MODULE 2: FOUNDATIONS OF COMPUTER VISION SYSTEMS 1

Dr. Matthew CHUA, PhD

Institute of System Science, NUS

Main References

- Sonka, M., Hlavac, V., & Boyle, R. (2014). *Image processing, analysis, and machine vision*. Cengage Learning.
- Rafael C.. Gonzalez, Richard E.. Woods, & Steven L.. Eddins. (2010). *Digital Image Processing Using MATLAB®*. McGraw Hill Education.
- Ogata, K., & Yang, Y. (1970). Modern control engineering.
- CS131 Computer Vision: Foundations and Applications, Stanford University.http://vision.stanford.edu/teaching/cs131_fall1516/lectures/lecture5_edges_cs131.pdf
- David Forsyth and Jean Ponce –Chapter 8, Computer Vision: A Modern Approach.
- Chui *et. al.* ME5405 Machine Vision. NUS

Contents of this lecture

1. Image Enhancement

2. Image Segmentation

3. Color Image Processing

Image Enhancement

Introduction

Image enhancement is to process an image so that the result is more suitable than the original image for specific applications.

The techniques employed are problem oriented. There are 2 broad categories of approaches:

i. Spatial Domain methods:
   - Direct manipulation of the pixels

ii. Frequency Domain methods
   - Based on modifying the Fourier transform of an image

Enhancements using a combinations of these two also exists. However, we will only cover spatial domain techniques in this course.

Background

Spatial Domain Methods process the pixel directly on an image. The function is known as

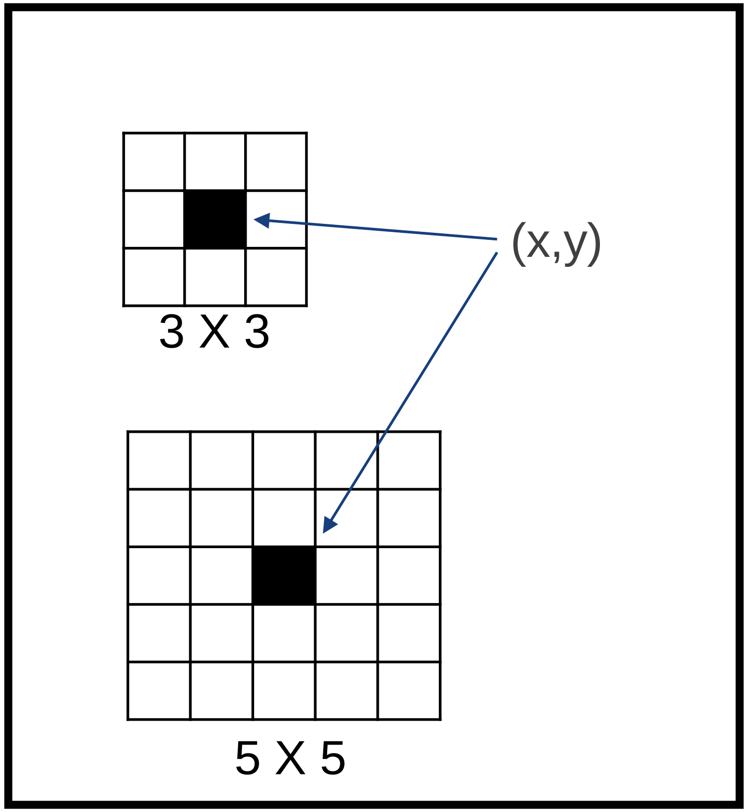
$$g(x, y) = T[f(x, y)]$$

f(x,y) : input image
g(x,y) : processed image
T : an operator on f(x,y), which may be defined over some neighbourhood of (x,y)

T can also operate on a set of input images (e.g. pixel-by-pixel averaging for noise removal).

Background

Neighborhood about (x,y) is usually a square or rectangular area centered at (x,y).The centre of this sub-image is moved from pixel to pixel during the operation. The operator is applied at each location to yield g(x,y).



image

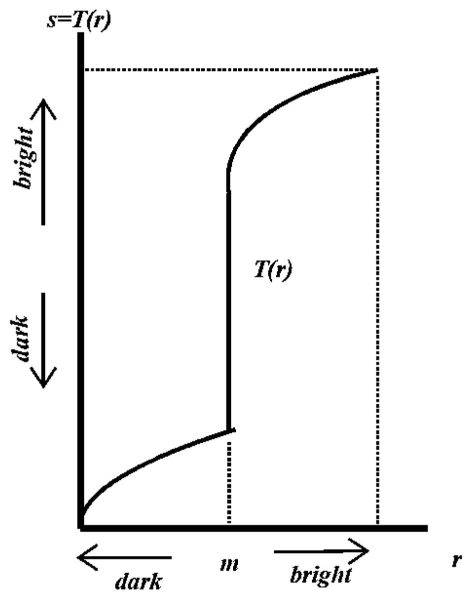
The simplest form of *T* is when the neighborhood is 1 X 1.

g(*x*,*y*) depends only on the value of *f*(*x*,*y*) and *T* is simply a gray-level transformation.

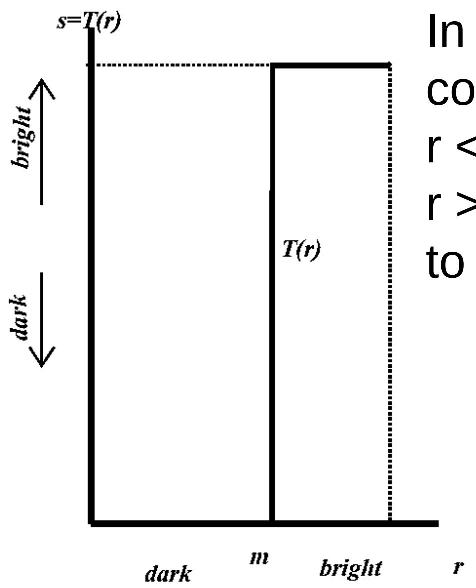
$$s = T(r)$$

r and *s* are the gray-levels of *f*(*x*,*y*) and *g*(*x*,*y*) at (*x*,*y*), respectively

Background



(a)



(b)

Chui *et. al.* ME5405 Machine Vision. NUS

In (a), T(r) produces image of higher contrast:
r < m darkening; and
r > m brightening
to narrower bands contrast - stretching.

In Fig. 2(b) *T(r)*produces a binary image.

Techniques in this category are referred to as **Point Processing.**

Background

The general approach is to let the values of f(x,y) in a pre-defined neighborhood of (x,y) to determine the value of g(x,y).

One of the approaches uses masks (also referred to as template, windows, or filters).

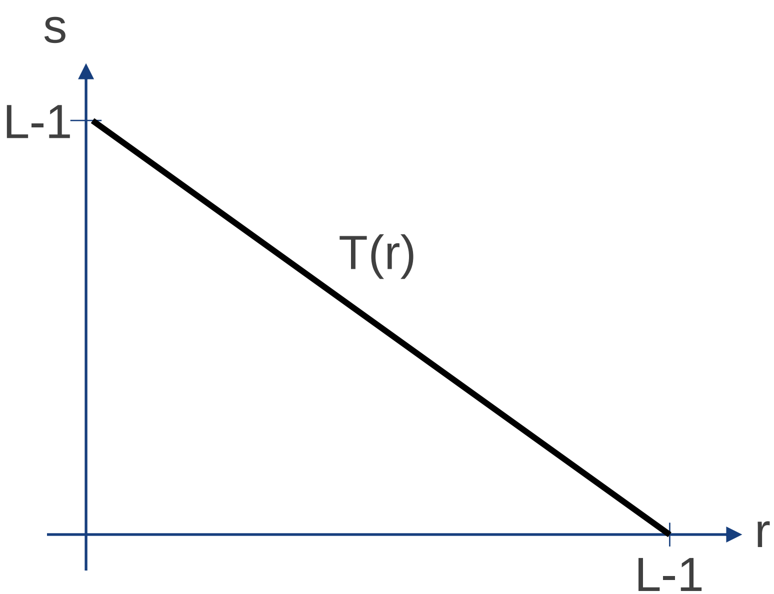
A mask is basically a small (say 3 X 3) 2-D array. The values of the coefficients wi , determine the nature of the process (see Fig. 3).

Image Enhancement based on this type of approach is known as Mask Processing or Filtering.

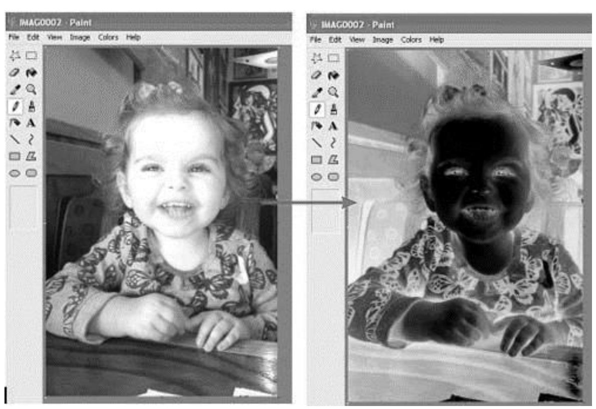
W_1	W_2	W_3
W_4	W_5	W_6
W_7	W_8	W_9

3 by 3 mask

Enhancement by Point Processing



Gray level transformation function



https://www.computerhope.com/jargon/n/negative.htm

Some Simple Intensity Transformations

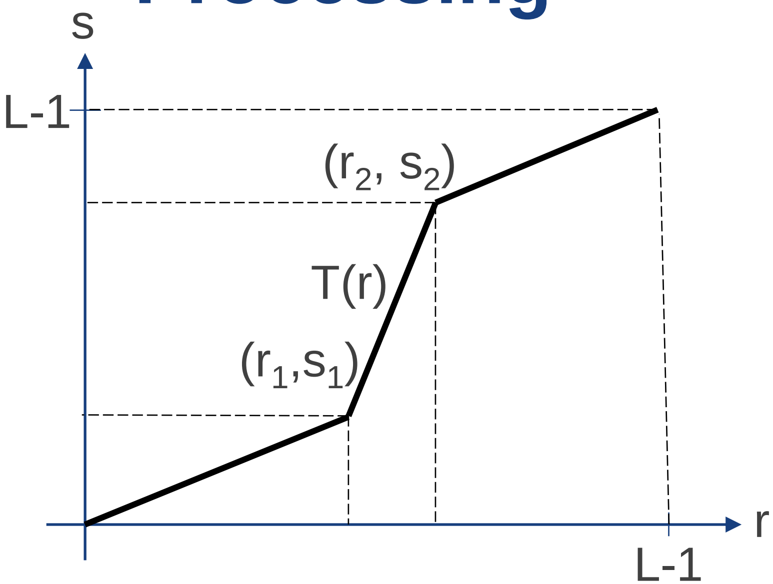
Image Negatives

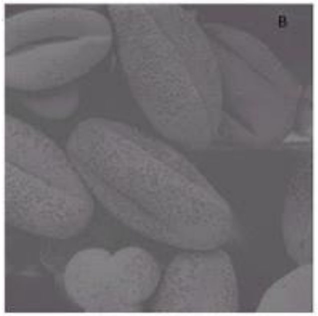
Negatives of digital images are useful in:

- Displaying medical images
- Photography

s=T(r), with T(r) having the shape shown in (a).

Enhancement by Point Processing





Original Low Contrast

Contrast Stretching

Binary thresholding

http://opencvvision.blogspot.com/

Some Simple Intensity Transformations

Contrast stretching

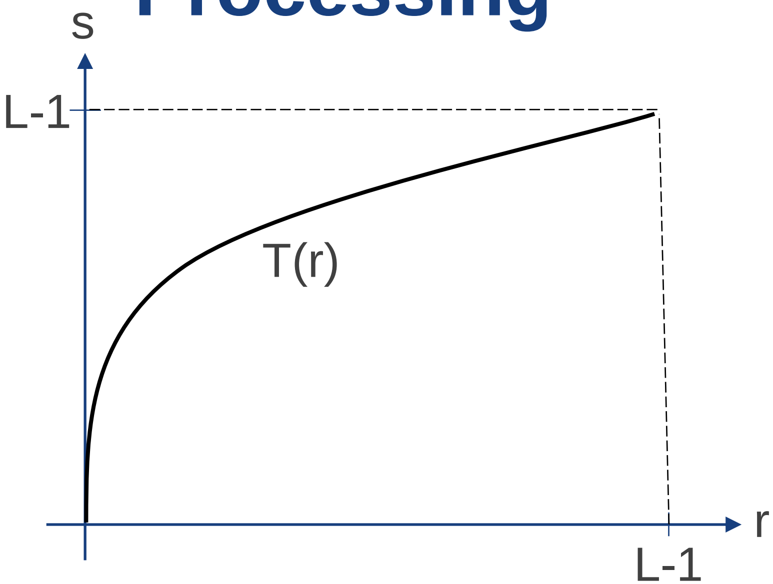
Low contrast results from:

1. Poor illumination
2. Lack of dynamic range
3. Wrong aperture setting during acquisition

T(r) for contrast stretching is shown on left, (r_1,s_1) and (r_2, s_2) control the shape of T(r).

If $r_1=r_2$, $s_1=s_2$ --- no change in image
If $r_1=r_2$, $s_1=0$, s_2 = L-1 --- thresholding function

Enhancement by Point Processing



Original large dynamic range

After transformation

https://fstoppers.com/education/why-dynamic-range-my-favorite-and-why-i-couldnt-care-less-about-megapixel-count-99550

Some Simple Intensity Transformations

Compression of Dynamic Range

When dynamic range of image exceeds capability of display device, only bright parts are visible on the display.

To compress the dynamic range of pixel values, use the following transformation:

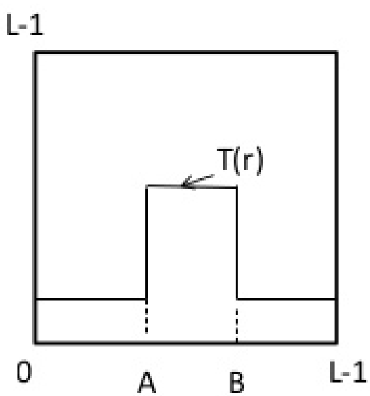
$$s = c.log(1 + |r|)$$

where (1+|r|) yield 0 to 6.4.

