Edit Select View Image Layer Colors Tools Filters Windows Help
Brushes Patterns Fonts Document History Histogram
Lenna.pgm
Value
0
255
Mean: 0.491
Std dev: 0.183
Median: 0.514
Pixels: 262144
Count: 262144
Percentile: 100.0
Layers Channels Paths
Mode Normal
Opacity 100.0
Lock:
Lenna.pgm
Levels
Sample average
Radius 3
px 100 % Lenna.pgm (1.2 MB)

Histogram

UNIVERSITÀ DI PARMA

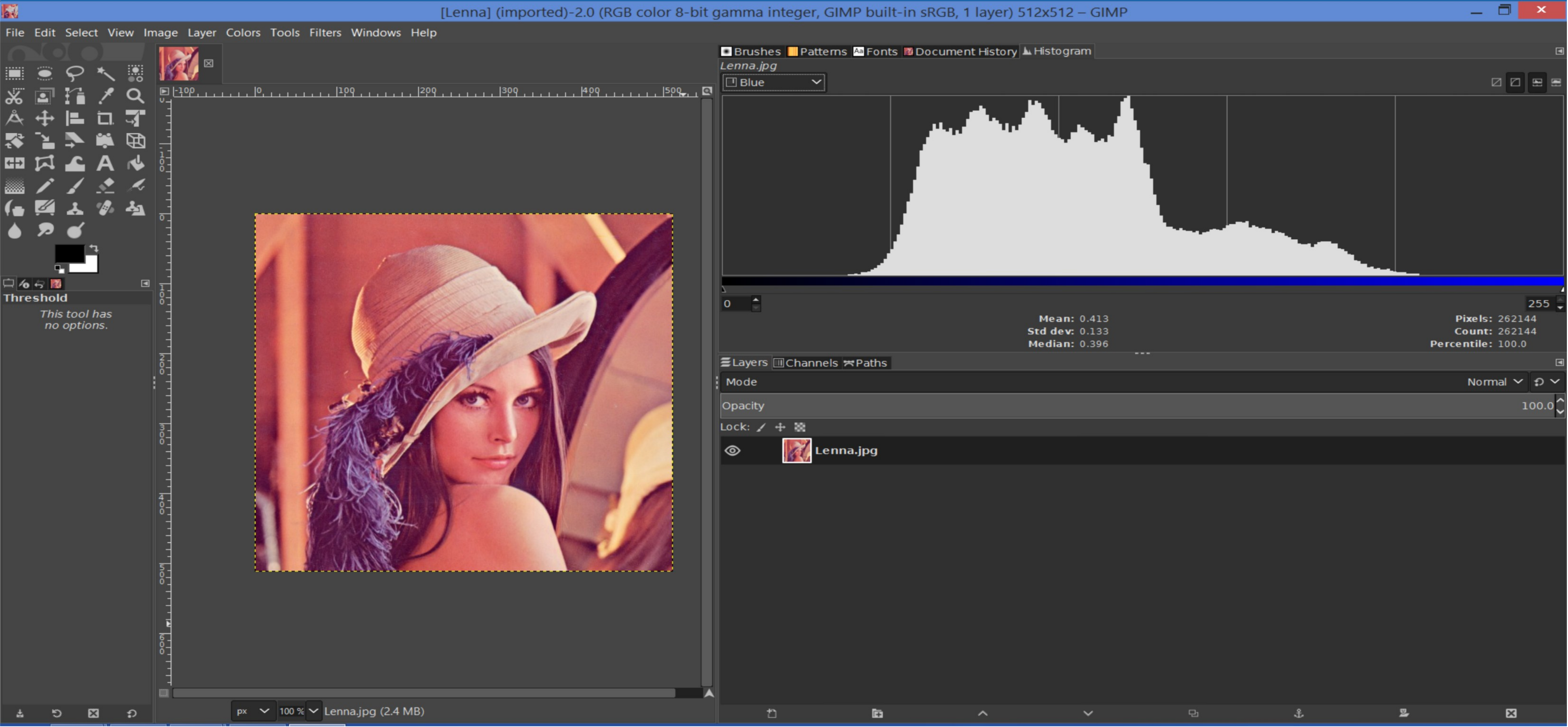


Histogram

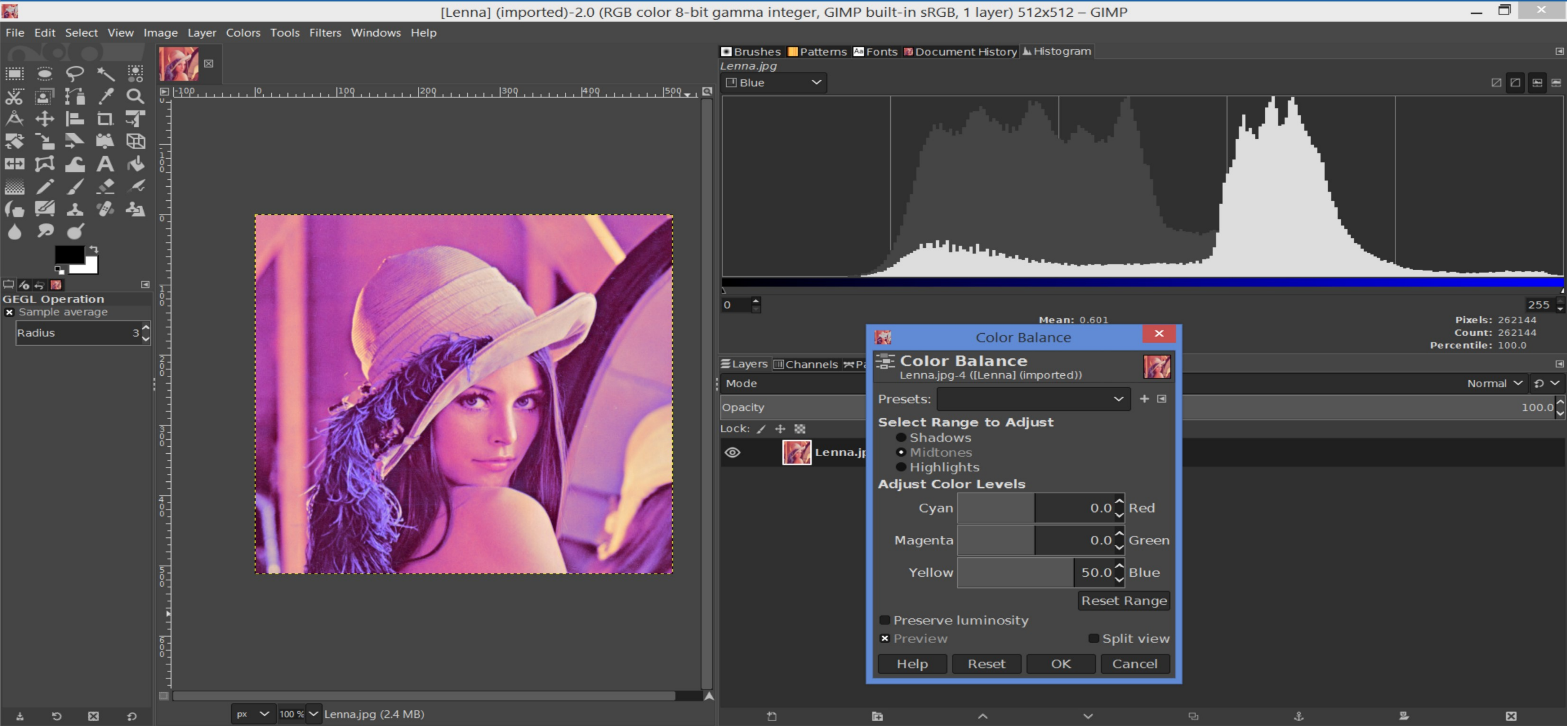


[Lenna] (imported)-2.0 (RGB color 8-bit gamma integer, GIMP built-in sRGB, 1 layer) 512x512 – GIMP
File Edit Select View Image Layer Colors Tools Filters Windows Help
Brushes Patterns Fonts Document History Histogram
Lenna.jpg
Green
0
255
Mean: 0.388
Std dev: 0.206
Median: 0.384
Pixels: 262144
Count: 262144
Percentile: 100.0
Threshold
This tool has no options.
Layers Channels Paths
Mode
Normal
Opacity
100.0
Lock:
Lenna.jpg
Lenna.jpg (2.4 MB)

Histogram


[Lenna] (imported)-2.0 (RGB color 8-bit gamma integer, GIMP built-in sRGB, 1 layer) 512x512 – GIMP
File Edit Select View Image Layer Colors Tools Filters Windows Help
