EECE/CS 4353 Image Processing

Lecture Notes: Image Histograms

Richard Alan Peters II

Department of Electrical Engineering and Computer Science

Fall Semester 2016

Image Histograms

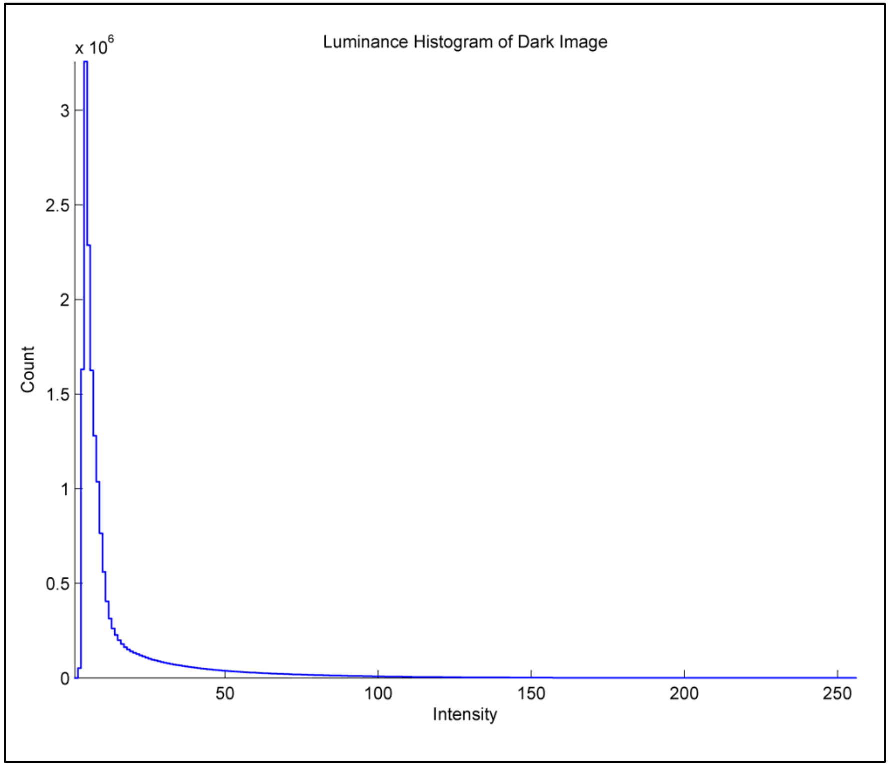
The histogram of an image is a tally of the number of pixels at each intensity level or color. The shape of the histogram is related to the ranges and groupings of intensity values in the image.

In the following monochrome examples notice how the peaks of in the histogram correspond to concentrations of intensities in the image globally.

In the color examples the primary that has the largest value at any intensity dominates the image.

Monochrome Intensity Distributions

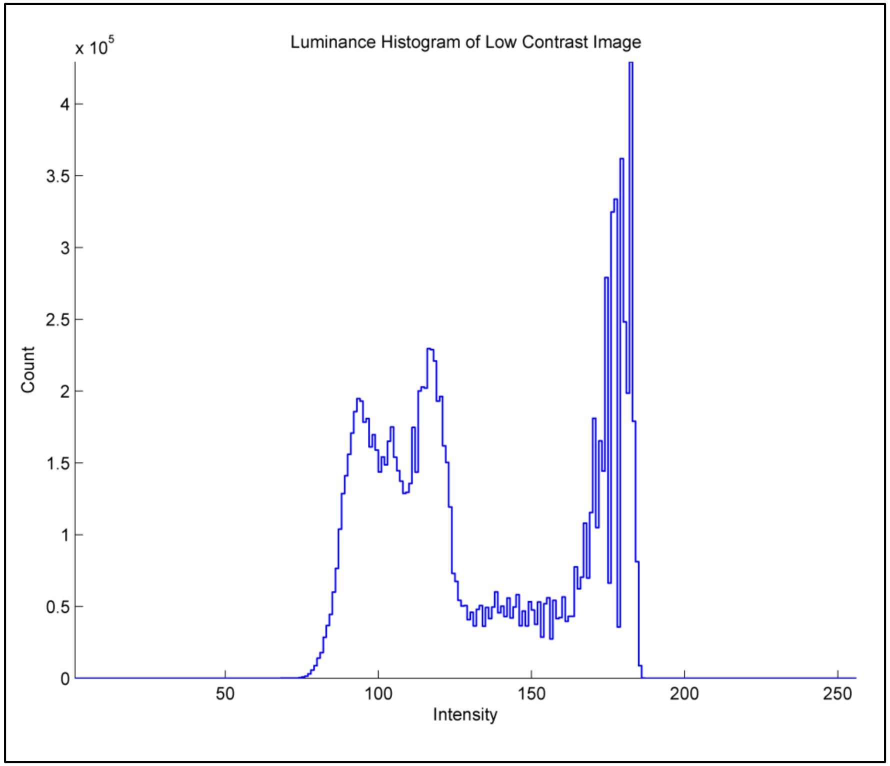
This image is a small, monochrome version of a huge color mosaic made by the ESO[1]. It contains both celestial hemispheres; it is what you would see in 360° from empty space in the plane of the galaxy above or below the earth.



[1]The Euopean Southern Observatory in the Atacama desert of Chile, http://www.eso.org/public/usa/images/eso0932a/

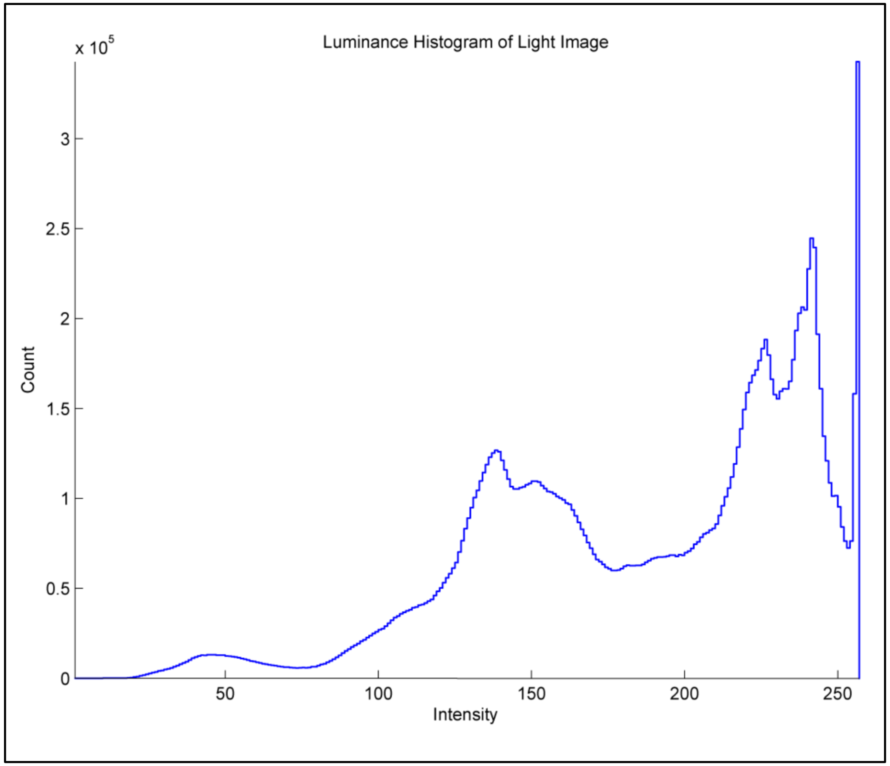
Monochrome Intensity Distributions

This picture, taken in the morning fog, displays low contrast – a narrow range of intensities – with energy at the extremes.



Photographer unknown, downloaded from [http://hqwallbase.com/21961-trees-fog-wallpaper-[2]/](http://hqwallbase.com/21961-trees-fog-wallpaper-[2]/)

Monochrome Intensity Distributions



Castner glacier in the Delta mountains, Alaska. Monochrome extracted from original color image. Note how the peaks in the histogram correspond to regions in the image.

Photographer unknown, downloaded from https://contest.thesca.org/snow2012/zig-zags-snow

Color Intensity Distributions

Castle Rock, Sedona, Arizona. There is one histogram for each of red, green, and blue. The red rock's color is in the midrange of intensities while the greenery is darker. Blue peaks correspond to the haze on the mountainside (dark) and the sky (bright).