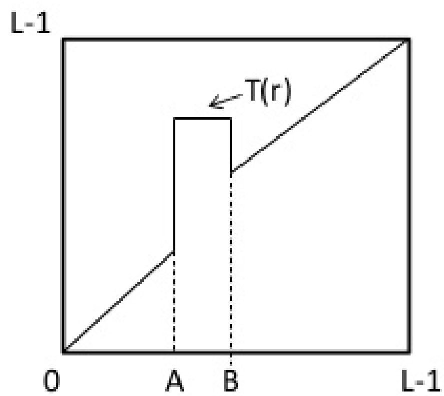
To scale to 0 to 255 (8 bit system), set c = 255/6.4, where c is a scaling constant.

Enhancement by Point Processing

Some Simple Intensity Transformations

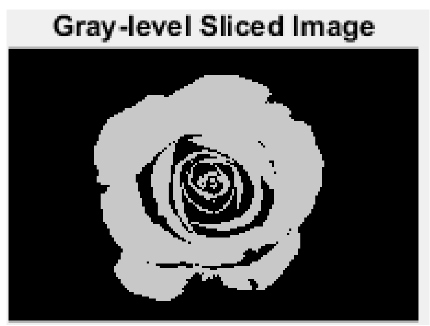


(a)



(b)

(c)



(d)

https://stackoverflow.com/questions/54559177/linear-gray-level-slicing-transform

Gray Level Slicing

Used to highlight specific range of gray levels of interest.

T(r) in (a) produces a binary image. T(r) in (b) brightens the gray levels ranging from A to B, but preserves the background and tonalities in the image.

(c) shows an image and (d) shows the resulting image after the image has been processed by the T(r).

Histogram processing

Histogram for a digital image with gray level in the range [0,L - 1], is a discrete function

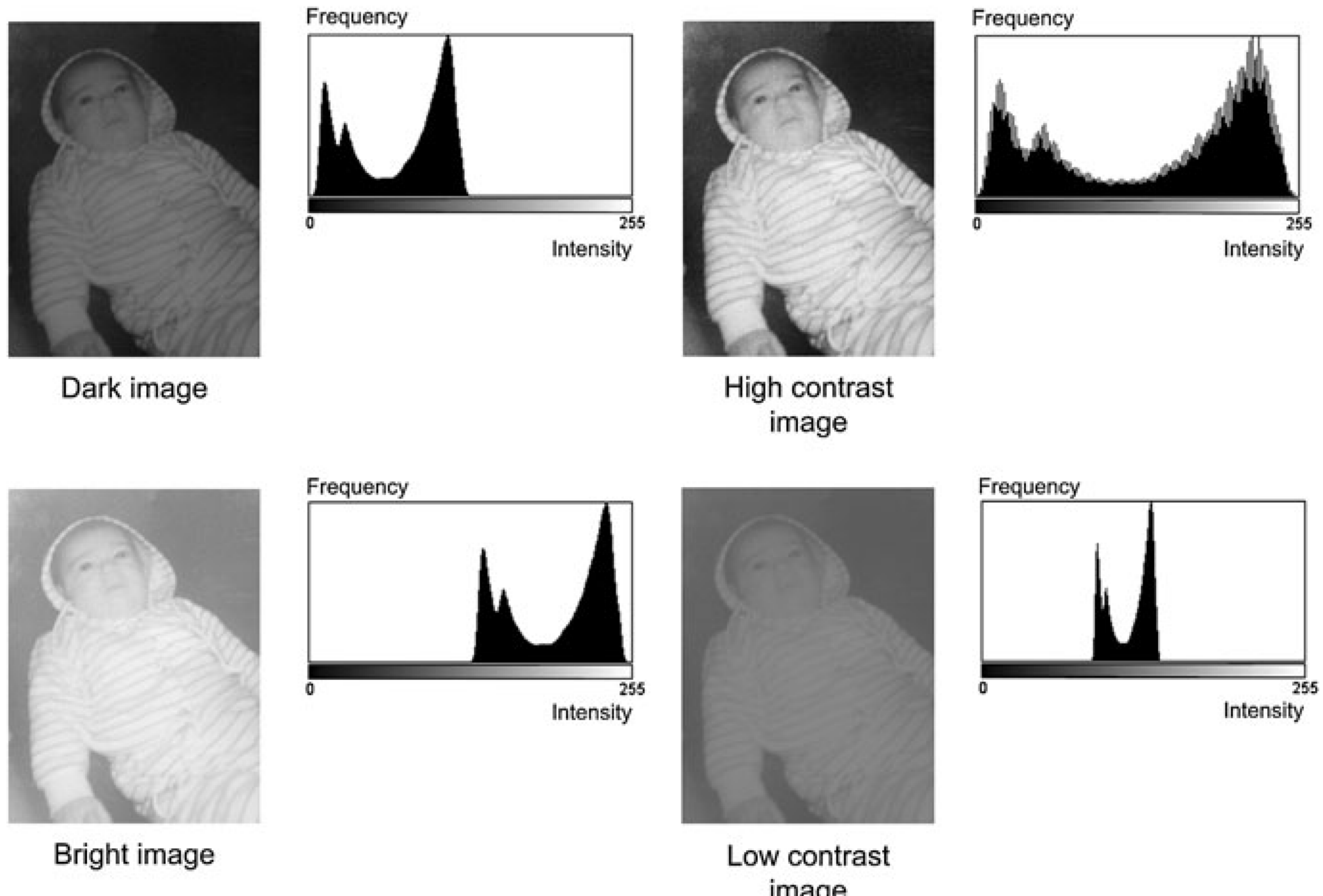
$$p(r_k) = \frac{n_k}{n}$$

where:

r_k is kth gray level

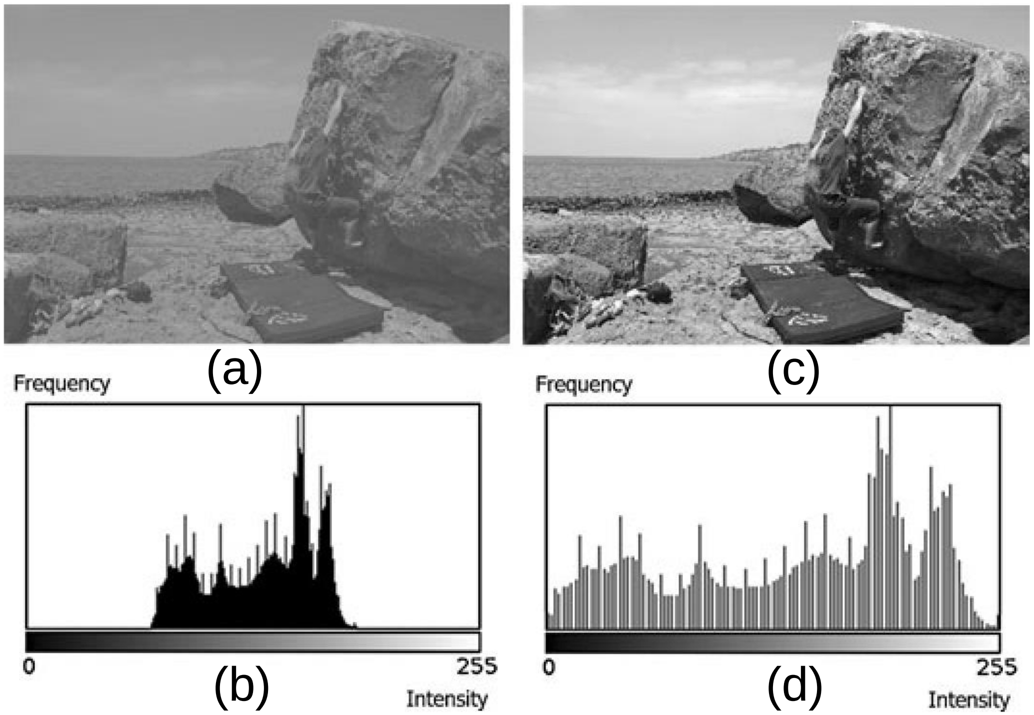
N_k is the number of pixel with gray-level r_k

N is the total number of pixels in image



http://what-when-how.com/introduction-to-video-and-image-processing/point-processing-introduction-to-video-and-image-processing-part-2/

Histogram Equalization – Discrete Example



http://what-when-how.com/introduction-to-video-and-image-processing/point-processing-introduction-to-video-and-image-processing-part-2/

Example

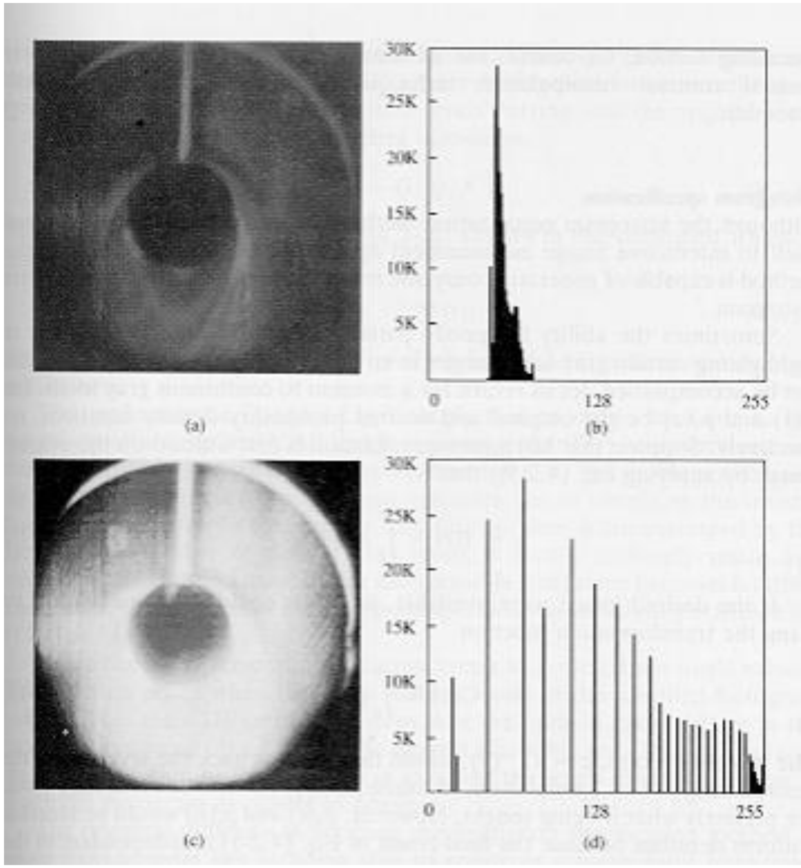
(a) shows an 8-bit image that has poor dynamic range. It has a narrow histogram shown in (b).

(c) shows the result after histogram equalization enhancement, and (d) is its corresponding histogram.

This increases image contrast.

Histogram Equalization – Discrete Example

Another Example



Chui *et. al.* ME5405 Machine Vision. NUS

Grey level	# of pixels	Pr	s	S*255
0	0	0	0	0
45	10500	0.05043	0.05043	13
46	3600	0.01729	0.06772	17
48	24600	0.11816	0.18588	47
50	28500	0.13689	0.32277	82
51	22800	0.10951	0.43228	110
53	18900	0.09078	0.52305	133
54	17100	0.08213	0.60519	154
55	14400	0.06916	0.67435	172
57	12000	0.05764	0.73199	187
60	10800	0.05187	0.78386	200
62	7200	0.03458	0.81844	209
64	6000	0.02882	0.84726	216
66	6600	0.03170	0.87896	224
68	7200	0.03458	0.91354	233
70	5400	0.02594	0.93948	240
71	3600	0.01729	0.95677	244
73	3000	0.01441	0.97118	248
74	1800	0.00865	0.97983	250
76	1500	0.00720	0.98703	252
78	900	0.00432	0.99135	253
80	1800	0.00865	1.0000	255
255	0	0.00000	1.0000	255
Total	208200	1.00000		

Histogram Equalization – Discrete Example

This is a discrete case of **Histogram Equalization** with the following procedure.

1. Determine Pr by dividing each value of # of pixels by the total number of pixels (208200 in this example).
2. Determine the value of sk using the formula:

$$s_k = T(r_k) = \sum_{j=0}^{k} \frac{n_j}{n} = \sum_{j=0}^{k} p_r(rj) \qquad 0 \le r \le 1, \text{ and } k = 0, 1, 2, \ldots, L-1$$

For example,

$$s_{53} = \sum_{j=0}^{53} \frac{n_j}{n} = \sum_{j=0}^{53} pr_j = Pr_0 + Pr_{45} + Pr_{46} + Pr_{48} + Pr_{50} + Pr_{51} + Pr_{53} = 0.52305$$

As the rest of *Pr*’s are equal to zero.