Brushes Patterns Fonts Document History Histogram
Lenna.jpg
Blue
0
255
Mean: 0.413
Std dev: 0.133
Median: 0.396
Pixels: 262144
Count: 262144
Percentile: 100.0
Threshold
This tool has no options.
Layers Channels Paths
Mode
Normal
Opacity
100.0
Lock:
Lenna.jpg
Lenna.jpg (2.4 MB)

Histogram



[Lenna] (imported)-2.0 (RGB color 8-bit gamma integer, GIMP built-in sRGB, 1 layer) 512x512 – GIMP
File Edit Select View Image Layer Colors Tools Filters Windows Help
GEGL Operation
Sample average
Radius 3
Brushes Patterns Fonts Document History Histogram
Lenna.jpg
Blue
0
255
Mean: 0.601
Pixels: 262144
Count: 262144
Percentile: 100.0
Layers Channels
Mode
Normal
Opacity
100.0
Lock:
Lenna.jp
Color Balance
Color Balance
Lenna.jpg-4 ([Lenna] (imported))
Presets:
Select Range to Adjust
Shadows
Midtones
Highlights
Adjust Color Levels
Cyan 0.0 Red
Magenta 0.0 Green
Yellow 50.0 Blue
Reset Range
Preserve luminosity
Preview
Split view
Help Reset OK Cancel
px 100 % Lenna.jpg (2.4 MB)

Contrast & Brightness

- Contrast and/or Brightness can be adjusted as:

$$g(r,c) = a \cdot f(r,c) + b$$

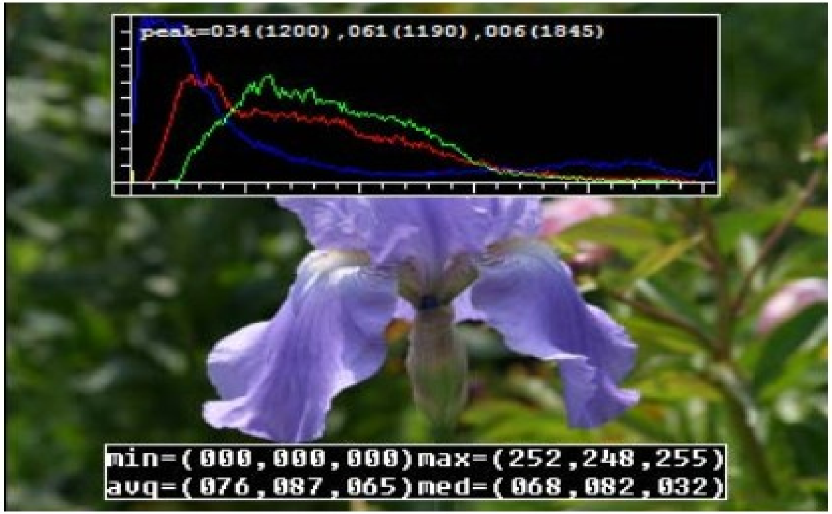
- a is the **gain**
- b is the **bias**

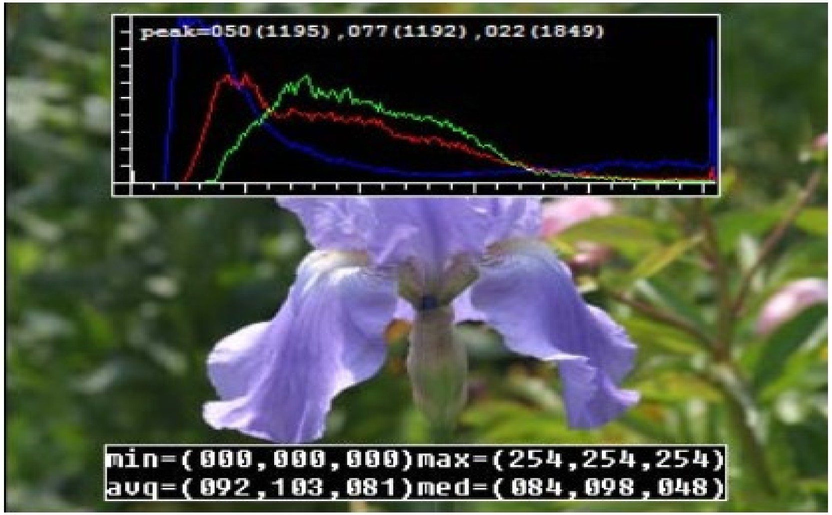
  or

- a $\longrightarrow$ contrast control
- b $\longrightarrow$ brightness control

Brightness/Gain

$$g(r,c)=f(r,c)+b \quad with \quad b=16$$

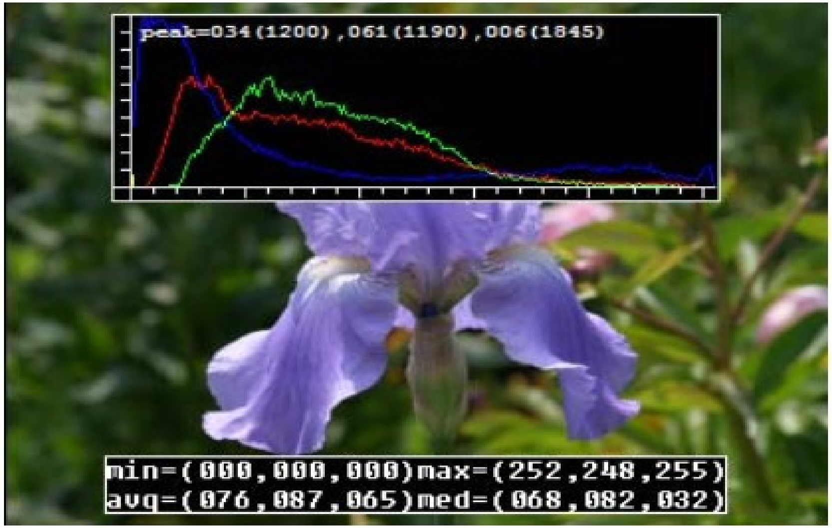


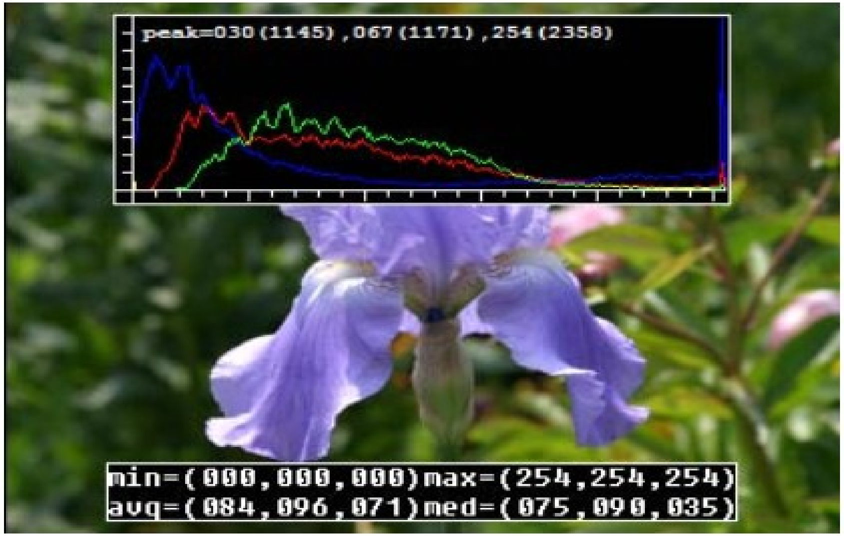


Source:Szeliski