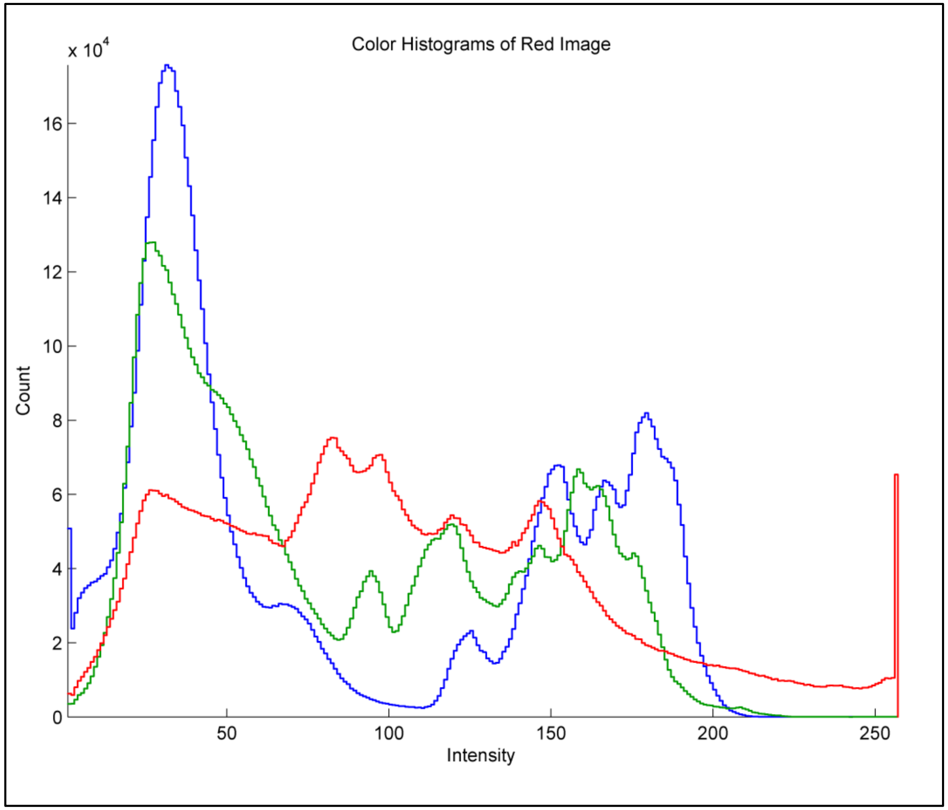
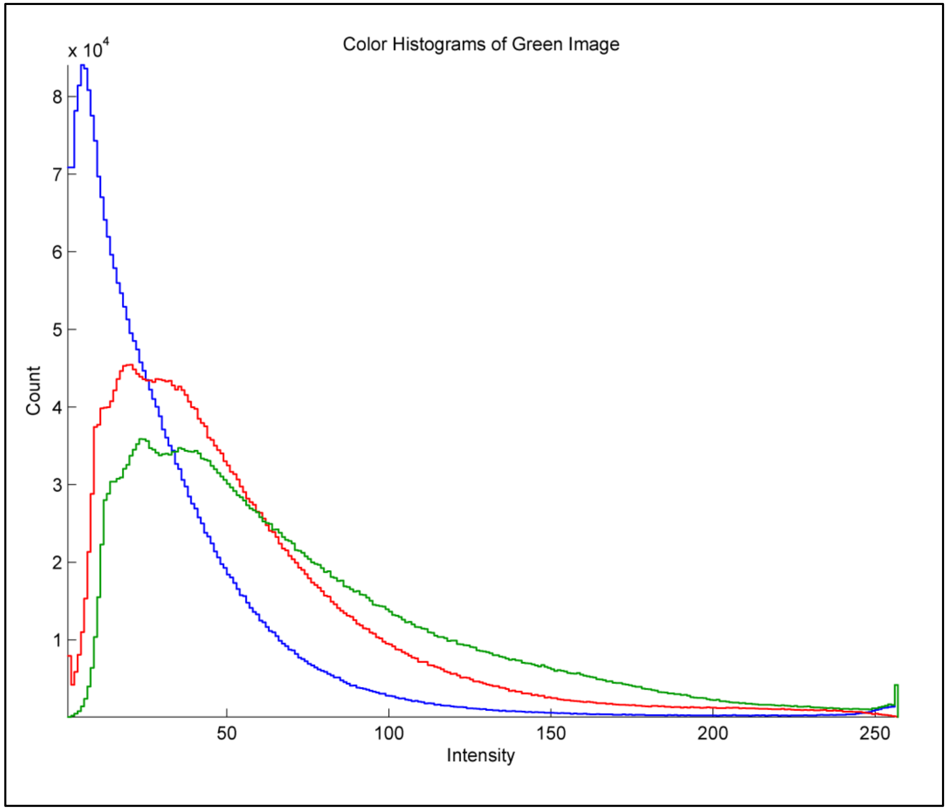


Photo by Edward Chavez, http://www.zensoulstyle.com

Color Intensity Distributions

Unidentified place in a photo from the website below. Notice that the intensity of green dominates the others over much of the range. Red dominant corresponds to yellow-green regions. Blue dominates in the shadows.



Photographer Unknown, downloaded from http://forum.baboo.com.br/index.php?/gallery/image/20033-floresta-80/

Color Intensity Distributions

Blue Poison Dart Frog (Dendrobates azureus) in the Frankfurt Zoo, Germany. Dominant colors in increasing intensity: brown, blue, tan brown, blue.

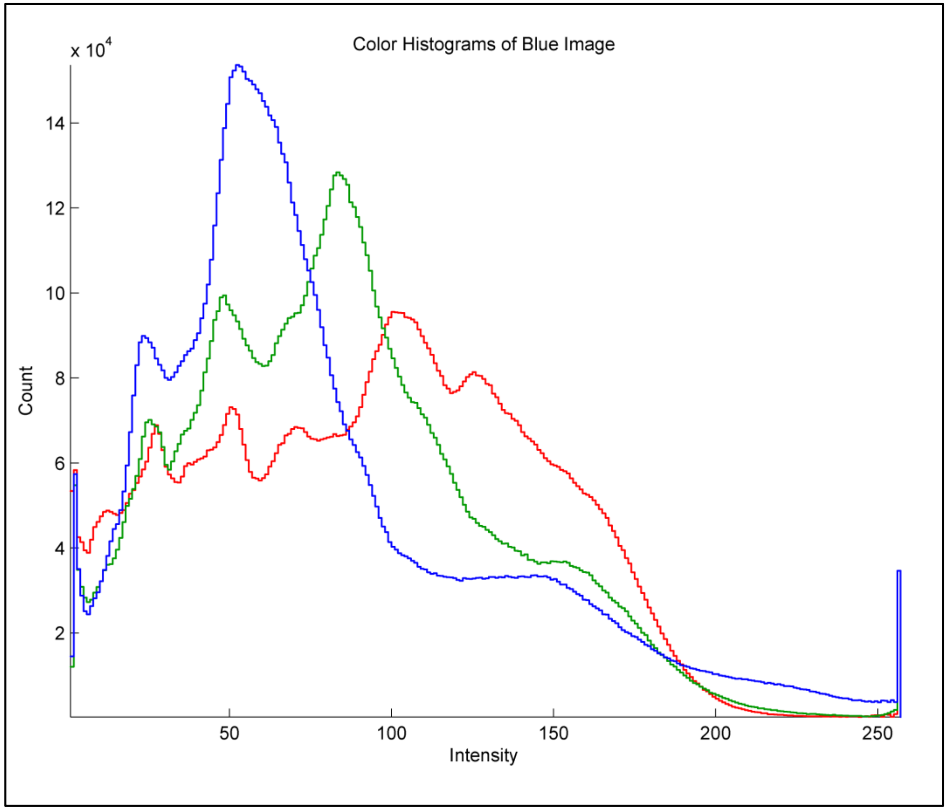


Photo by Wikipedia user, Quartl: http://en.wikipedia.org/wiki/File:Dendrobates_azureus_qtl1.jpg/ .

Color Intensity Distributions

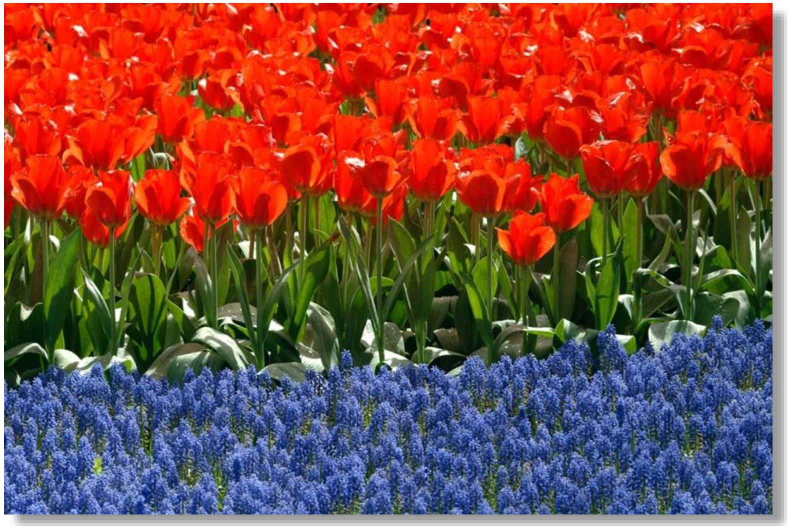


Photo taken in the gardens at Keukenhof, Holland, The Netherlands. RGB primaries dominant at different intensities: blue shadows, green tulip stems, blue hyacinths, red tulip flowers.