Histogram Equalization – Discrete Example

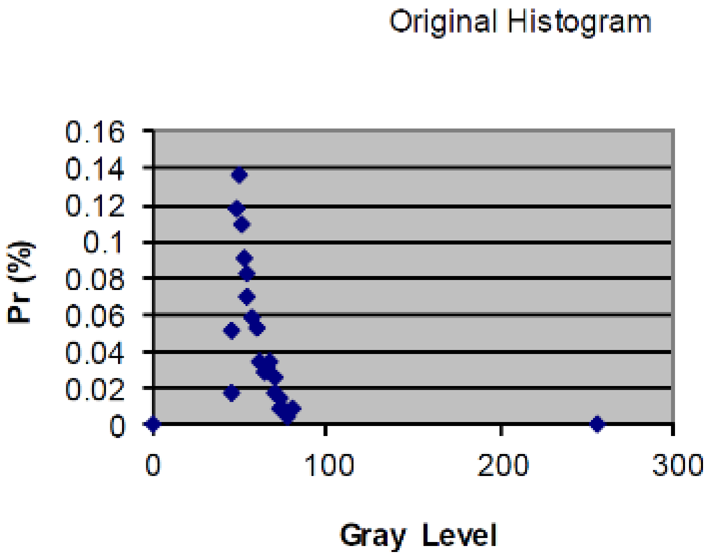
Grey level	# of pixels	Pr	s	S*255
0	0	0	0	0
45	10500	0.05043	0.05043	13
46	3600	0.01729	0.06772	17
48	24600	0.11816	0.18588	47

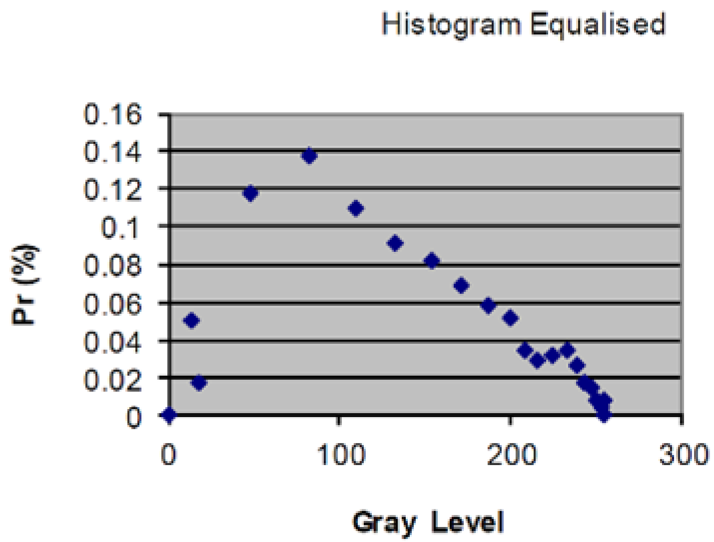
3. Multiply the values of s_k by the maximum value of the gray level, which is 255 in this example (see column marked s*255).

4. We now have the equalized gray level. r is now mapped to s. For example, r = 45 is now mapped to s = 13, and r=71 is mapped to s = 244, and so on.

5. We can now plot the new histogram using s and the corresponding Pr value. For example, when s = 13, use Pr = 0.05043 (Value of Pr when r =45), and when s = 244, Pr = 0.01729 (Value of Pr when r = 71).

Histogram Equalization – Discrete Example





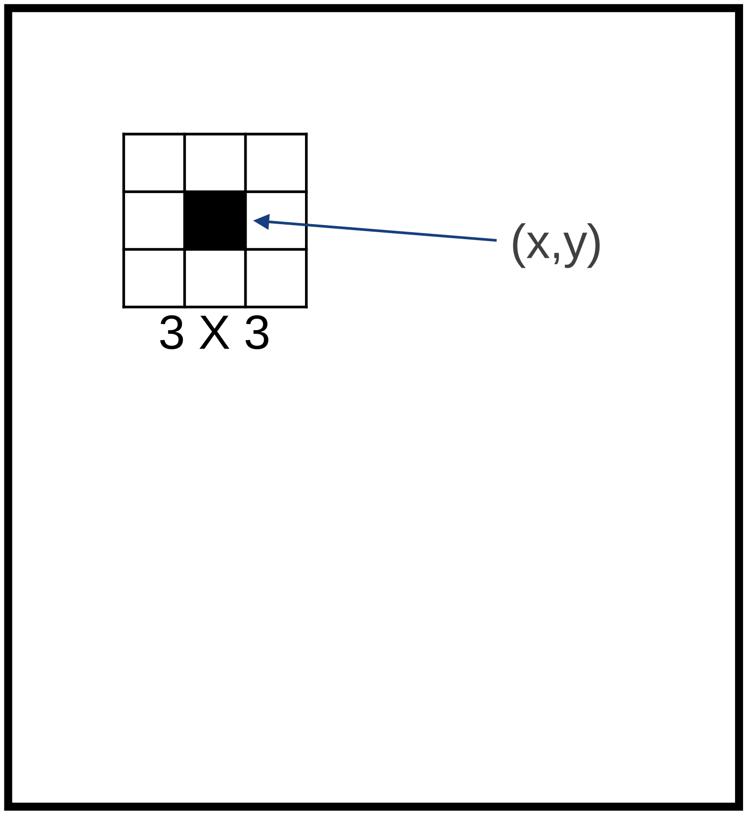
Chui *et. al.* ME5405 Machine Vision. NUS

Histogram based Processing

Local Enhancement

- Necessary to enhance details over small areas for some applications
- Approach is to devise transformation functions based on gray level distribution in neighborhood of every pixel in the image.

Histogram based Processing -Local



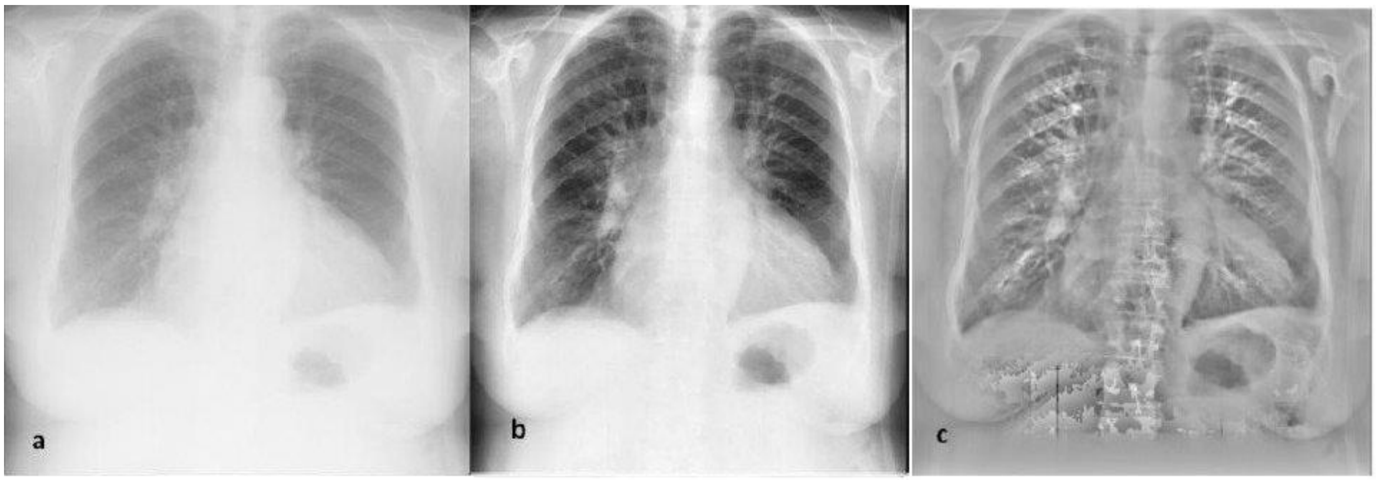
image

Local Enhancement

- Define a square or rectangular neighborhood and move the center of this mask from pixel to pixel.
- At each location, the histogram of the points in the neighborhood is computed, and either a histogram equalization or specification is computed.
- The function is used to map the gray level of the pixel centered in the neighborhood.
- The center of the mask is moved to the adjacent pixel and the procedure is repeated.

Histogram based Processing -Local

Local Enhancement - Example



Original image | Global enhancement | Local enhancement using mask

https://www.researchgate.net/publication/224980640_Fully_Automatic_Lung_Segmentation_and_Rib_Suppression_Methods_to_Improve_Nodule_Detection_in_Chest_Radiographs/figures?lo=1

Other Enhancing Techniques

We can also use other properties of the pixel intensities in a neighborhood to enhance a point.

Two common properties are

m(x,y), intensity mean - measure of average brightness.

σ(x,y), variance - measure of contrast

The transformation is done with

$$g(x,y) = A(x,y).[f(x,y) - m(x,y)] + m(x,y)$$

where

$$A(x,y) = k\frac{M}{\sigma(x,y)} \qquad 0 < k < 1$$

Other Enhancing Techniques

1. m(x,y) and σ(x,y) are computed in a neighborhood centered at (x,y)

2. M is the global mean of f(x,y)

3. k is a constant.

A, m and σ depend on the neighborhood of (x,y). Application of the gain factor A(x,y) amplifies the difference between f(x,y) and the local mean m(x,y), and hence the local variations.

As

$$A(x,y) = \frac{1}{\sigma(x,y)}$$

, areas with low contrast receive larger gain.

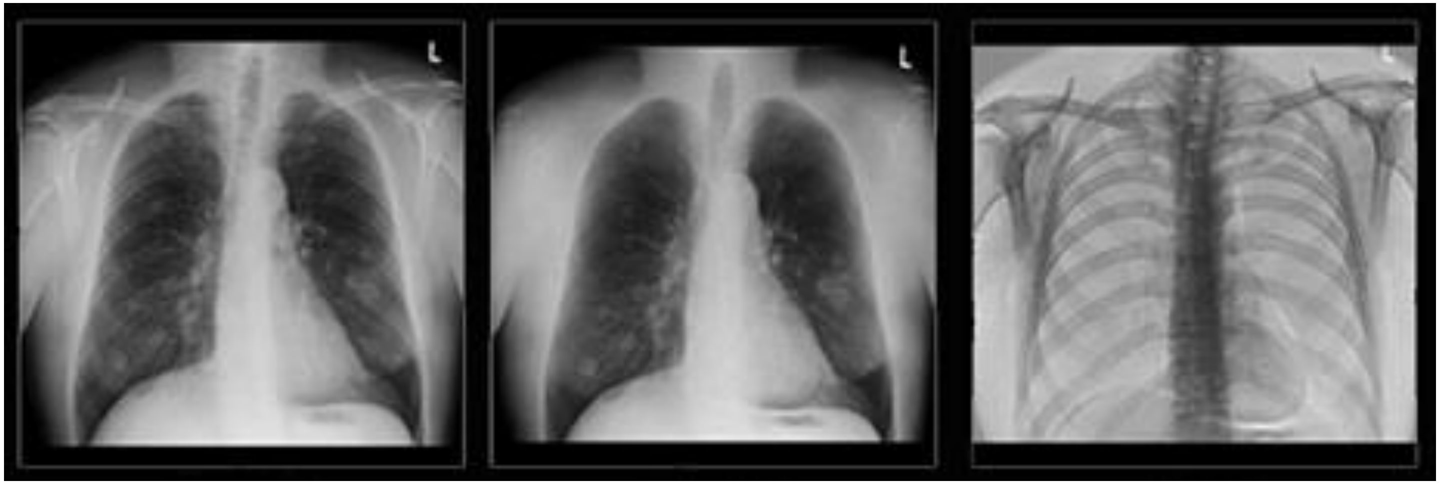
The mean m(x,y) is added back into the equation to restore the average intensity level of the image in the local region.

Image Subtraction

Difference between two images f(x,y) and h(x,y),

$$g(x,y) = f(x,y) - h(x,y)$$

The difference is taken between all pairs of corresponding pixels from f(x,y) and h(x,y).



Standard (High kV)	Soft Tissue (Low kV)	Bone (Subtracted)

http://www3.gehealthcare.com.au/en-au/products/categories/radiography/advanced_applications/dual_energy_subtraction

Image Averaging

Consider a noisy image g(x,y) formed by the addition of noise n(x,y) to an original image

g(x,y) = f(x,y) +n(x,y)

Assumption: at each (x,y), the noise is un-correlated and has zero average mean.
If the noise satisfies these constraints, then, if an image G(x,y) is formed by averaging M different noisy images,

Then $$G(x,y) = \frac{1}{M}\sum_{i=1}^{M} g_i(x,y)$$

And $$E\{G(x,y)\} = f(x,y)$$

Where $$\sigma^2_{G(x,y)} = \frac{1}{M}\sigma^2_{n(x,y)}$$

$E\{G(x,y)\}$ is the expected value of G, and $\sigma^2_{G(x,y)}$ and $\sigma^2_{n(x,y)}$ are the variances of G and n at (x,y).

Image Averaging

The standard deviation at any point in the average image is

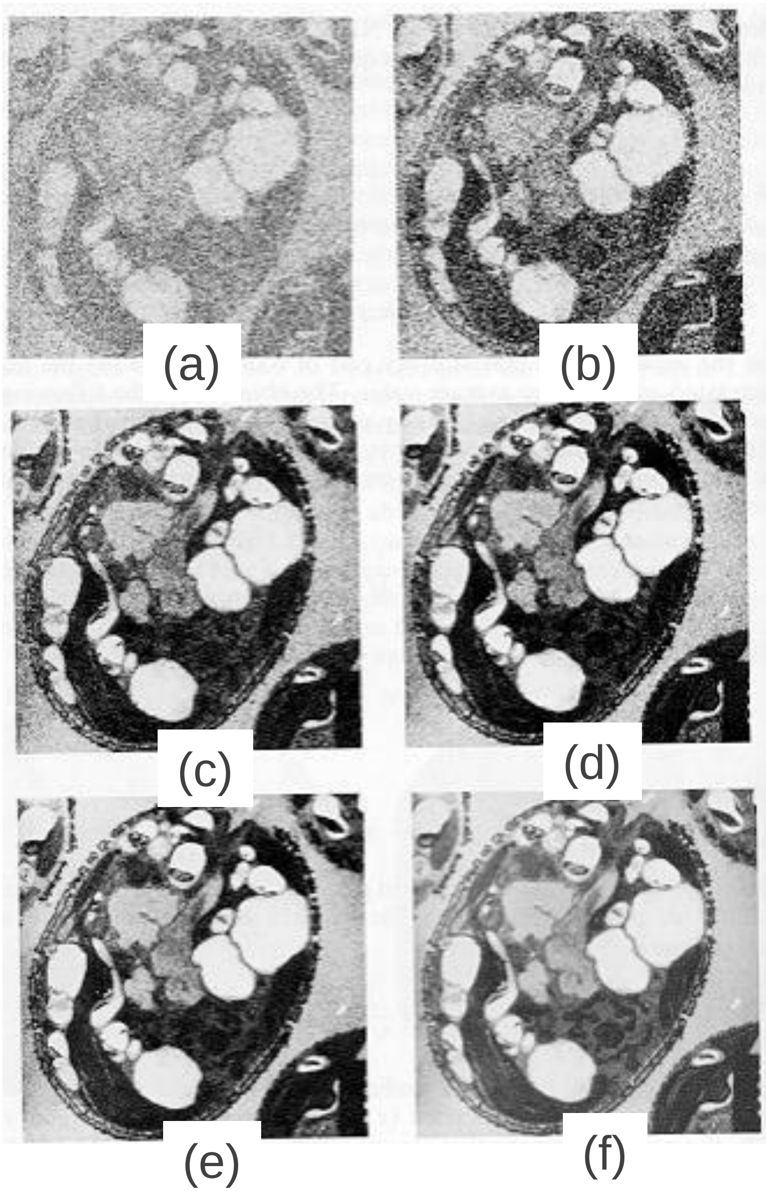
$$\sigma_{G(x,y)} = \frac{1}{\sqrt{M}} \sigma_{n(x,y)}$$

Hence, as M increases, the variability of the pixel values at each location (x,y) decreases

Because $E\{G(x,y)\} = f(x,y)$

$\ggg \quad G(x,y) \rightarrow f(x,y)$ as M increases.