Contrast

$$g(r,c)=a\cdot f(r,c) \quad with \quad a=1.1$$





Source:Szeliski

Contrast stretching

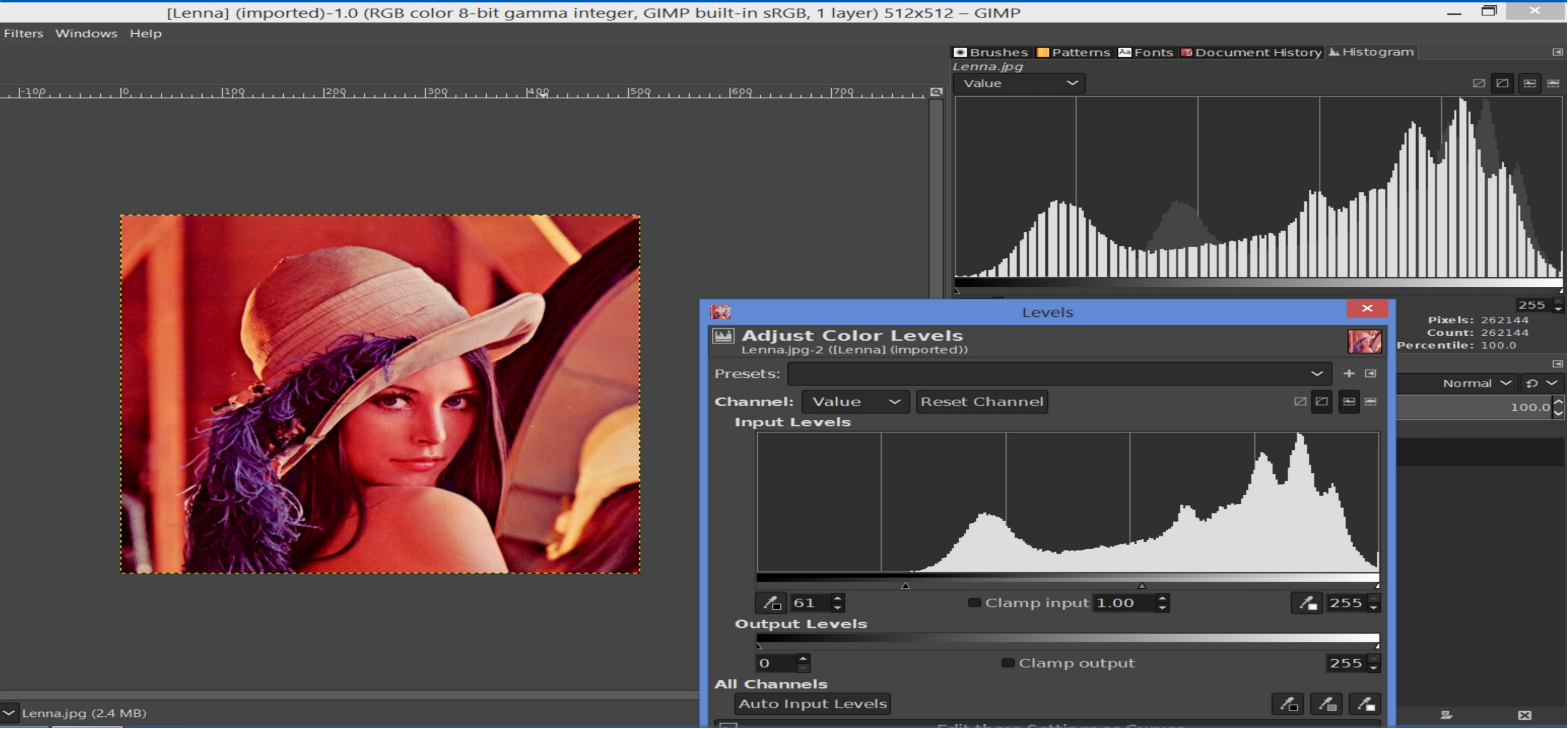
- Specific values of a & b can lead histogram to span to fill the range of potential values

$$g(r,c)=\frac{255\cdot(f(r,c)-\min(f))}{\max(f)-\min(f)}¿$$

- When $f(r,c) == \max(f) \longrightarrow g(r,c) = 255$
- Where $f(r,c) == \min(f) \longrightarrow g(r,c) = 0$