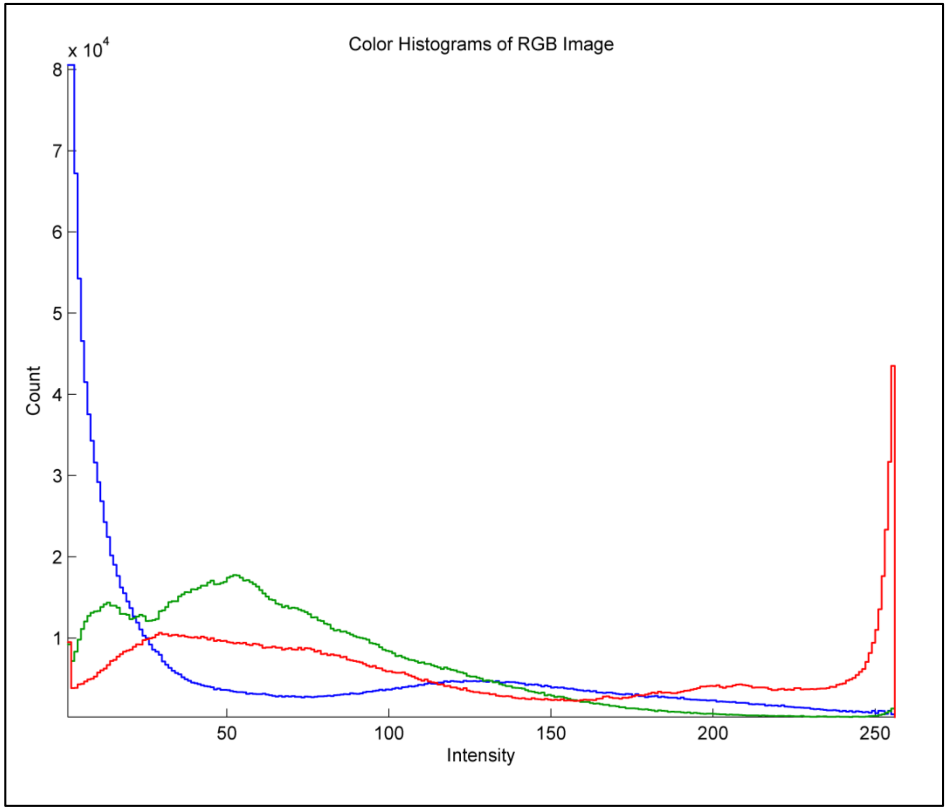
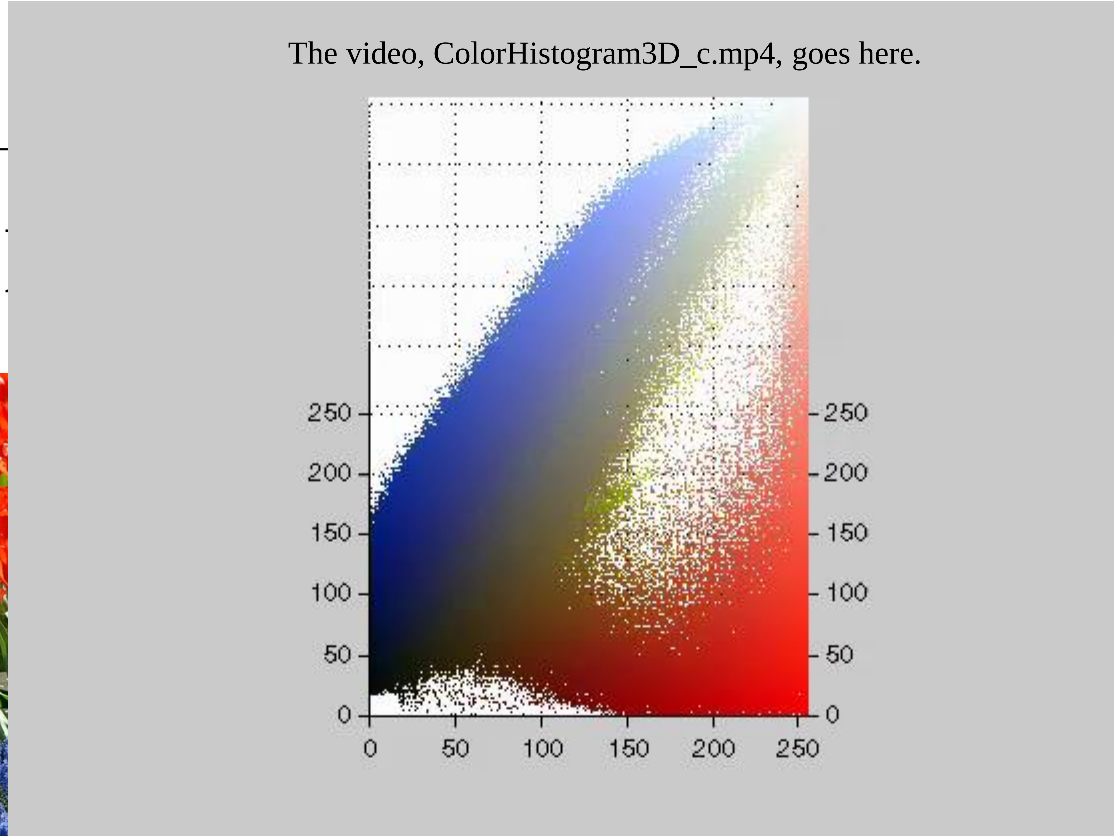
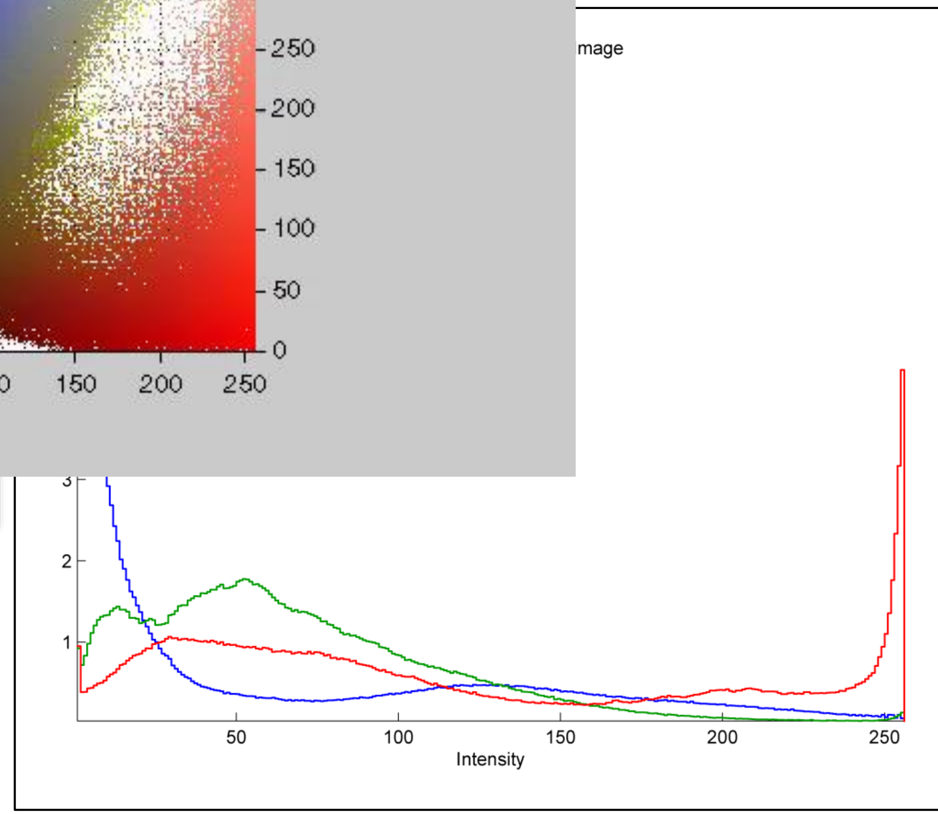


Photo by Jim Pyre: http://thedude.com/archives/2005/04/amsterdam.html

Color



Notice the Necker cube illusion[1] as the virtual cube rotates.



The video is a 3D true histogram (unique index for 1 RGB 3-tuple / color) of the image. The color of each occupied voxel corresponds to a color in the image. This is for visualization only. The actual 3D histogram contains scalar counts of the colors at each location.