Image Averaging



Chui *et. al.* ME5405 Machine Vision. NUS

Example of noise reduction by averaging (a) a typical noisy image; (b)-(f), results of averaging 2, 8, 16, 32, and 128 noisy images.

Spatial Filtering

Definition

- Spatial filtering refers to the use of spatial masks for image processing. The masks are called spatial filters.

Low-pass filters

- Eliminate high frequency components in the Fourier Domain while leaving low frequencies un-touched.
- High-frequency components characterize edges and other sharp details in an image. Hence, low-ass filters will cause image blurring.

High-pass filters

- Eliminate low frequencies. These filters has the effect of sharpening edges and other details.

Spatial Filtering

Application of Filters

The common approach is to sum products between the mask coefficients and the intensities of the pixels under the mask. This is presented as:

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

A 3 by 3 mask with arbitrary coefficients

$$R = w_1 z_1 + w_2 z_2 + \cdots + w_9 z_9$$

$$R = \sum_{i=1}^{9} w_i z_i$$

The center of the mask is placed at (x,y), the gray level of the pixel is then replaced by R.

The mask is then moved to the next pixel position and the process is repeated, until all the pixel locations in the image have been covered.

Spatial Filtering

Application of Filters

R is computed using partial neighborhood for pixels located at the border. A new image is usually created to store the new values of the pixels.

Examples of non-linear filters are

- median filtering for noise reduction;
- max filter: $R = \max \{z_k \mid k = 1, 2, ..., 9\}$ to find the brightest spot.
- min filter, $R = \min \{z_k \mid k = 1, 2, ..., 9\}$ to find the dimmest spot.

Spatial Filtering – Smoothing Filters

Smoothing Filters

Smoothing filters are used for blurring and for noise reduction. Blurring is used in pre-processing steps, such as:

- Removal of small details prior to large object extraction
- Bridging gaps in lines or curves
- Noise reduction

$$\frac{1}{9} \times \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\frac{1}{25} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$\frac{1}{49} \times \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

Spatial Filtering – Smoothing Filters

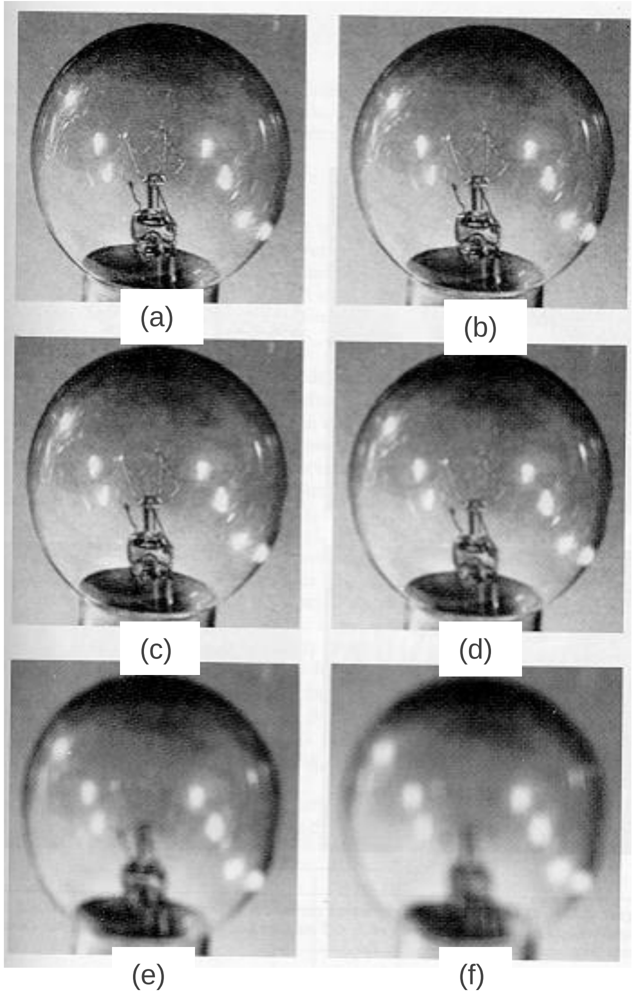
- This type of filter must have all positive coefficients.
- Division is necessary to prevent R from exceeding the valid gray level range.
- R is simply the average of all pixels in the area of the mask
- Also known as neighborhood averaging.

$$\frac{1}{9}\times\begin{array}{|c|c|c|}\hline 1&1&1\\\hline 1&1&1\\\hline 1&1&1\\\hline\end{array}$$

$$\frac{1}{25}\times\begin{array}{|c|c|c|c|c|}\hline 1&1&1&1&1\\\hline 1&1&1&1&1\\\hline 1&1&1&1&1\\\hline 1&1&1&1&1\\\hline 1&1&1&1&1\\\hline\end{array}$$

$$\frac{1}{49}\times\begin{array}{|c|c|c|c|c|c|c|}\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline 1&1&1&1&1&1&1\\\hline\end{array}$$

Spatial Filtering – Smoothing Filters



Chui *et. al.* ME5405 Machine Vision. NUS

(a) Original images, (b)-(f) results of low pass filtering with a mask of size nXn, where n=3, 5, 7, 15 and 25 respectively.

Spatial Filtering – Smoothing Filters

Median Filtering – Non-linear filter

- Low-pass filter has a limitation in which it blurs out sharp details while removing noise.
- Median filtering can remove noise, yet maintain the sharp details.
- In median filtering, gray level of each pixel is replaced by the median of its neighborhood.
- Effective in removing noise patterns consisting of strong and spike-like components while preserving sharp edges.

Spatial Filtering – Smoothing Filters

Steps:

1. Sort the values of the pixel and its neighbours
2. Determine the median
3. Assign this value to the pixel

Eg. In a 3 x 3 mask, the median is the 5^{th} largest value, while in a 5x 5 mask, it would be the 13^{th}.

(10, 20, 20, 20, 15, 20, 20, 25, 100)
After sorting would be (10, 15, 20, 20, 20, 20, 20, 25, 100)

The median is then 20 and the pixel is replaced by 20.

Spatial Filtering – Smoothing Filters

(a) (b) (c) (d)

Comparison of images enhanced by neighborhood averaging and by median filtering.

(a) Original image; (b) image corrupted by noise; (c) result of 5 x 5 averaging, (d) result of 5 x 5 median filtering

http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/OWENS/LECT5/node3.html

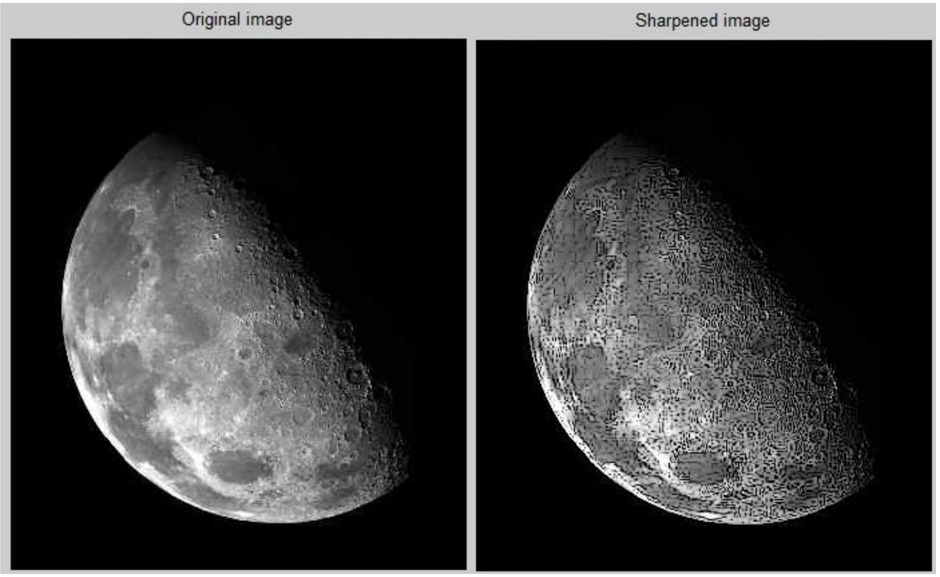
Spatial filtering – Sharpening Filter

Objective

To highlight fine details in an image or to enhance detail that has been blurred

Applications

Electronic printing, medical imaging, industrial application/inspection, and automatic target detection in smart weapons.



https://stackoverflow.com/questions/36688103/laplacian-image-filtering-and-sharpening-images-in-matlab

Spatial filtering – Sharpening Filter

Basic high-pass spatial filtering

- Filters should have positive coefficients near its center, and negative coefficients in the outer periphery.

- Note that the sum of coefficients is equal to zero. When the mask shown is over an area of constant or slow varying gray level, the output is zero or very small.

i.e.

$$R = \frac{1}{9}\sum_{i=1}^{9} w_i z_i \approx 0$$

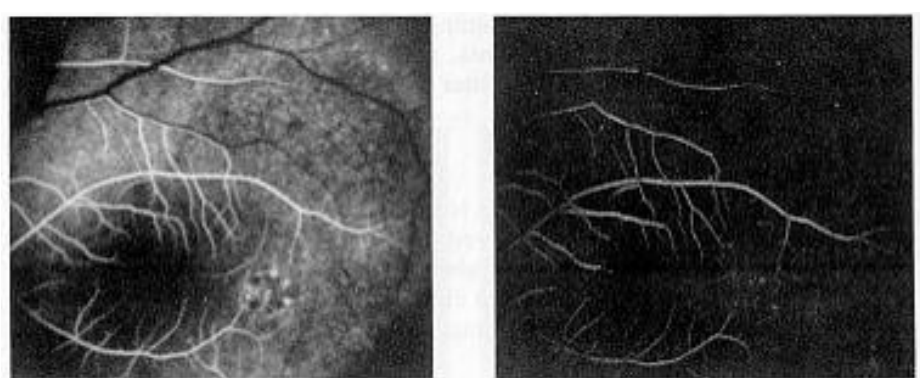
$\frac{1}{9} \times$

-1	-1	-1
-1	8	-1
-1	-1	-1

3 x 3 high pass spatial filter

Spatial filtering – Sharpening Filter

- Applying the above filter may yield negative gray-level values, hence clipping or scaling may be required to convert it to positive values
- Absolute values of results should not be taken as large negative gray-level value will give very bright spots in the image
- Application of high pass spatial filters will reduce average intensity in image to zero, thus <u>reducing contrast</u>.



(a) image of human retina

(b) result after spatial sharpening filter

Chui *et. al.* ME5405 Machine Vision. NUS

Spatial filtering – Sharpening Filter

High Boost Filtering

High-pass filtered image may be expressed as

High-pass = Original – Low-pass

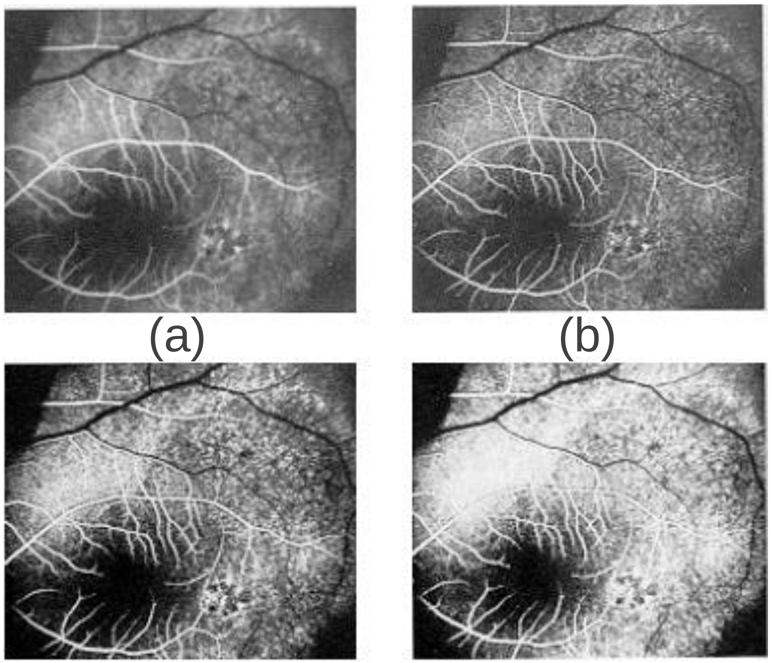
High-boost or High Frequency emphasis filter is given by

High boost = (A)(Original)-Low-pass
= (A-1)(Original)+Original – Low-pass
= (A-1)(Original) + High-pass

Where A is an amplification factor ≥ 1
When A = 1, standard high-pass results
When A > 1, part of the original is added back to the high-pass result

Spatial filtering – Sharpening Filter

- Amplification, A, has the effect of restoring partially the low-frequency components lost in the high-pass filtering operation.
- This gives an end result that looks more like the original image.
- Degree of edge enhancement is dependent on value of A



(a) (b) (c) (d)

Chui *et. al.* ME5405 Machine Vision. NUS

(a) original image; (b)-(d) result of high-boost filtering with A=1.1, 1.15 and 1.2 respectively.