Contrast stretching

UNIVERSITÀ DI PARMA



Contrast & Gain

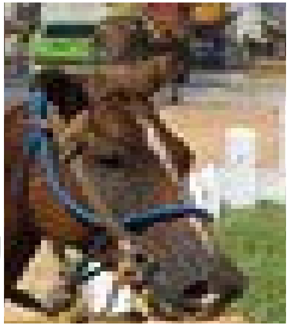
- **Contrast** and **gain**

$$g(r,c)=a(r,c)\cdot f(r,c)+b(r,c)$$

- a **spatial contrast** control
- b **spatial brightness** control

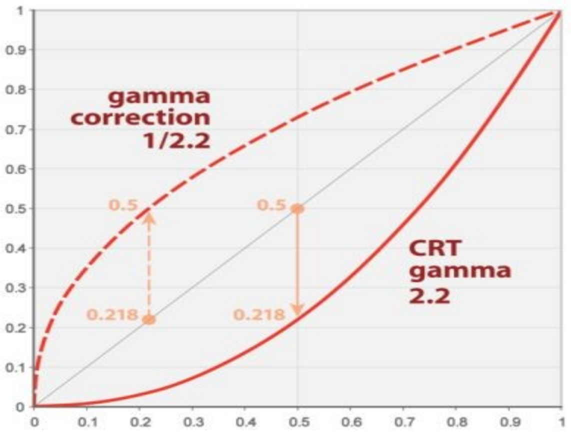
Linear Blend

$$g(r,c)=(1-\alpha)f_0(r,c)+\alpha f_1(r,c).$$

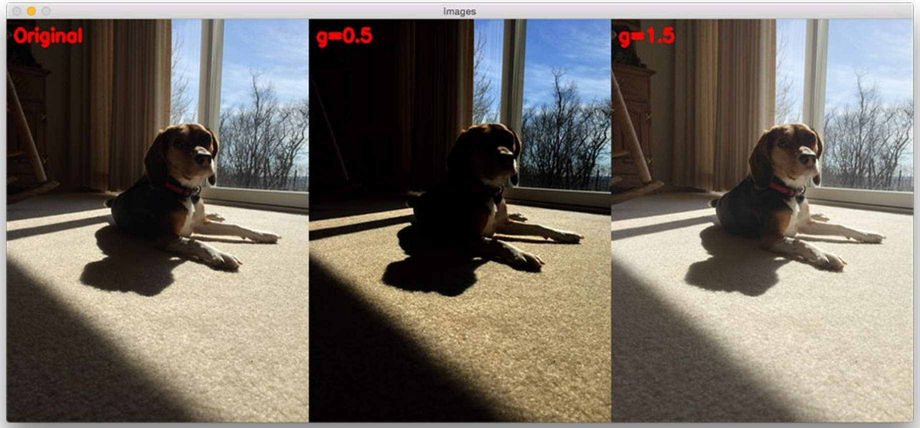
F( , ) = 

Gamma Correction

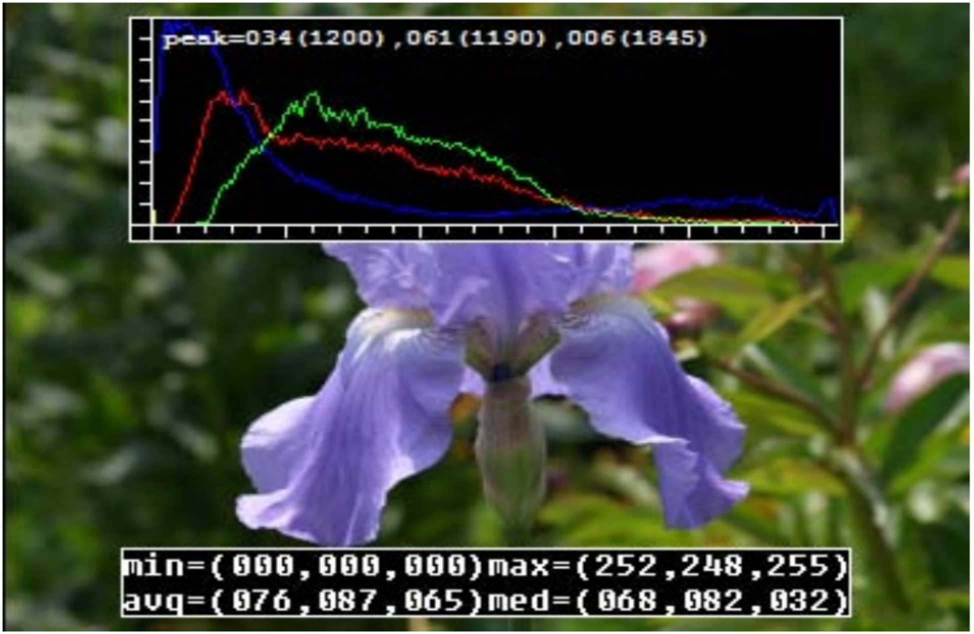
- Rimuove eventuali non linearità tra segnale e risultato
- Sensori telecamere != occhio umano
- Applicato a volte per compensare non linearità dei monitor

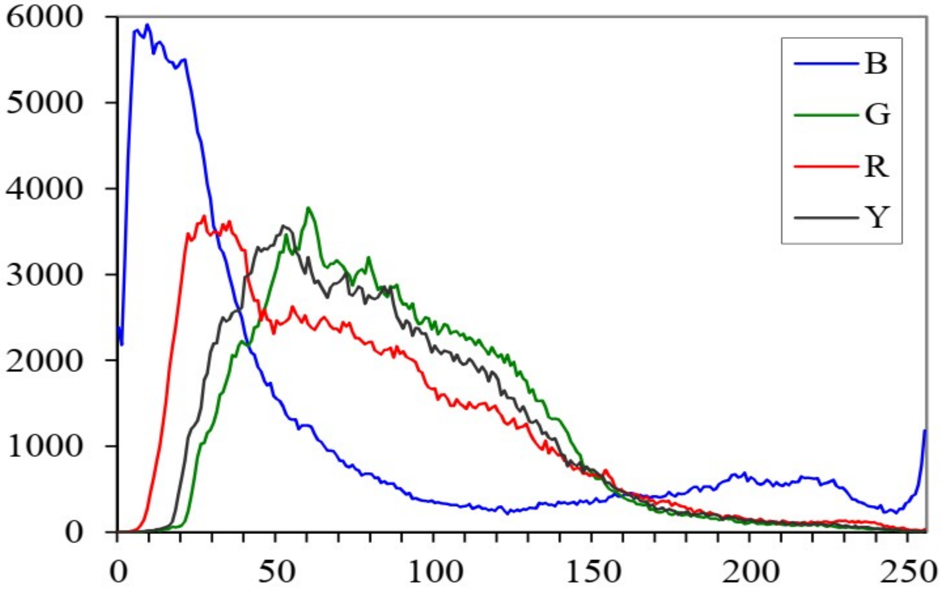


$$g(r,c)=f(r,c)^{\frac{1}{\lambda}}$$



Histogram Equalization





In this image we have more “dark” values wrt “bright” ones

Source:Szeliski

Histogram Equalization

UNIVERSITÀ DI PARMA

- The underlying idea of histogram equalization is to balance values to "expand" intensity values to fit the available range
- The result should have the same amount of low/medium/high intensity values

Histogram Equalization

- The histogram can be considered as a *Probability Density Function* for pixel values
- When we integrate (aka sum) the histogram we obtain the *Cumulative Distribute Function*
- This function encodes the *percentile score* for pixel values

Histogram Equalization

UNIVERSITÀ DI PARMA

- The *Cumulative Distribute Function* can be obtained integrating the histogram *h(I)*

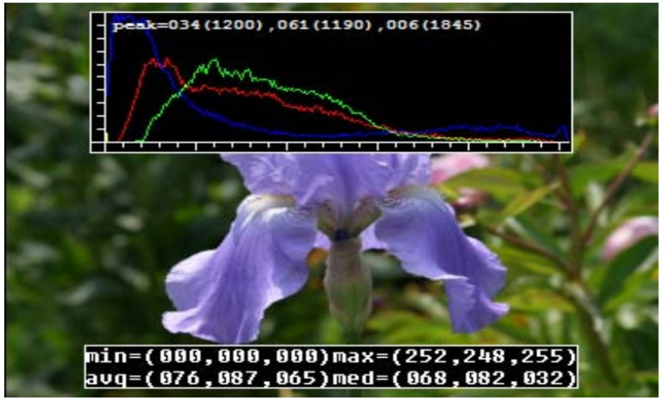
$$CI(j) = \sum_{i=0}^{j} h(i) = CI(j-1) + h(j)$$