[1] http://en.wikipedia.org/wiki/Necker_cube

Image Histograms: Monochrome

The histogram of an image is a tally of the number of pixels at each intensity level or color. For a monochrome image **G**,

$$H_{\mathbf{G}}(g) = \#\{\mathbf{p} \mid \mathbf{G} = g\}.$$

The value of the histogram at *g* is the number of pixels for which image **G** has intensity level *g*. For an 8-bit image, *H* has 256 values

$$H_{\mathbf{G}} : \{0, \ldots, 255\} \rightarrow \{0, \ldots, RC\}.$$

If G is an $R \times C$ image and all its pixels have the same intensity, g_0, then $H(g_0) = RC$ and $H(g) = 0$ for all intensities $g \neq g_0$.

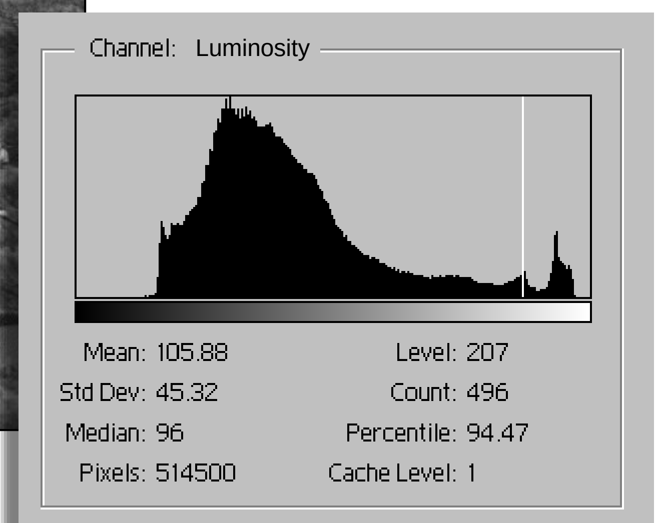
Image Histograms: Monochrome

- If **I** is a 1-band (monochrome) image, then
- the pixel $\mathbf{I}(r,c)$ is an 8-bit integer between 0 and 255.
- The histogram, $h_{\mathbf{I}}$, of **I** is:
  - a 256-element array, $h_{\mathbf{I}}$, where
  - $h_{\mathbf{I}}(g)$ is an integer for g = 1, 2, 3, …, 256, such that
  - $h_{\mathbf{I}}(g)$ = number of pixels in **I** that have value g-1.

In Matlab an array of length *n* has indices from 1 to *n*. In many computer languages, *e.g.* "C" or "C++" an n-element array is indexed from 0 to *n*-1.

The Histogram of a Monochrome Image

$h_{\mathbf{I}}(g+1)$ = the number of pixels in **I** with intensity level, g.



Giant Mesquite Bug

a.k.a. Banjo Beetle

24-bit truecolor image

Photo by Alan Peters,
Tucson, Arizona 1986.

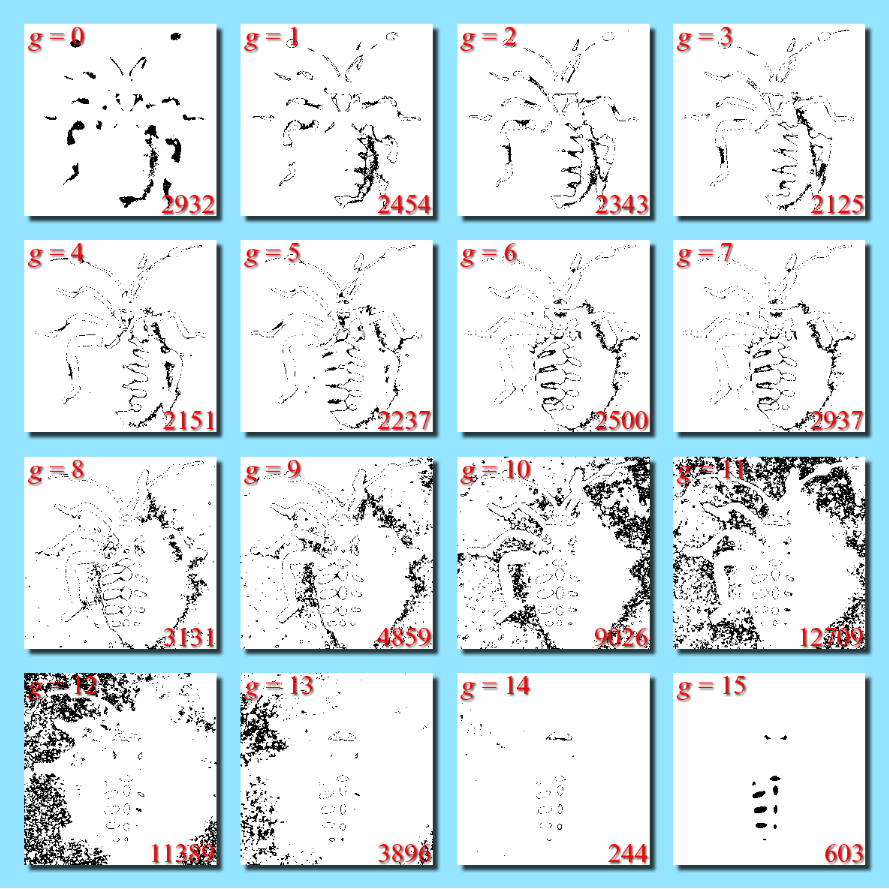
During early summer in the Sonoran Desert of Southern Arizona, clusters of large, strange-looking, red and white bugs can be spotted on the foliage of mesquite trees (Prosopis spp.). These colorful bugs are the immature, wingless nymphs of the Giant Mesquite Bug or Leaf-footed Bug (Thasus neocalifornicus).