(b) Shows significant improvement over pure high-pass filtering. But at A=1.2, the image (d) is at the verge of being unacceptable

Spatial filtering – Sharpening Filter

In term of implementation, the high-boost filter can be achieved by letting the central pixel of the mask shown below to be

$$w = 9A - 1$$

with $A \geq 1$. A determines the nature of the filter

$\frac{1}{9} \times$

-1	-1	-1
-1	w	-1
-1	-1	-1

High-boost filter mask with $w=9A-1$ with $A\geq1$

Spatial filtering – Sharpening Filter

Derivative Filters

- Image averaging tends to blur details.
- Since averaging is analogous to integration, differentiation have the opposite effect and sharpen an image.
- The most common method of differentiation in image processing application is gradient.

For a function f(x,y), the gradient of coordinates (x,y) is defined as the vector

$$\nabla\vec{f} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

The magnitude

$$\nabla f = mag(\nabla f) = \left[\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2\right]^{\frac{1}{2}}$$

Spatial filtering – Sharpening Filter

Derivative Filters

It can be applied at z_5 (center pixel of a 3x3 mask) in a number of ways.

The simplest is

$$\nabla f = [(z_5 - z_8)^2 + (z_5 - z_6)^2]^{\frac{1}{2}}$$

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

or by absolute values

$$\nabla f = |z_5 - z_8| + |z_5 - z_6|$$

Using cross differences:

$$\nabla f \approx [(z_5 - z_9)^2 + (z_6 - z_8)^2]^{\frac{1}{2}}$$

or

$$\nabla f \approx |z_5 - z_9| + |z_6 - z_8|$$

Spatial filtering – Sharpening Filter

Different Types of Derivative Filters

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

(a)

1	0
0	-1

0	1
-1	0

(b) Roberts

-1	-1	-1
0	0	0
1	1	1

-1	0	1
-1	0	1
-1	0	1

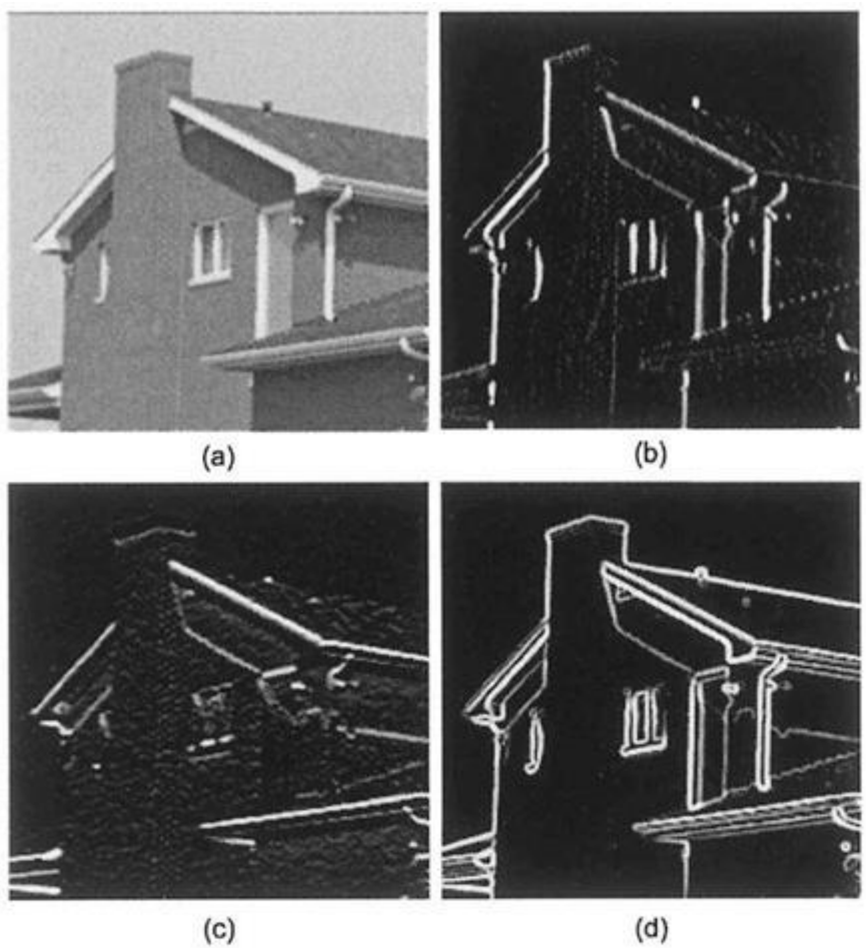
(c) Prewitt

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

(d) Sobel

Spatial filtering – Sharpening Filter



http://what-when-how.com/embedded-image-processing-on-the-tms320c6000-dsp/edge-detection-image-processing-part-1/

On the left shows 2D edge enhancement using the Sobel filters.

(a) Original house image

(b) Gx image (vertical edges)

(c) Gy image (horizontal edges)

(d) Combined edges

Image Segmentation

Introduction

Introduction

Segmentation:
- Is First step in image analysis
- Sub-divides an image into areas or regions
- Results in isolating objects of interest

Example:

In air-to-ground target acquisition, vehicles are the objects of interest.

Segmentation algorithms for monochrome images generally are based on one of the two basic properties of gray-level values:

- Discontinuity and Similarity

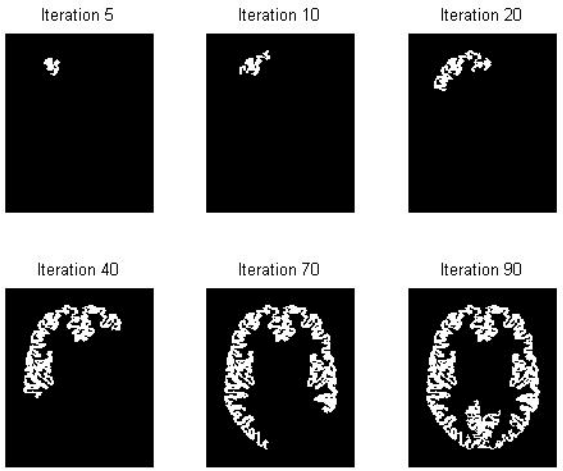
Introduction

Discontinuity

- Segment an image based on abrupt changes in gray level.
- The areas of interest are the detection of isolated points, lines and edges in an image.

Similarity

- Based on thresholding, region growing, and region merging and splitting.



https://it.mathworks.com/matlabcentral/fileexchange/35269-simple-single-seeded-region-growing

Segmentation

Detection of Discontinuities

The detection of three basic types of discontinuities in a digital image: points, lines and edges.

The common way is to run a mask shown through the image. If z_i's denote the gray-level values of the pixels under the mask , the response of the mask at any point in the image is:

w_1	w_2	w_3
w_4	w_5	w_6
w_7	w_8	w_9

$$R = w_1 z_1 + w_2 z_2 + \cdots + w_9 z_9$$

$$R = \sum_{i=1}^{9} w_i z_i$$

R is defined with respect to the central position of the mask

Point detection

Using the mask shown below, a point is considered isolated at the location on which the mask is centered if

$$|R| > T$$

-1	-1	-1
-1	8	-1
-1	-1	-1

Where T is a non-negative threshold

With the coefficients on the mask, the gray level of an isolated point will be different from its neighbors after computation.

Segmentation

Line Detection (Using Line Masks)

-1	-1	-1
2	2	2
-1	-1	-1

(a) Horizontal

-1	-1	2
-1	2	-1
2	-1	-1

(b) +45^{o}

-1	2	-1
-1	2	-1
-1	2	-1

(c) Vertical

2	-1	-1
-1	2	-1
-1	-1	2

(d) -45^{o}

If mask (a) is moved around an image, it would respond more strongly to line (one pixel thick) oriented horizontally.

1	1	1	1	1
5	5	5	5	5
1	1	1	1	1

R in this row will be high

Segmentation

Line detection (Using Line Masks)

Mask:

(b) Responds best to lines oriented at 45º
(c) For vertical lines
(d) For lines oriented at -45º

Let Ra, Rb, Rc and Rd denote the responses of the masks (a – d), from left to right.

All masks are run through an image. If at certain point in the image, |Ri|>|Rj|, for all i≠j, that point is said to be more likely to associate with a line in the direction of mask i.

Example, if |Rb|>|Rj| for j= a, c, d.
That point is more likely to be associated with a 45º line.

Edge Detection

An edge is the boundary between two regions with relatively distinct gray level properties.

Assumption: regions are sufficiently homogenous so that transition between two regions can be determined on basis of gray-level discontinuity alone.

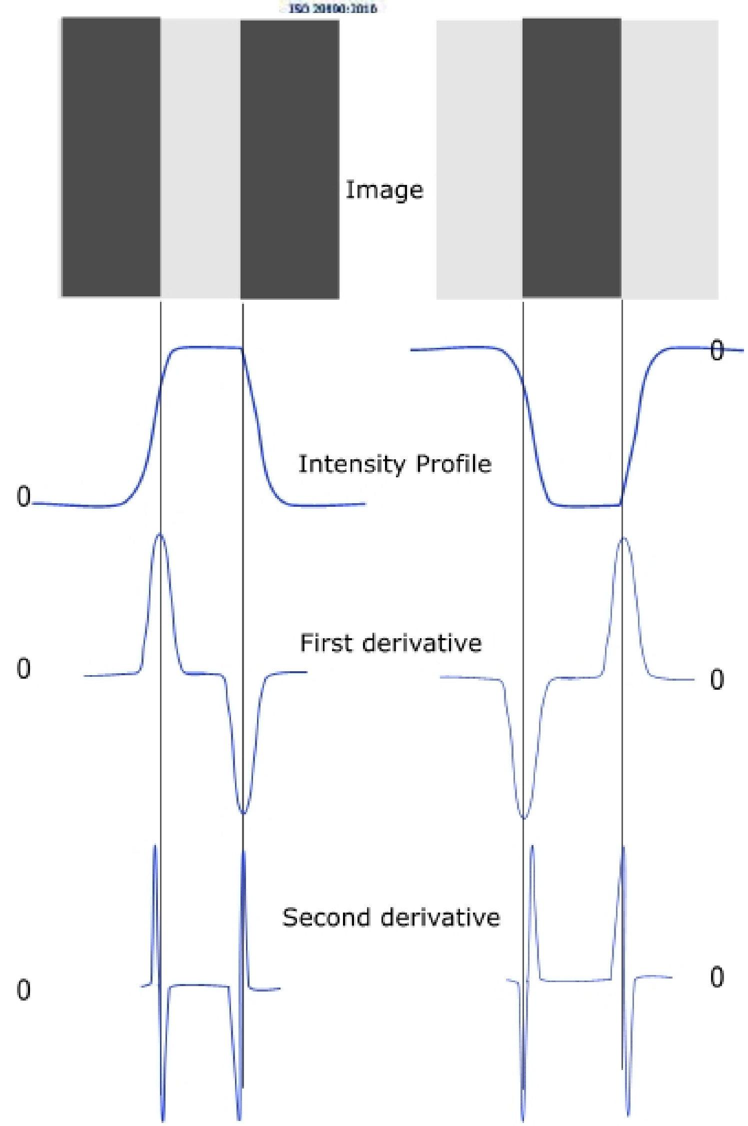
https://ch.mathworks.com/fr/discovery/edge-detection.html

Segmentation

Edge Detection

Fgure shows a light strip on a dark background

- Profile of gray-level at edge is modelled as a smooth change of gray level.
- First derivative of the gray level profile is positive at leading edge of transition, negative at edge and zero in areas of constant gray level.
- The second derivative is positive for the transition part associated with the dark side of the edge, negative for the transition with the light side of the edge



https://mipav.cit.nih.gov/pubwiki/index.php/Edge_Detection:_Zero_X_Laplacian

Segmentation

Gradient Operators

The gradient of an image f(x, y) at location (x,y) is given by:

$$\nabla\vec{f} = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix}$$

The gradient vector points in the direction of maximum rate of change.
In edge detection, an important quantity is the magnitude of $\nabla\vec{f}$, which is referred to simply as gradient, where

$$\nabla f = mag(\nabla\vec{f}) = [G_x^2 + G_y^2]^{\frac{1}{2}}$$

$mag(\nabla\vec{f})$= maximum rate of increase of f(x,y) per unit distance in the direction of $\nabla\vec{f}$

Segmentation

Gradient Operators

Gradient is commonly approximated by the absolute values:

$$\nabla f \approx |G_x| + |G_y|$$

Direction of $\nabla \vec{f}$, at (x, y)

$$\alpha(x, y) = \tan^{-1}\left(\frac{G_y}{G_x}\right)$$

Where angle is measured with respect to the x-axis.