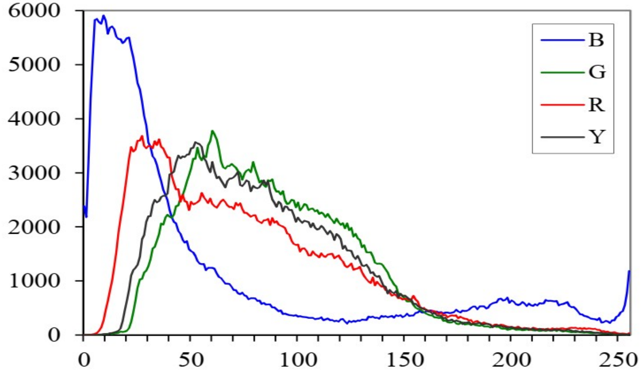
$$CDF(j) = \frac{1}{N} \sum_{i=0}^{j} h(i) = CDF(j-1) + \frac{1}{N} h(j)$$

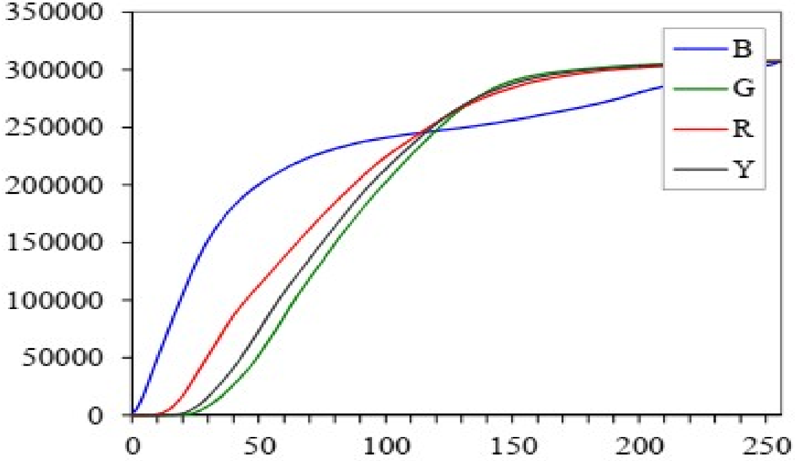
- Where N is the number of pixels

Source:Szeliski

Histogram Equalization







$$CI(j)=\sum_{i=0}^{i\leq j} h(i)=C(j-1)+h(j)$$

Source:Szeliski

Histogram Equalization

- The equalization is obtained as

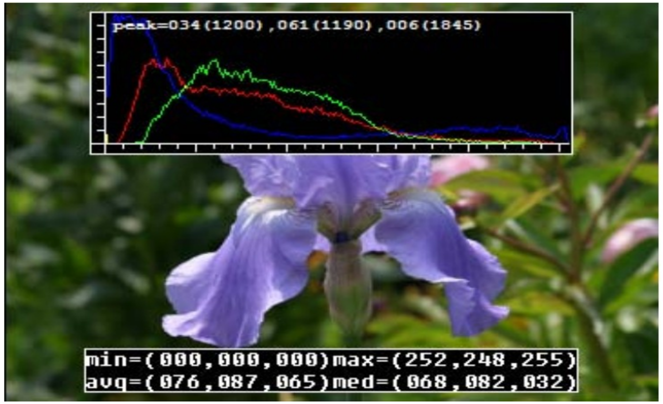
$$equalized \quad f(j) = CDF(j) * maxrange$$

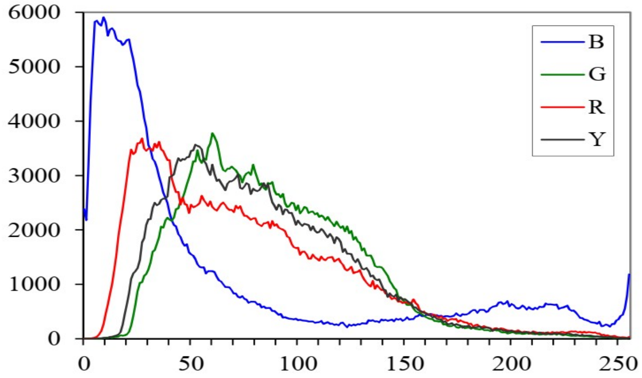
- Where maxrange is the maximum value for a pixel (i.e. for 8 bit $\longrightarrow$ 255) and not the current maximum value
- Equalization can produce innatural effects (flat image) and a partial approach can be used

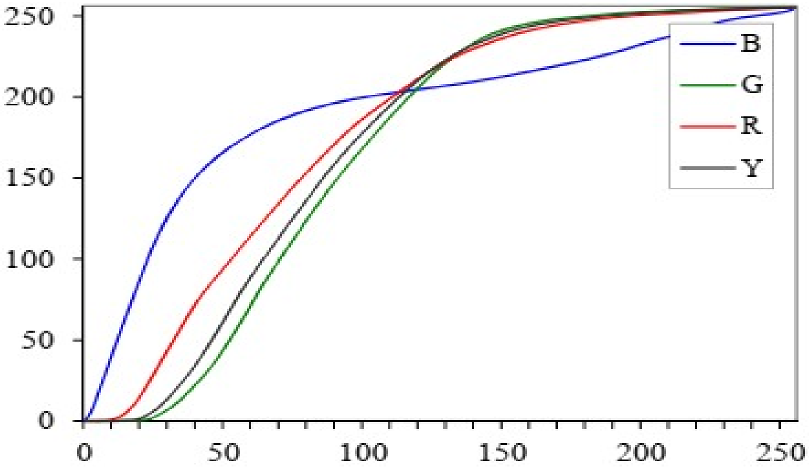
$$equalized \quad f(j) = ((1-\alpha) \cdot CDF(j) + \alpha \cdot j) * maxrange$$