Photo and description by T. Beth Kinsey
http://fireflyforest.net, 2008

The Histogram of a Monochrome Image

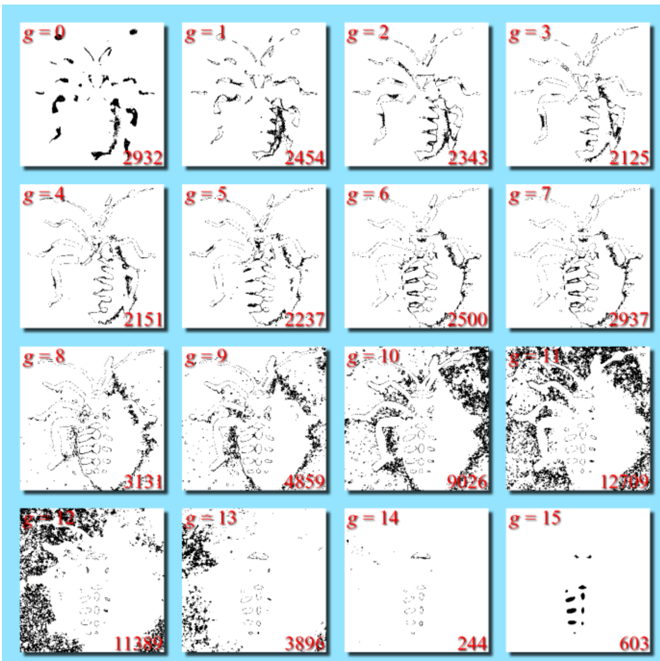
16-level (4-bit) image

lower RHC: number of pixels with intensity g

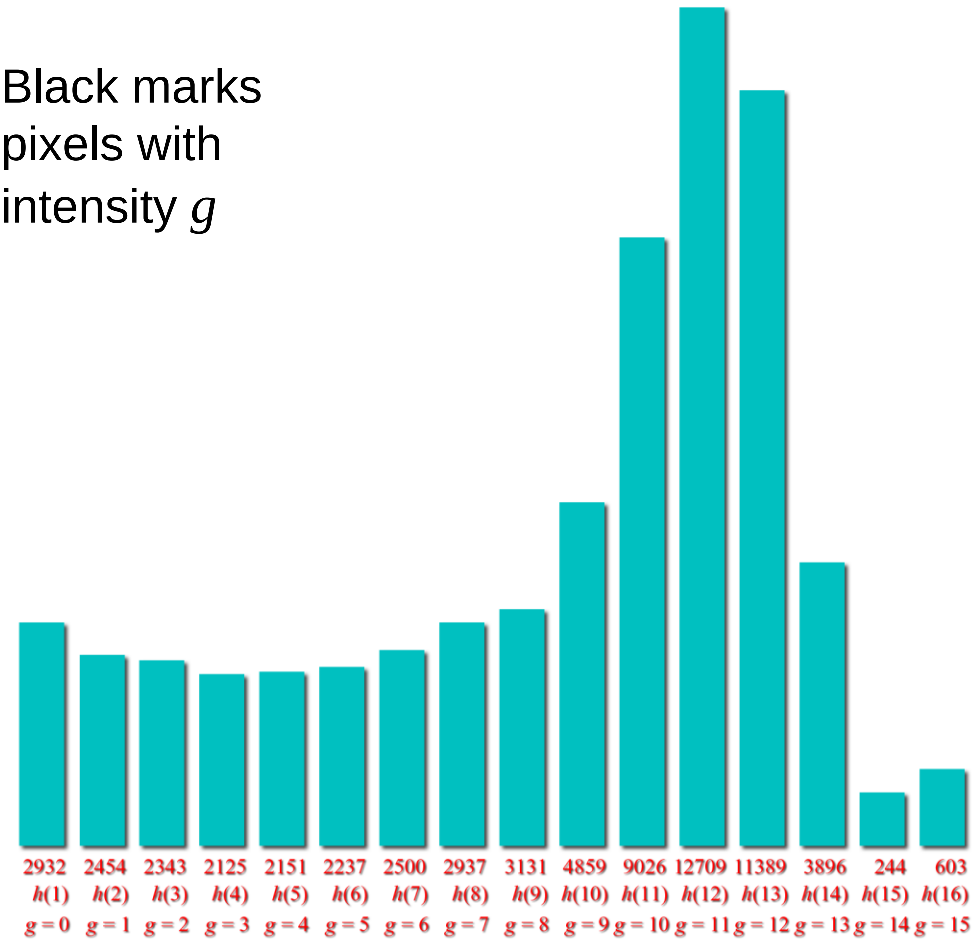


black marks pixels with intensity g

The Histogram of a Monochrome Image

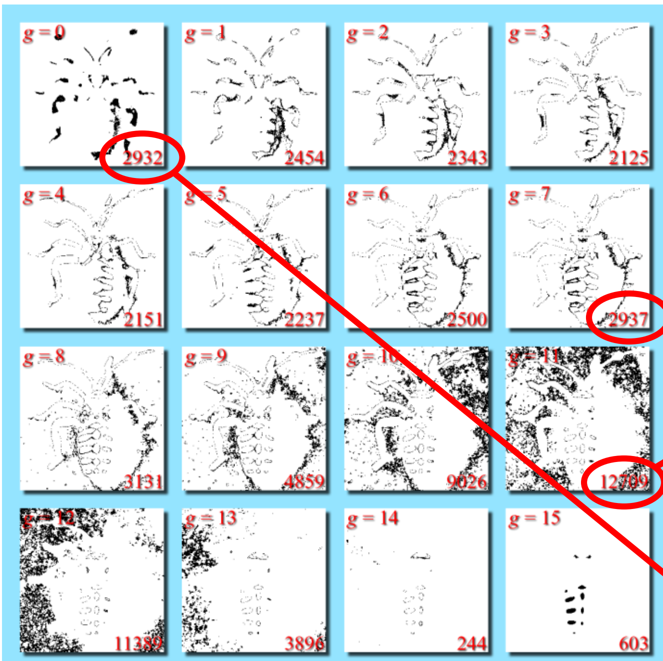


Black marks pixels with intensity g



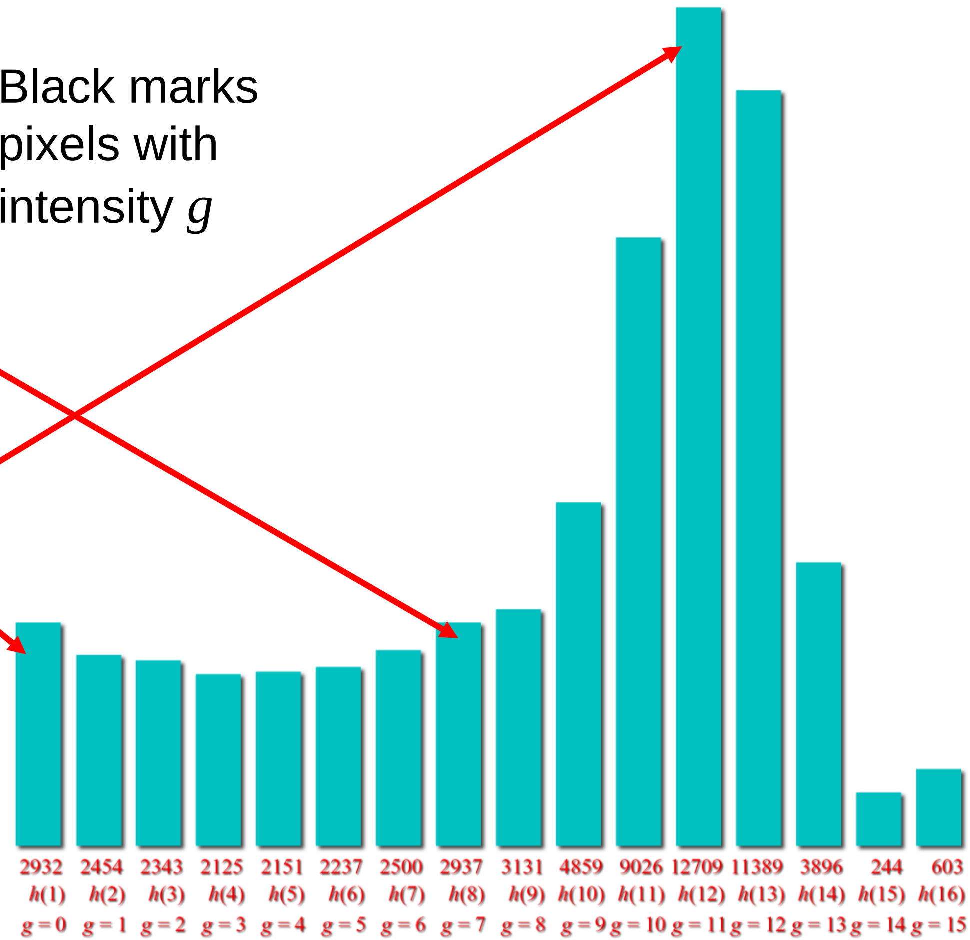
Plot of histogram:
number of pixels with intensity g

The Histogram of a Monochrome Image



Black marks pixels with intensity g

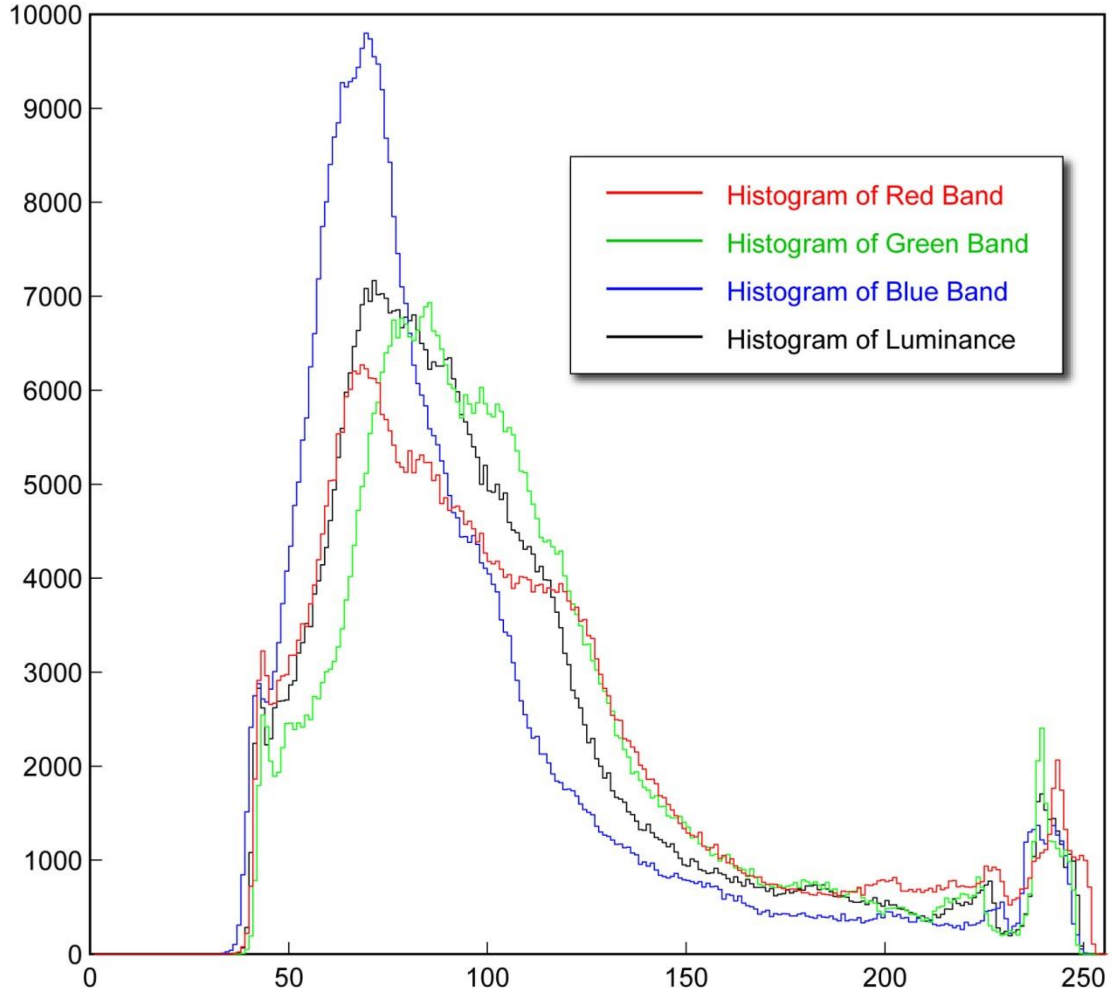
Plot of histogram:
number of pixels with intensity g