Segmentation

Gradient Operators – Sobel Operator

Sobel operators have the advantage of providing both a differencing and a smoothing effect

z_1	z_2	z_3
z_4	z_5	z_6
z_7	z_8	z_9

-1	-2	-1
0	0	0
1	2	1

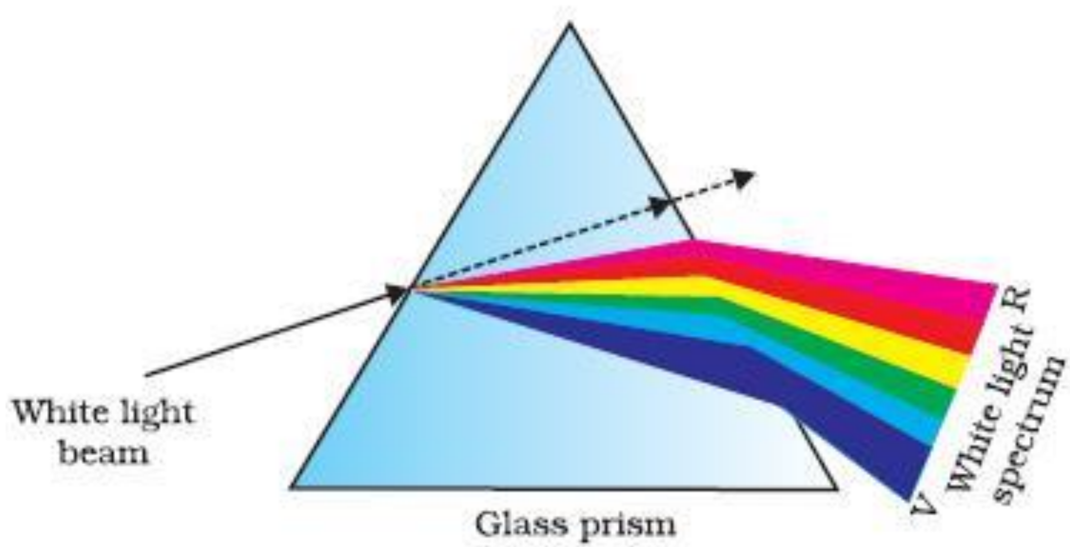
Compute Gx at center point of 3 x 3 region

-1	0	1
-2	0	2
-1	0	1

compute G_y at center point of 3 x 3 region

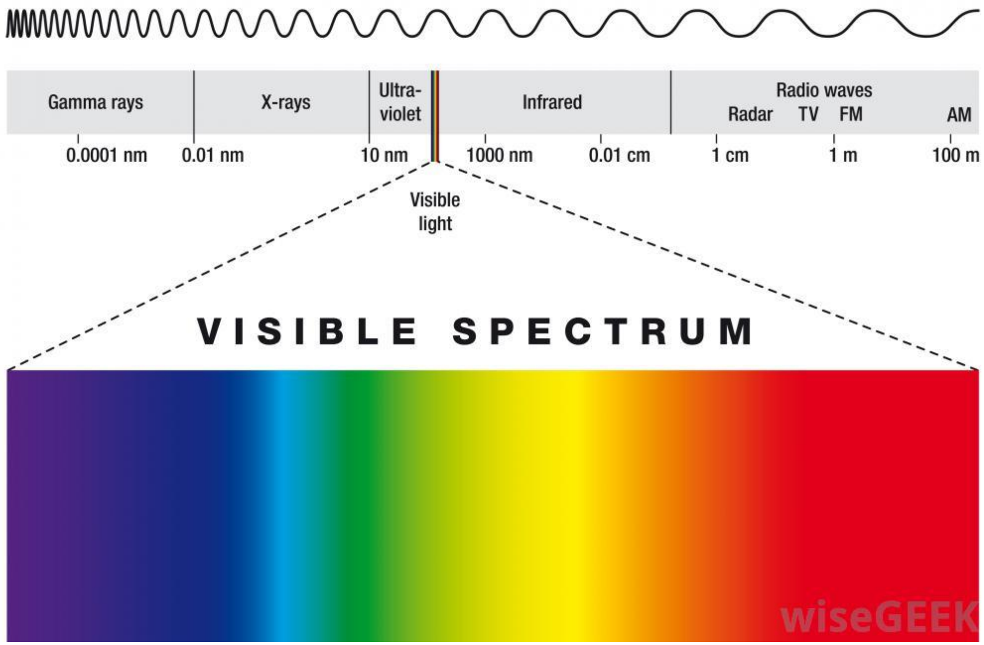
Color Image Processing

Spectrum of White Light



https://www.toppr.com/guides/science/light/sunlight-white-or-coloured/

Electromagnetic Spectrum



https://www.wisegeek.com/what-is-white-light.htm

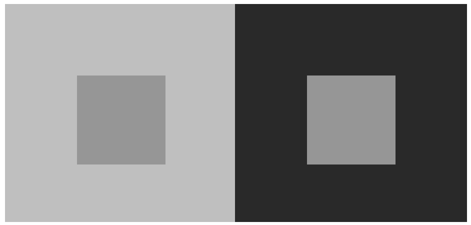
Visible light wavelength: from around 400 to 700 nm

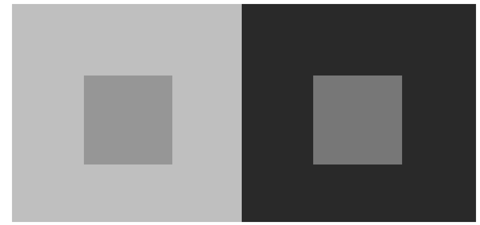
1. For an achromatic (monochrome) light source, there is only 1 attribute to describe the quality: intensity
2. For a chromatic light source, there are 3 attributes to describe the quality:

Radiance = total amount of energy flow from a light source (Watts)
Luminance = amount of energy received by an observer (lumens)
Brightness = intensity

Luminance vs. Brightness

Same lum.
Different brightness



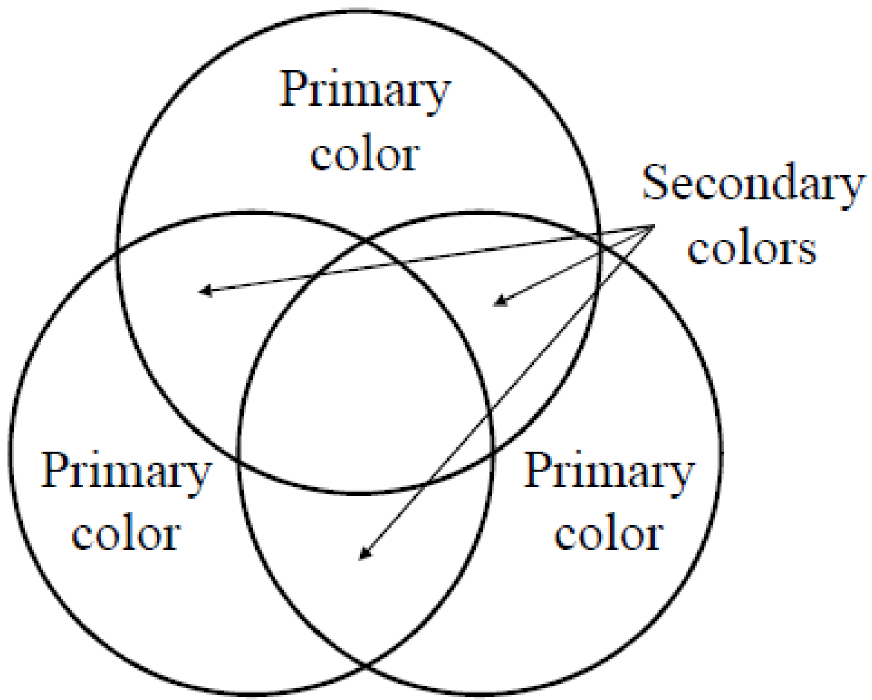


Different lum.
Similar brightness

http://slideplayer.com/slide/7698593/release/woothee

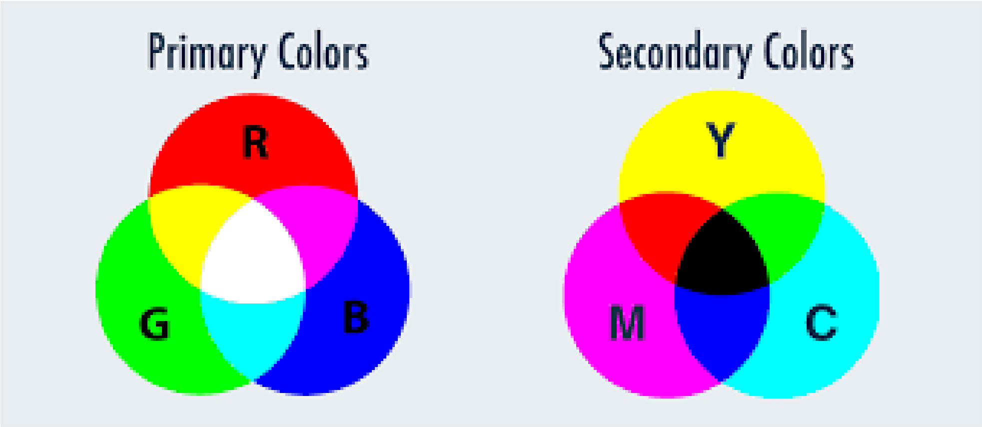
- Luminance (or intensity)
  - Independent of the luminance of surroundings
- Brightness
  - Perceived luminance
  - Depends on surrounding luminance

Primary and Secondary Colors



Primary and Secondary Colors

Additive | Subtractive



https://visme.co/blog/infographic-color-schemes/

Additive primary colours: RGB use in the case of light sources such as color monitors

Subtractive primary colours: CMY use in the case of pigments in printing devices

RGB add together to get white

White subtracted by CMY to get Black

Color Characterization

Hue: dominant color corresponding to a dominant wavelength of mixture light wave

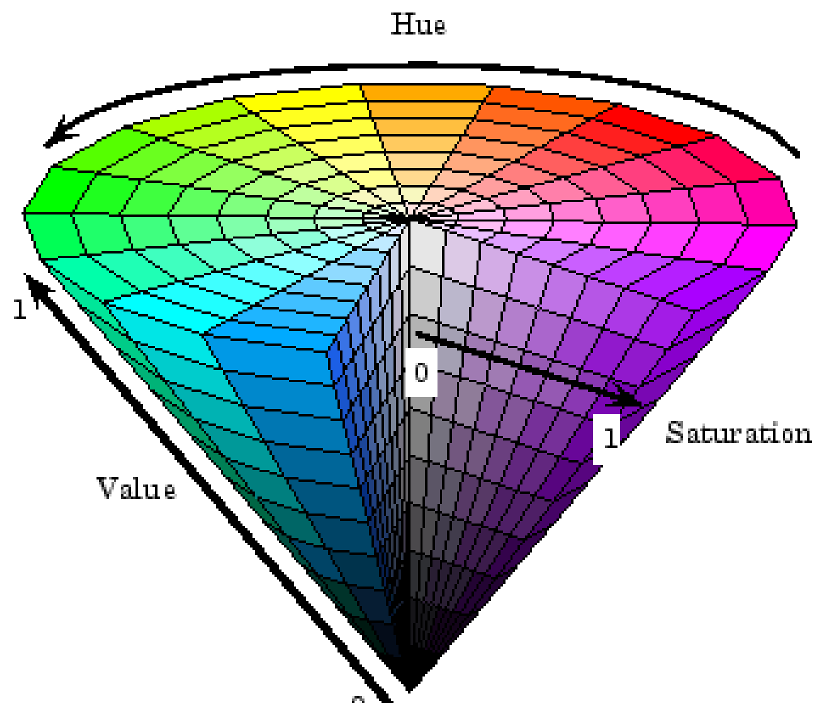
Saturation: Relative purity or amount of white light mixed with a hue (inversely proportional to amount of white light added)

Brightness: Intensity

$$\left.\begin{array}{l}\bullet\ \text{Hue}\\ \bullet\ \text{Saturation}\end{array}\right\}\ \text{Chromaticity}$$

Perceptual Attributes of Color

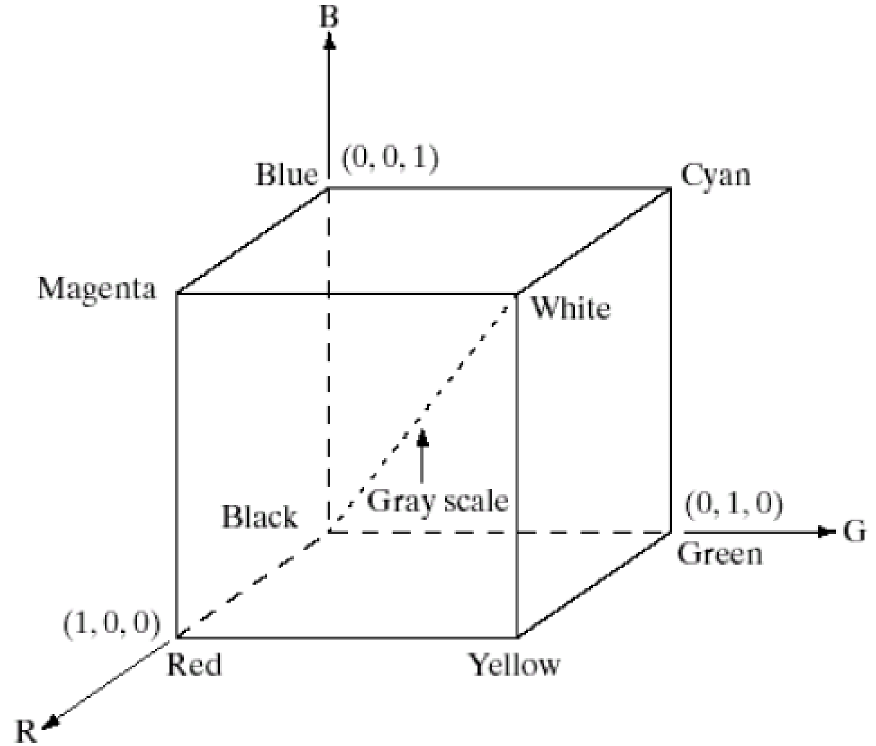
- Value of Brightness (perceived luminance)
- Chrominance
  - Hue
    - specify color tone (redness, greenness, etc.)
    - depend on peak wavelength
  - Saturation
    - describe how pure the color is
    - depend on the spread (bandwidth) of light spectrum
    - reflect how much white light is added
- RGB - HSV Conversion is nonlinear



https://www.mathworks.com/help/images/convert-from-hsv-to-rgb-color-space.html

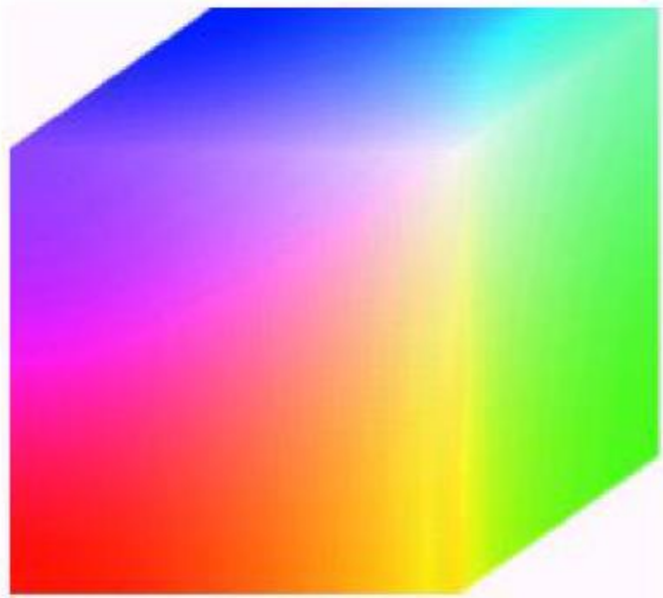
RGB Color Model

- Purpose of color models: to facilitate the specification of colors in some standard
- RGB color models:
  - based on Cartesian coordinate system



https://www.researchgate.net/publication/228677646_Case_study_in_effects_of_color_spaces_for_mineral_identification/figures?lo=1&utm_source=google&utm_medium=organic