Source:Szeliski

Histogram Equalization





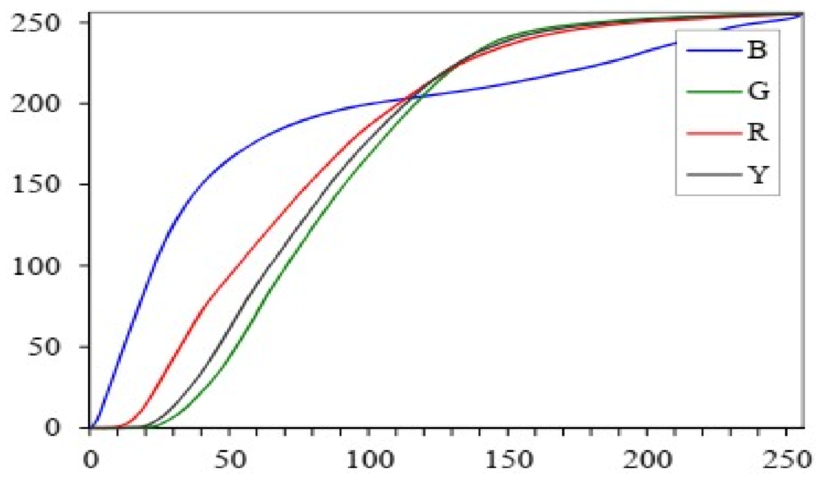


$CFD(j) \cdot maxvalue$
¿
$CDF(j) \cdot 255$

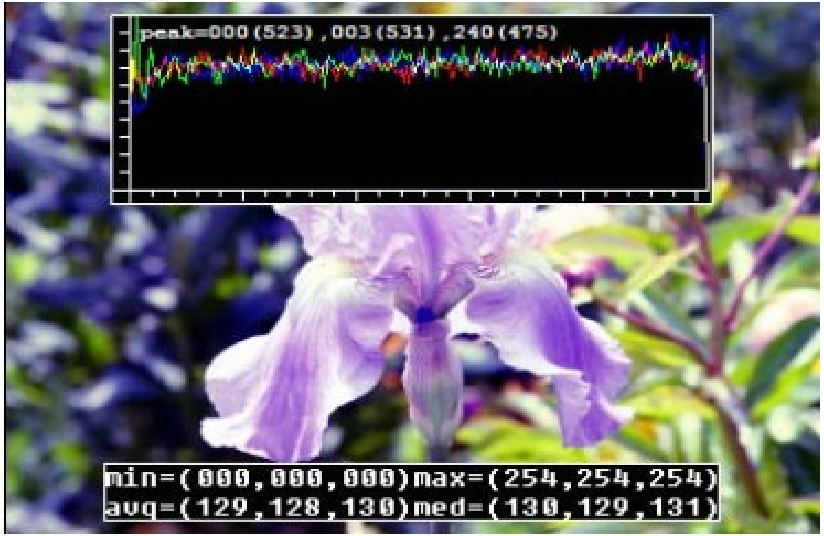
Source:Szeliski

Histogram Equalization

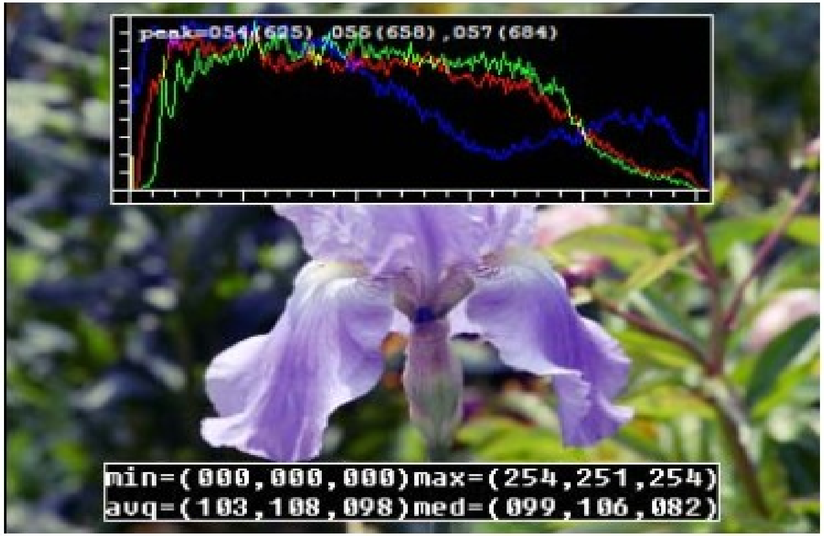
- The *CDF(j)*maxvalue* can be used as a **lookup table**
  - Vector index encodes the current pixel value
  - Vector value indicates the equalized value



Histogram Equalization



Full equalization



Partial equalization

Source:Szeliski

Locally adaptive histogram equalization

- Histogram equalization does not work well for all regions of an image
- Just think about images that features region with very different luminance values

Source:Szeliski

Locally adaptive histogram equalization

- The image is subdivided in M x M sized blocks
- For each block a separate histogram is computed