The Histograms of a Color Image

- If **I** is a 3-band image (truecolor, 24-bit)
- then $\mathbf{I}(r,c,b)$ is an integer between 0 and 255.
- Either **I** has 3 histograms:
  - $h_{\mathbf{R}}(g+1)$ = # of pixels in $\mathbf{I}(:,:,1)$ with red intensity value g
  - $h_{\mathbf{G}}(g+1)$ = # of pixels in $\mathbf{I}(:,:,2)$ with green intensity value g
  - $h_{\mathbf{B}}(g+1)$ = # of pixels in $\mathbf{I}(:,:,3)$ with blue intensity value g
- or 1 vector-valued histogram, $\mathbf{h}(g+1,1,b)$ where
  - $\mathbf{h}(g+1,1,1)$ = # of pixels in **I** with red intensity value g
  - $\mathbf{h}(g+1,1,2)$ = # of pixels in **I** with green intensity value g
  - $\mathbf{h}(g+1,1,3)$ = # of pixels in **I** with blue intensity value g

The Histograms of a Color Image



Value Image

How to extract a monochrome intensity image from a color image.

If **I** is an rgb image, then **I**'s value image, **V**, has one band that is the pixel-wise average of **I**'s **R**, **G**, & **B** bands:

$$\mathbf{V}(r,c)=\frac{1}{3}\left[\mathbf{R}(r,c)+\mathbf{G}(r,c)+\mathbf{B}(r,c)\right].$$

This is easily computed in Matlab by

```
V=sum(I,3)/3;
```

The 3 in the 2nd argument of sum tells it to act along dimension 3 of the image – across the color bands.

Luminance Image

How to extract a monochrome intensity image from a color image.

I's luminance image, **L**, is a 1-band image that is a specific, weighted, pixel-wise average of **I**'s **R**, **G**, and **B** bands:

$$\mathbf{L}(r,c) = 0.299 \cdot \mathbf{R}(r,c) + 0.587 \cdot \mathbf{G}(r,c) + 0.114 \cdot \mathbf{B}(r,c)$$

The numbers were derived by the NTSC[1] to weight each color band according to the relative intensity resolution that color by the human eye. The following Matlab code will compute it (very quickly)

```
L = uint8(sum(bsxfun(@times,double(I),...
    reshape([0.299 0.587 0.114],[1 1 3])),3));
```

but it is not obvious how it does. That is explained on the next slide.

[1] National Television System Committee, 1953, http://en.wikipedia.org/wiki/NTSC

Computing the Luminance Image in Matlab

The first steps are to create a 1 × 1 × 3 matrix (vector) containing the weights

```
w = reshape([0.299 0.587 0.114],[1 1 3]);
```

and to convert the `uint8`, 3-band image, `I`, to class double,

```
J = double(I);
```

`bsxfun` combines image `J` and vector `w` using `@times`, a multiplication operator. Effectively, it makes an image, `W`, the same size as `J` with a copy of `w` in every pixel location. Then it multiplies the 2 images together pointwise.

```
T = bsxfun(@times,J,w);
```

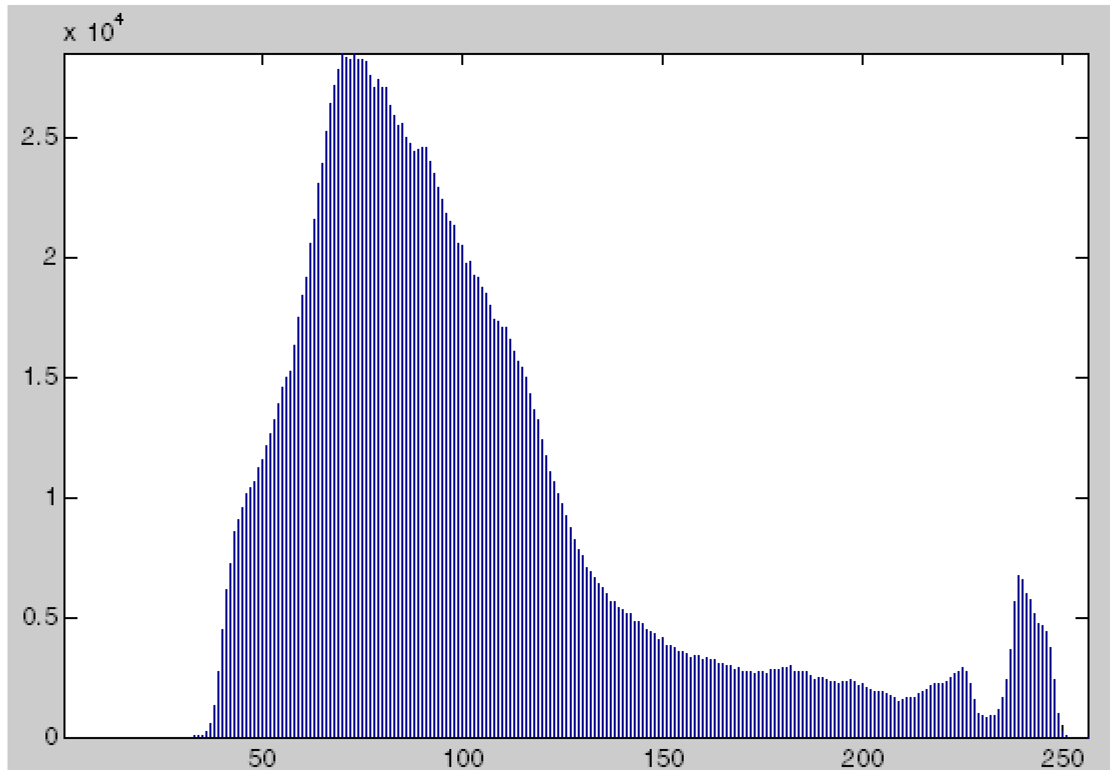
`T` has bands 0.299`R`, 0.587`G`, and 0.144`B`, which are then summed and converted (rounded) to class `uint8`.

```
L = uint8(sum(T,3));
```

All the steps are combined into 1 line of code on the previous page.

Value Histogram

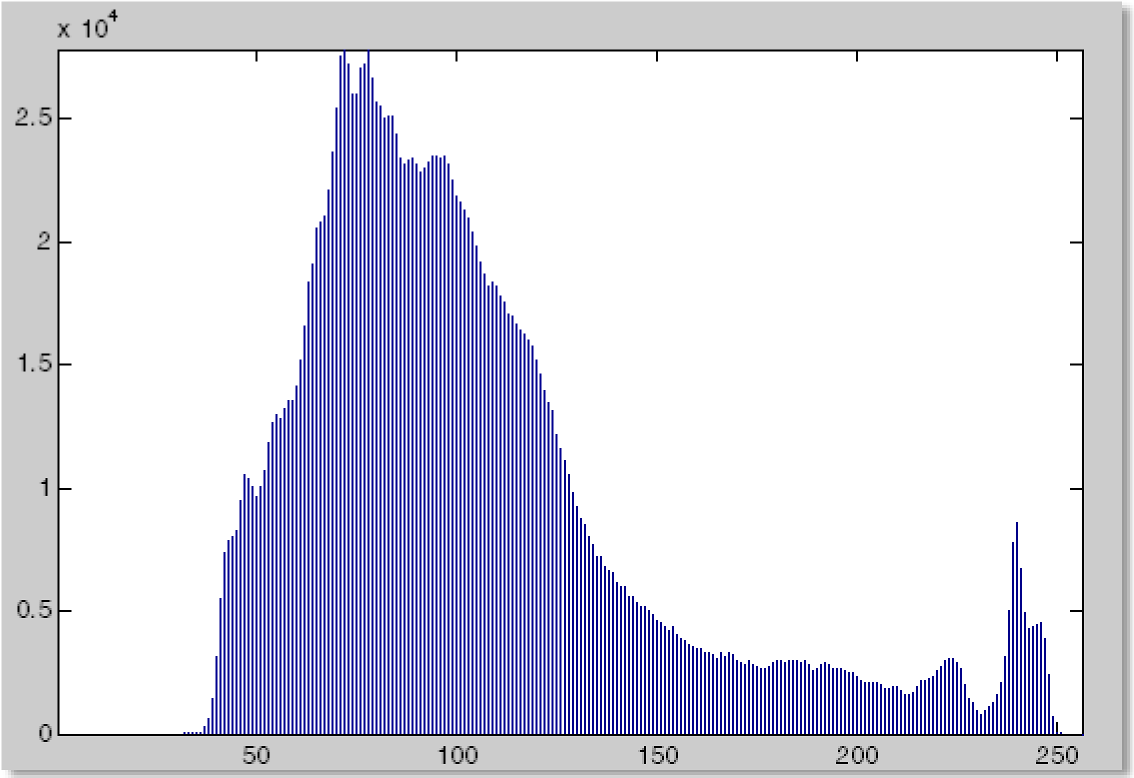
Value image, **V**.