Source:Szeliski

Locally adaptive histogram equalization

- Locally the result is better
- Anyway discontinuities are produces

Source:Szeliski

Locally adaptive histogram equalization

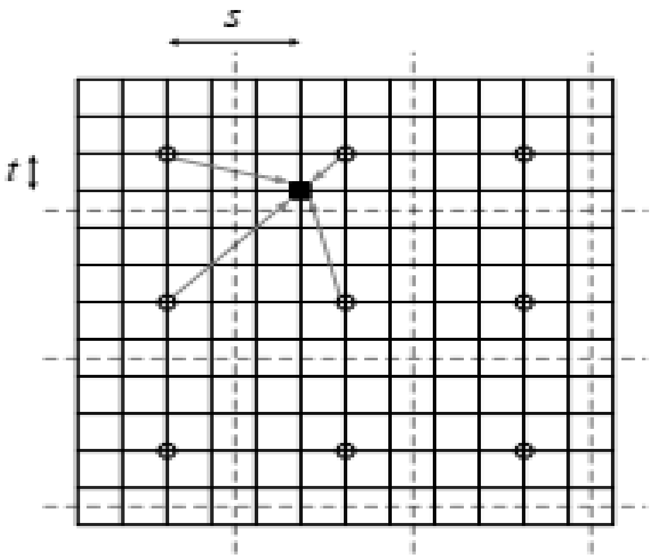
UNIVERSITÀ DI PARMA

- Different approaches to avoid this effect
- Moving window
  - Recompute histogram using a M x M window centered on each pixel
  - Highly effective
  - Highly inefficient $\longrightarrow M^2$ operations per pixel!
- Bilinear interpolation
  - Same blocks setup as before
  - A weight function is used that takes in account also adjacent blocks

Locally Adaptive Histogram Equalization

UNIVERSITÀ DI PARMA

- Is no strictly “locally”
- For each pixel the distance from the center of each block is used as weight factor



(a)

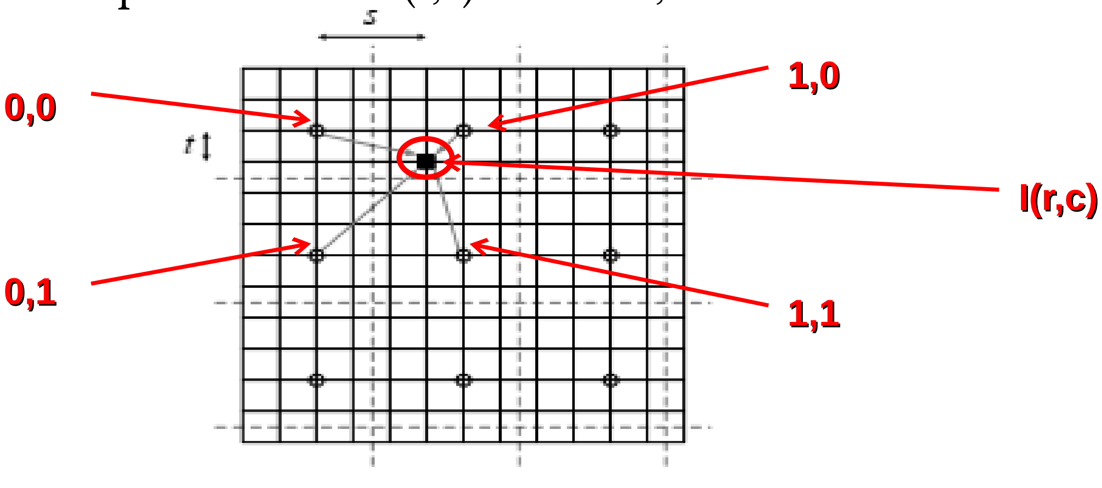
Note: pixels are located on grid intersections

Source:Szeliski

Locally Adaptive Histogram Equalization

- I(r,c) is the intensity value of pixel r,c,
  - $f_{00}(I(r,c))$ is the equalization of I(r,c) for the 0,0 block
  - $f_{01}(I(r,c))$ is the equalization of I(r,c) for the 0,1 block
  - $f_{10}(I(r,c))$ is the equalization of I(r,c) for the 1,0 block
  - $f_{11}(I(r,c))$ is the equalization of I(r,c) for the 1,1 block

Note: $f_{xy}()$ are the histogram lookup tables!



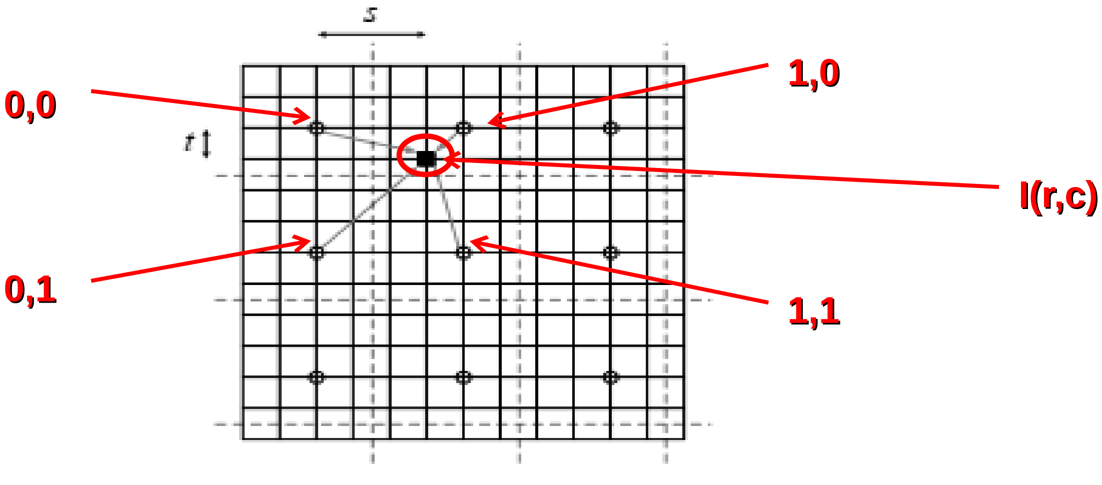
(a)

Source:Szeliski

Locally Adaptive Histogram Equalization

- Being s and t the normalized distances of pixel (r,c) from the center of 0,0 block

$$f_{s,t}=(1-s)(1-t)f_{00}(I)+s(1-t)f_{10}(I)+(1-s)tf_{01}(I)+stf_{11}(I)$$



(a)

Source:Szeliski

Locally Adaptive Histogram Equalization

Source:Szeliski