Histogram of the value image.

Luminance Histogram

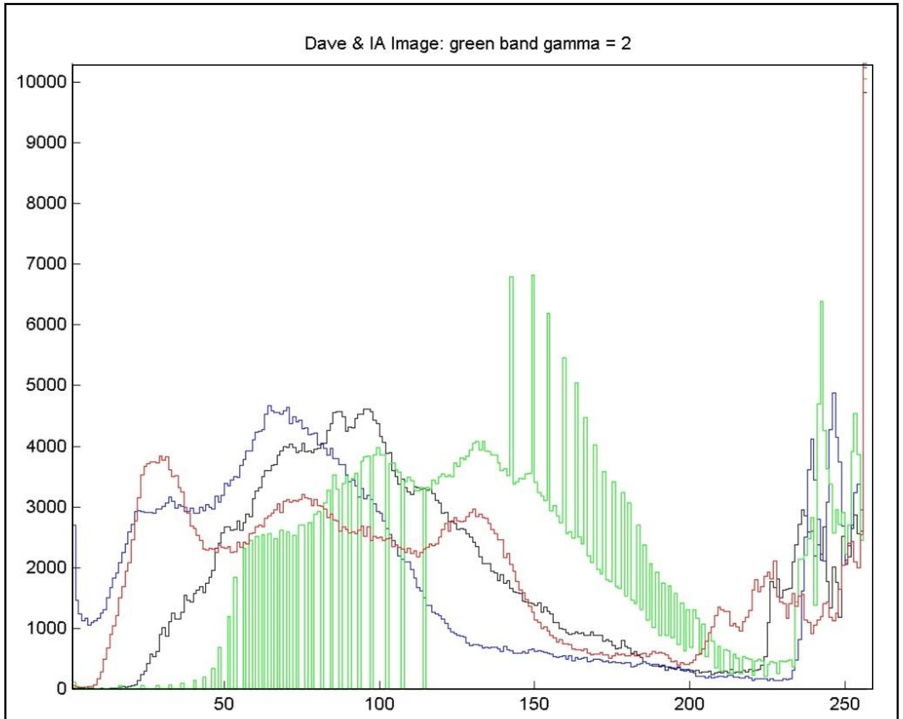
Luminance image, **L**.



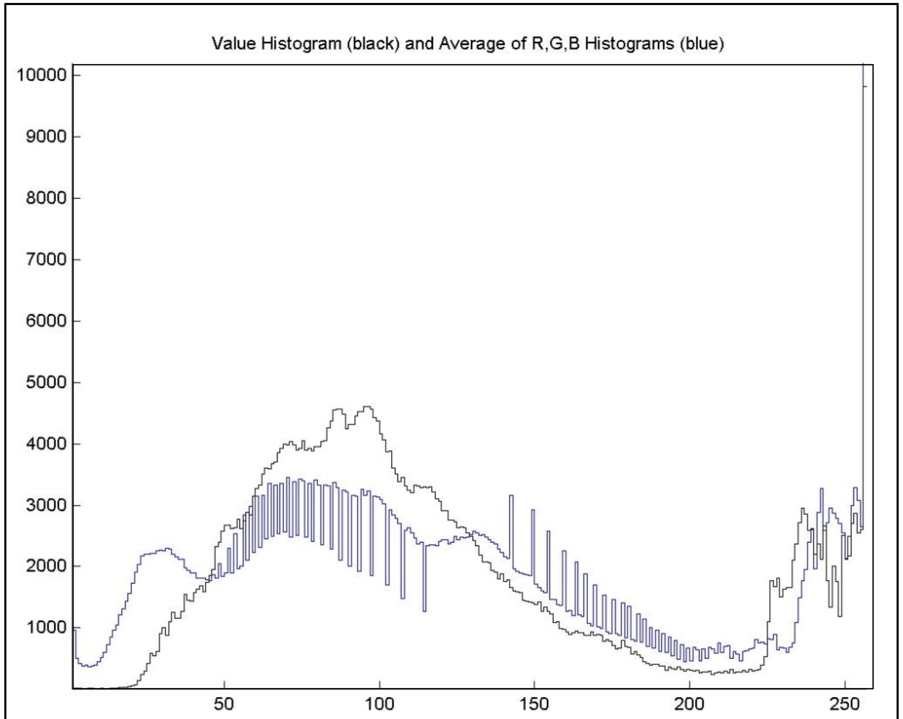
Histogram of the luminance image.

The value histogram is **not** the average of the three 1-D color intensity histograms.

Value Histogram vs. Average of R,G,&B Histograms



R,G,B,&V histograms



V & avg. of R,G,&B histograms

Multi-Band Histogram Calculator in Matlab

```
% Multi-band histogram calculator
function h=histogram(I)

[R C B]=size(I);

% allocate the histogram
h=zeros(256,1,B);

% range through the intensity values
for g=0:255
   h(g+1,1,:) = sum(sum(I==g)); % accumulate
end

return;
```

This is an old, slow version. See pp 28-29.

Multi-Band Histogram Calculator in Matlab

```
% Multi-band histogram calculator
function h=histogram(I)

[R C B]=size(I);

% allocate the histogram
h=zeros(256,1,B);

% range through the intensity values
for g=0:255
   h(g+1,1,:) = sum(sum(I==g)); % accumulate
end
```

Loop through all intensity levels (0-255)
Tag the elements that have value g.
The result is an RxCxB *logical* array that has a 1 wherever $\mathbf{I}(r,c,b) = g$ and 0's everywhere else.
Compute the number of ones in each band of the image for intensity g.
Store that value in the 256x1xB histogram at $h(g+1,1,b)$.

If `B==3`, then `h(g+1,1,:)` contains 3 numbers: the number of pixels in bands 1, 2, & 3 that have intensity `g`.

`sum(sum(I==g))` computes one number for each band in the image.

Vectorized Single-Band Histogram Calculator

```
% Vectorized single-band histogram calculator using
% Matlab's built-in histogram calculator, histcounts().
% Result, h, is 256×1 for a one-band, uint8 image.

function h = histogram(I)

   h = histcounts(I,(0:256))';

end;
```

(0:256) are the bin boundaries.
# of pixels < 1 is in the 1st bin.
# of pixels ≥ 255 & < 256 are in the last bin

histcounts is a new function in version 2014b of Matalab

Vectorized Multi-Band Histogram Calculator

```
% Vectorized multi-band histogram calculator using
% Matlab's built-in histogram calculator, histcounts().
% Result, h, is 256×1 for a one-band image and 256×1×3
% for a three-band, uint8 image.

function h = histogram(I)

   [~,~,B] = size(I);
   h = zeros(256,1,B);
   for b = 1:B
     h(:,1,b) = histcounts(I(:,:,b),[0:256])';
   end;

end;
```

The Probability Density Function of an Image

Let $A = \sum_{g=0}^{255} h_{\mathbf{I}_k}(g+1)$.

Note that since $h_{\mathbf{I}_k}(g+1)$ is the number of pixels in $\mathbf{I}_k$ (the kth color band of image $\mathbf{I}$) with value g, A is the number of pixels in $\mathbf{I}$. That is if $\mathbf{I}$ is R rows by C columns then $A = R \times C$.

Then,

$$p_{\mathbf{I}_k}(g+1) = \frac{1}{A} h_{\mathbf{I}_k}(g+1)$$

is the graylevel probability density function of $\mathbf{I}_k$.

pdf
[lower case]

This is the probability that an arbitrary pixel from $\mathbf{I}_k$ has value g.

The Probability Density Function of an Image

- $p_{\text{band}}(g+1)$ is the fraction of pixels in (a specific band of) an image that have intensity value g.
- $p_{\text{band}}(g+1)$ is the probability that a pixel randomly selected from the given band has intensity value g.
- Whereas the sum of the histogram $h_{\text{band}}(g+1)$ over all g from 1 to 256 is equal to the number of pixels in the image, the sum of $p_{\text{band}}(g+1)$ over all g is 1.
- p_{band} is the normalized histogram of the band.

The Probability Distribution Function of an Image

Let $\mathbf{q} = [q_1\ q_2\ q_3] = \mathbf{I}(r,c)$ be the value of a randomly selected pixel from $\mathbf{I}$. Let g be a specific graylevel. The probability that $q_k \leq g$ is given by

$$P_{I_k}(g+1) = \sum_{\gamma=0}^{g} p_{\mathbf{I}_k}(\gamma+1) = \frac{1}{A}\sum_{\gamma=0}^{g} h_{\mathbf{I}_k}(\gamma+1) = \frac{\sum_{\gamma=0}^{g} h_{\mathbf{I}_k}(\gamma+1)}{\sum_{\gamma=0}^{255} h_{\mathbf{I}_k}(\gamma+1)},$$

where $h_{\mathbf{I}k}(\gamma+1)$ is the histogram of the kth band of $\mathbf{I}$.

PDF [upper case]

This is the probability that any given pixel from $\mathbf{I}_k$ has value less than or equal to g.

The Probability Distribution Function of an Image

Let $\mathbf{q} = [q_1\ q_2\ q_3] = \mathbf{I}(r,c)$ be the value of a randomly selected pixel from $\mathbf{I}$. Let g be a specific graylevel. The probability that $q_k \le g$ is given by

$$P_{I_k}(g+1) = \sum_{\gamma=0}^{g} p_{\mathbf{I}_k}(\gamma+1) = \frac{1}{A}\sum_{\gamma=0}^{g} h_{\mathbf{I}_k}(\gamma+1) = \frac{\sum_{\gamma=0}^{g} h_{\mathbf{I}_k}(\gamma+1)}{\sum_{\gamma=0}^{255} h_{\mathbf{I}_k}(\gamma+1)},$$

where $h_{\mathbf{I}k}(\gamma+1)$ is the histogram of the kth band of $\mathbf{I}$.

Also called CDF for "Cumulative Distribution Function".

This is the probability that any given pixel from $\mathbf{I}_k$ has value less than or equal to g.

The Cumulative Distribution Function of an Image

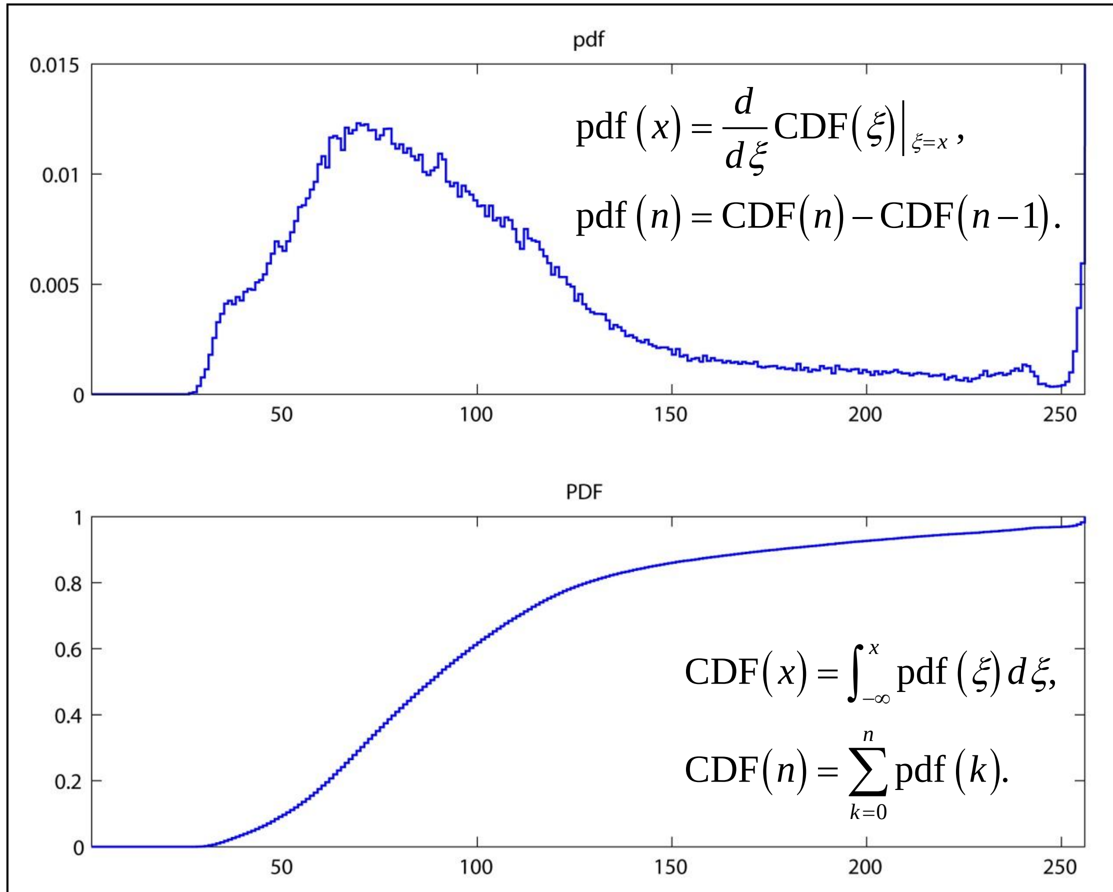
- $P_{\text{band}}(g+1)$ is the fraction of pixels in (a specific band of) an image that have intensity values less than or equal to g.
- $P_{\text{band}}(g+1)$ is the probability that a pixel randomly selected from the given band has an intensity value less than or equal to g.
- $P_{\text{band}}(g+1)$ is the cumulative (or running) sum of $p_{\text{band}}(g+1)$ from 0 through g inclusive.
- $P_{\text{band}}(1) = \mathrm{p}_{\text{band}}(1)$ and $P_{\text{band}}(256) = 1$; $P_{\text{band}}(g+1)$ is nondecreasing.

Note: the Probability Distribution Function (PDF, capital letters) and the Cumulative Distribution Function (CDF) are exactly the same things. Both PDF and CDF will refer to it. However, pdf (small letters) is the *density* function.

The pdf vs. the CDF

pdf = backward difference of CDF.

CDF = running sum of pdf.



Simple Matlab Code for 1-Band Images

```
% Matlab code for histogram (h), pdf (p), and CDF (C)
% of a 1-band image, I.

% I’s 256-bin histogram
h = histcounts(I,[0:256])’;
% I’s pixel-intensity probability density function
p = h/length(I(:));
% I’s pixel-intensity cumulative distribution function
C = cumsum(p);
```

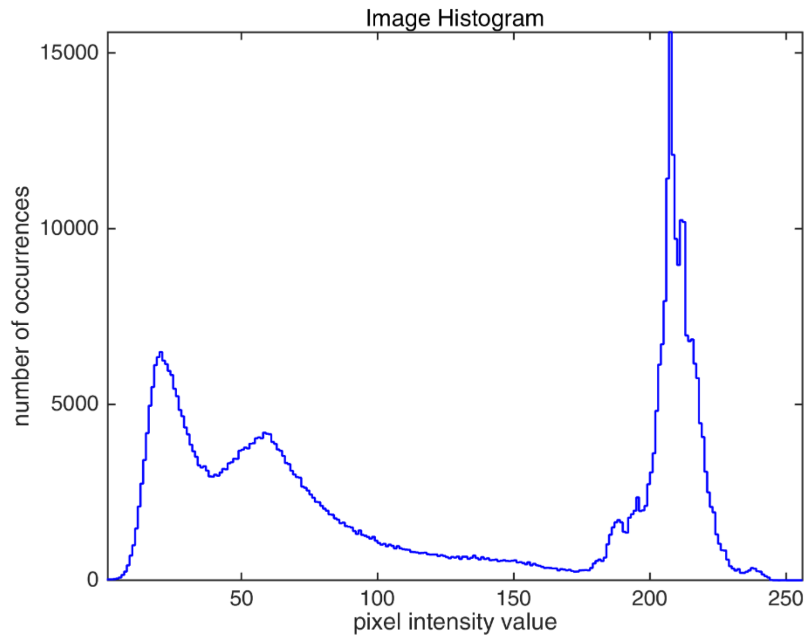
Code for Histograms of 3-Band Images

```
% Matlab code for red, green, blue, value, and luminance
% histograms of a 3-band image, I.

hr = histcounts(I(:,:,1),[0:256])'; % red-band histogram
hg = histcounts(I(:,:,2),[0:256])'; % green-band histogram
hb = histcounts(I(:,:,3),[0:256])'; % blue-band histogram
V  = uint8(sum(I,3)/3);             % value image
hk = histcounts(V,[0:256])';        % value histogram
L  = uint8(0.299*I(:,:,1) + ...     % luminance image
           0.587*I(:,:,2) + 0.114*I(:,:,3);
lk = histcounts(L,[0:256])';        % luminance histogram
```

Histogram Shapes

The histogram of a continuous tone image is usually a relatively smooth curve with peaks at the most frequently occurring intensities:



Histogram Shapes

If an image has had its contrast altered, often that can be seen in its histogram.