RGB Color Cube



$$\left.\begin{array}{l} R = 8 \text{ bits} \\ G = 8 \text{ bits} \\ B = 8 \text{ bits} \end{array}\right\} \begin{array}{l} \text{Color depth 24 bits} \\ = 16777216 \text{ colors} \end{array}$$

$(R = 0)$ $(G = 0)$ $(B = 0)$

Hidden faces of the cube

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

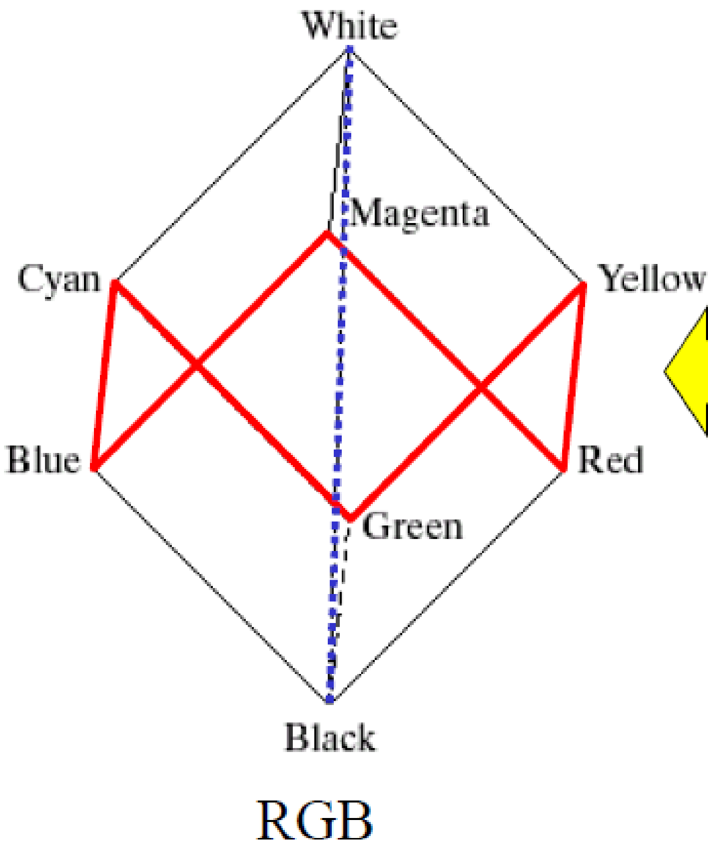
HSI Color Model

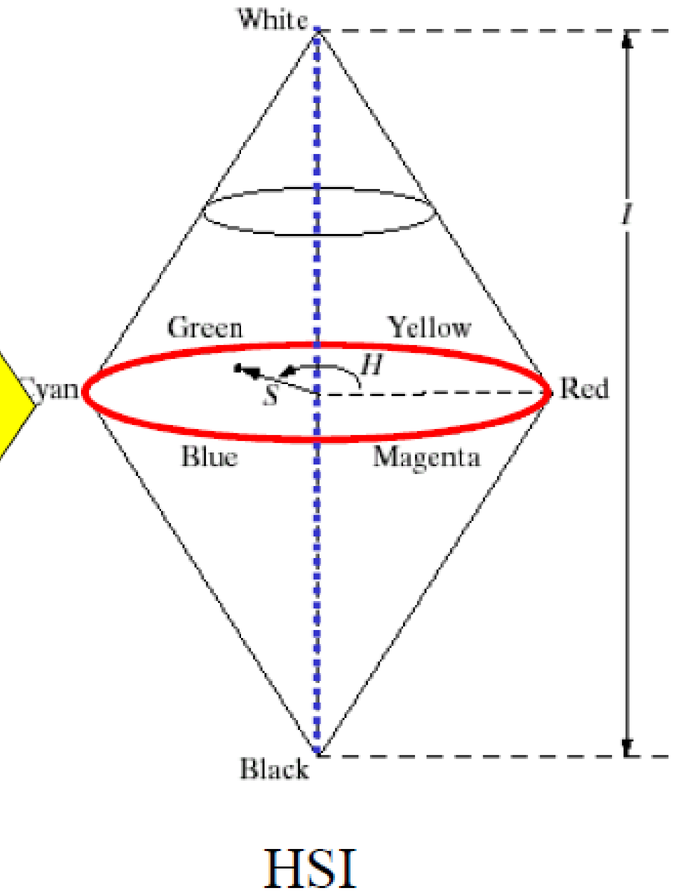
RGB, CMY models are not good for human interpreting

- HSI Color model:
  - Hue: Dominant color
  - Saturation: Relative purity (inversely proportional to amount of white light added)
  - Intensity: Brightness

Information to humans

Relationship Between RGB and HSI Color Models

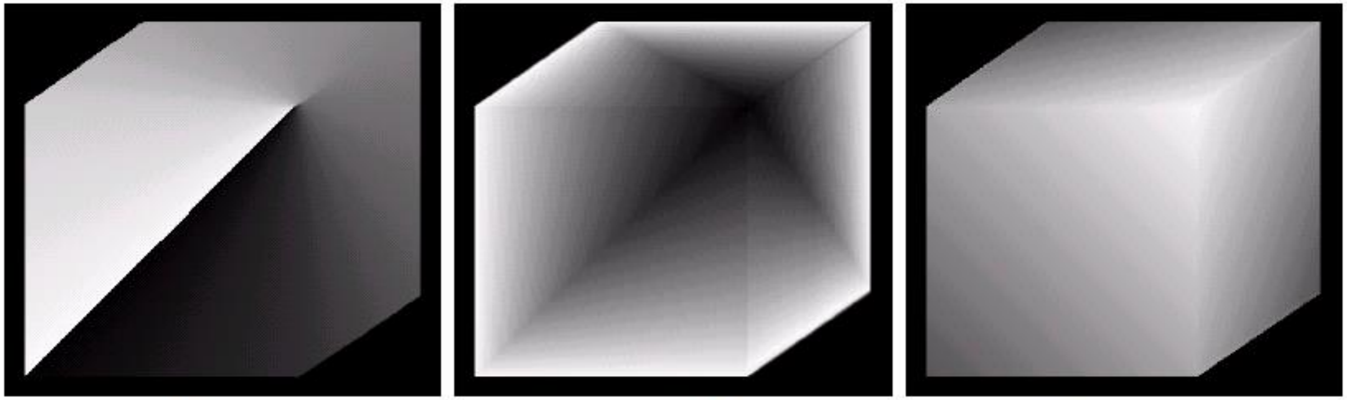




(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Example: HSI Components of RGB Cube

RGB Cube



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Converting Co Colors from RGB to HSI

$$H = \begin{cases} \theta & \text{if } B \le G \\ 360 - \theta & \text{if } B > G \end{cases}$$

$$\theta = \cos^{-1}\left\{ \frac{\frac{1}{2}\left[(R-G)+(R-B)\right]}{\left[(R-G)^2+(R-B)(G-B)\right]^{1/2}} \right\}$$

$$S = 1 - \frac{3}{R+G+B}$$

$$I = \frac{1}{3}(R+G+B)$$

Converting Co Colors from HSI to RGB

RG sector: $0 \leq H < 120$

$$R = I\left[1 + \frac{S\cos H}{\cos(60^{\circ} - H)}\right]$$

$$B = I(1 - S)$$

$$G = 1 - (R + B)$$

GB sector: $120 \leq H < 240$

$$H = H - 120$$

$$R = I(1 - S)$$

$$G = I\left[1 + \frac{S\cos H}{\cos(60^{\circ} - H)}\right]$$

$$B = 1 - (R + G)$$

BR sector: $240 \leq H \leq 360$

$$H = H - 240$$

$$B = I\left[1 + \frac{S\cos H}{\cos(60^{\circ} - H)}\right]$$

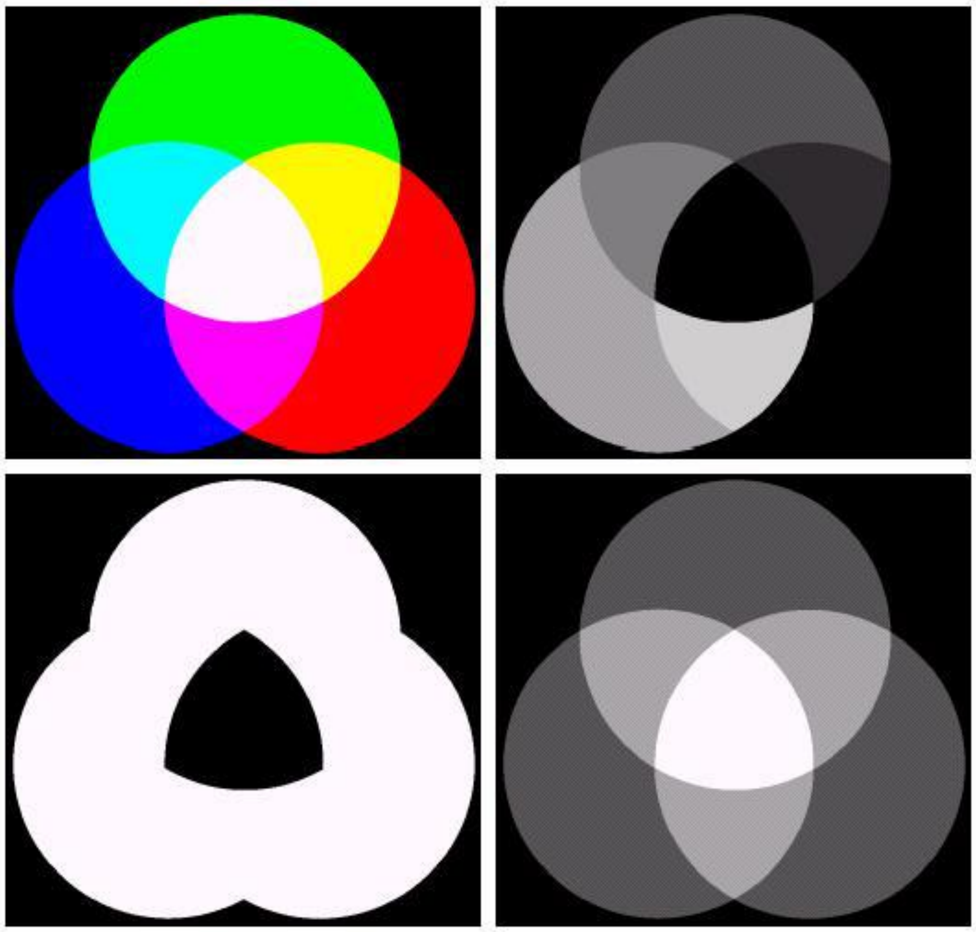
$$G = I(1 - S)$$

$$R = 1 - (G + B)$$

Example: HSI Components of RGB Colors

RGB Image

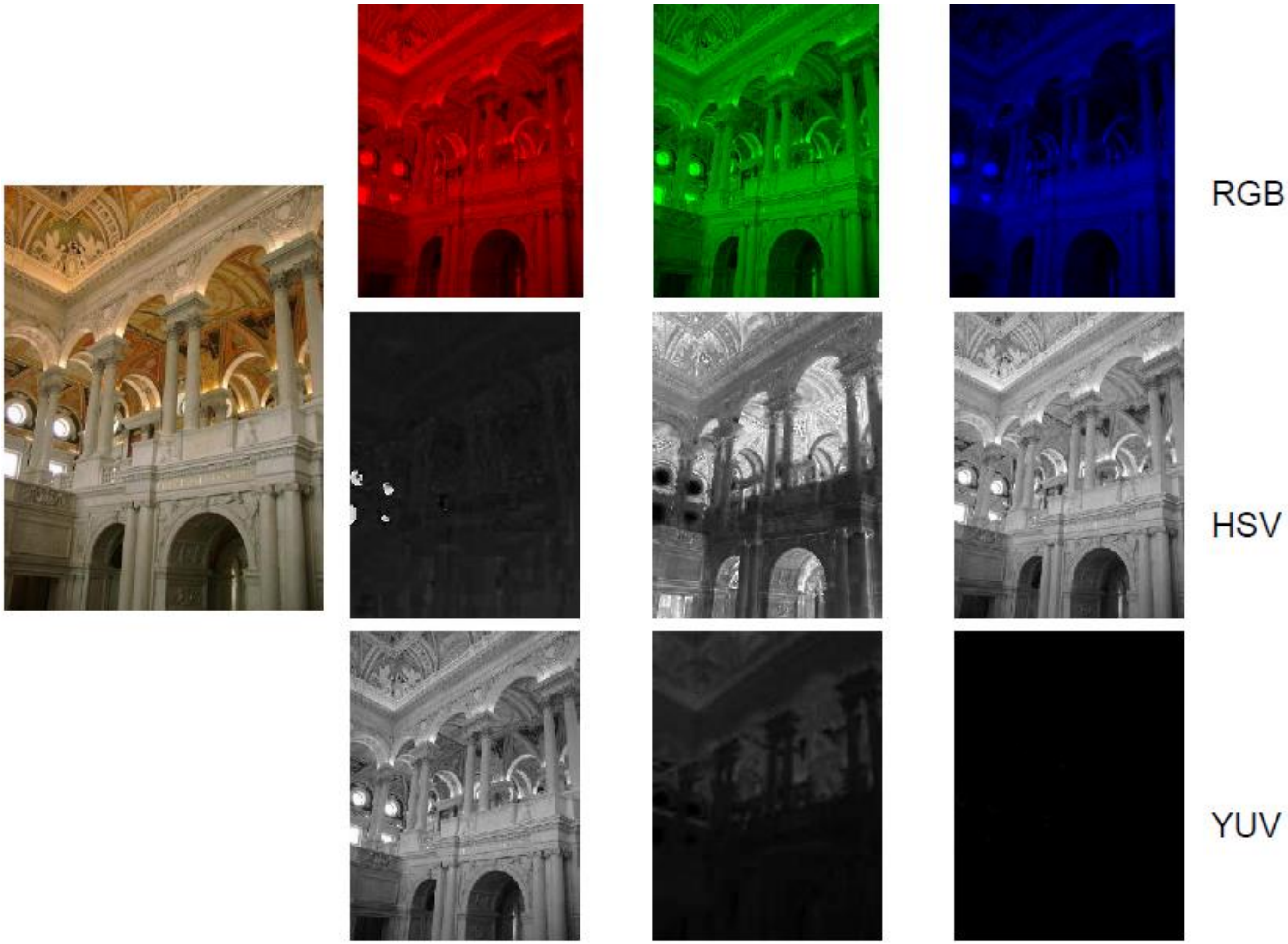
Hue



Saturation

Intensity

Examples



Color Image Processing

- There are 2 types of color image processes:

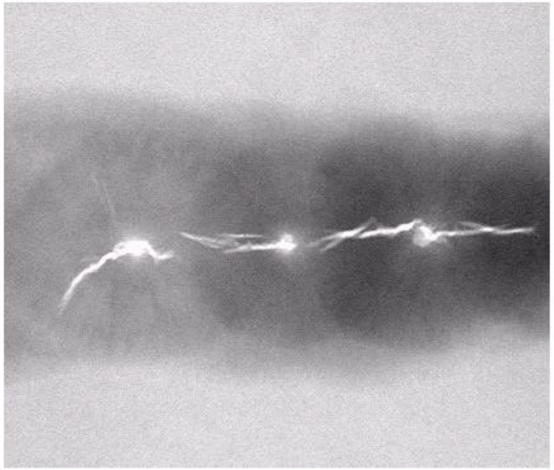
  - Pseudo-color image process: Assigning colors to gray values based on a specific criterion.

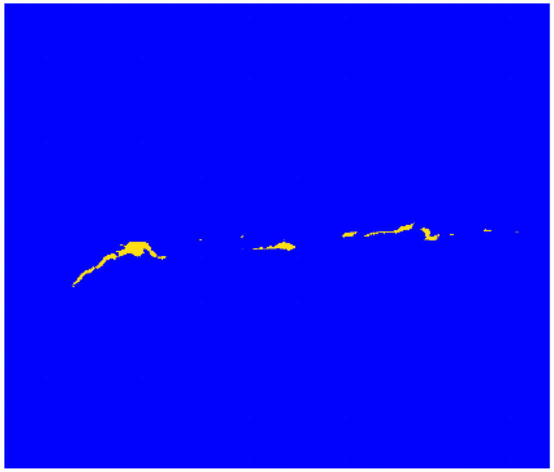
  - Full color image process: Manipulation of real color images such as color photographs.

Pseudo-color Image Processing

- Pseudo color = false color : In some case there is no “color” concept for a gray scale image but we can assign “false” colors to an image.

- Why we need to assign colors to gray scale image?
- Answer: Human can distinguish different colors better than different shades of gray.



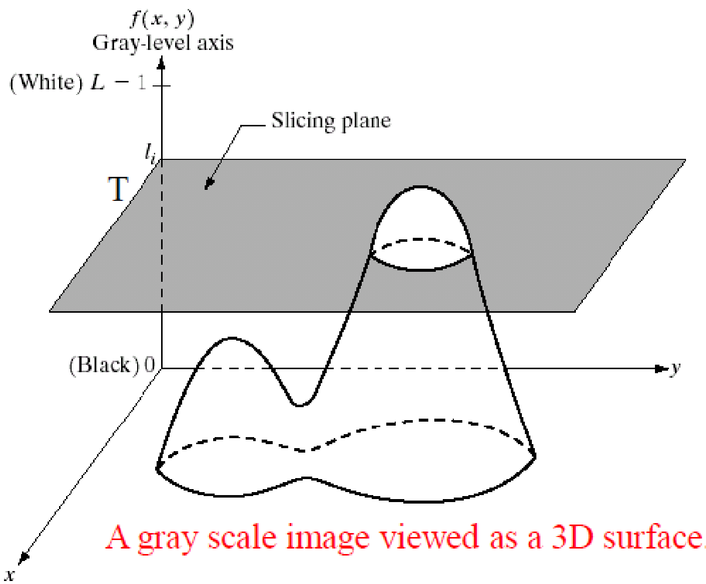


(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Intensity Slicing or Density Slicing

If C_1 is Color 1 and C_2 is Color 2

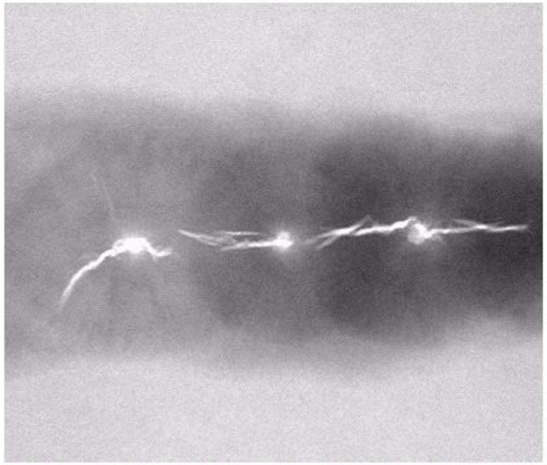
$$g(x,y) = \begin{cases} C_1 & \text{if } f(x, y) \leq T \\ C_2 & \text{if } f(x, y) > T \end{cases}$$



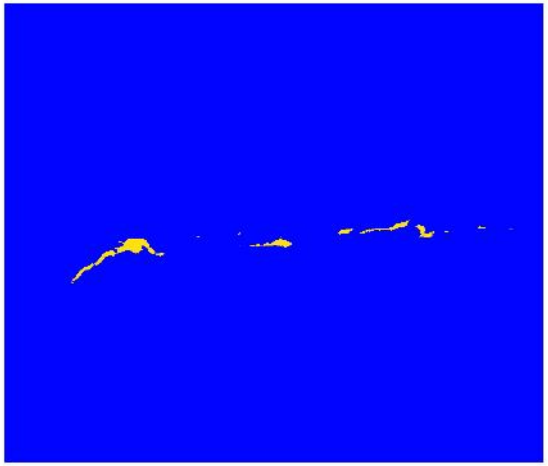
A gray scale image viewed as a 3D surface.

https://www.slideshare.net/lineking/10-color-image-processing-dip

Intensity Slicing Example



An X-ray image of a weld with cracks



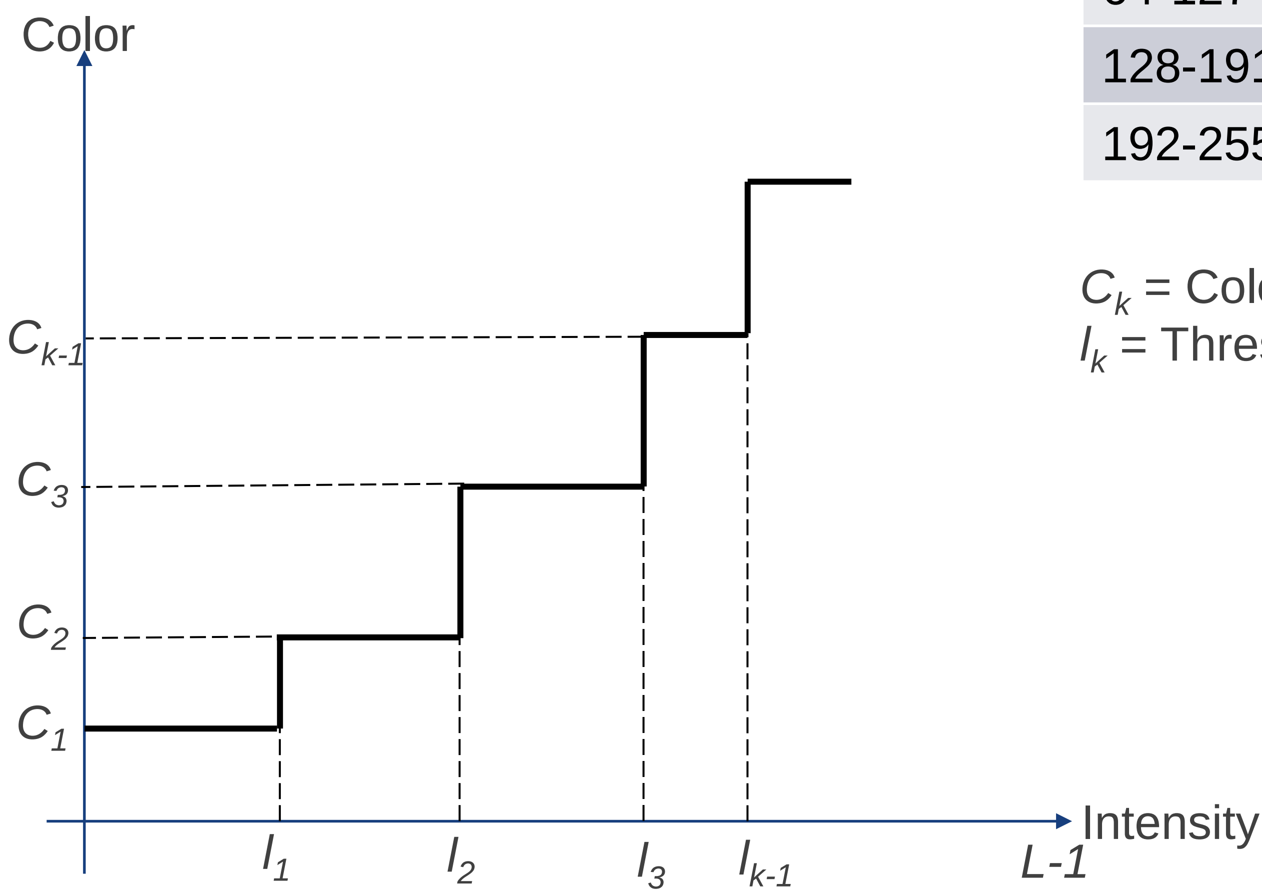
After assigning a yellow color to pixels with value 255 and a blue color to all other pixels.

https://slideplayer.com/slide/7577824/

Multi Level Intensity Slicing

$g(x, y) = C_k$ for $l_{k-1} < f(x, y) \leq l_k$

Grey Level	Color
0-63	Blue
64-127	Magneta
128-191	Green
192-255	Red

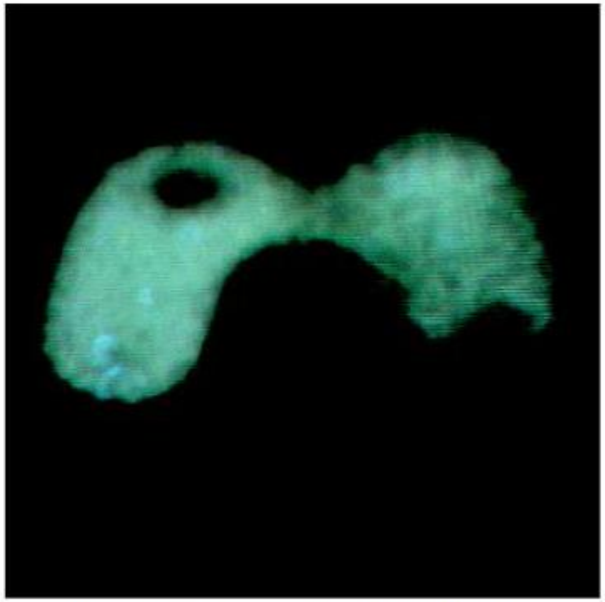


C_k = Color No. k
l_k = Threshold level k

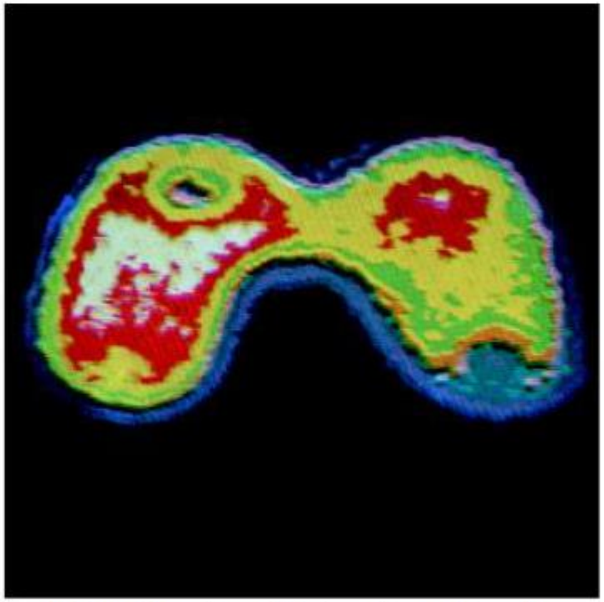
Multi Level Intensity Slicing Example

$$g(x, y) = C_k \quad \text{for } l_{k-1} < f(x, y) \le l_k$$

C_k = Color No. k
l_k = Threshold level k



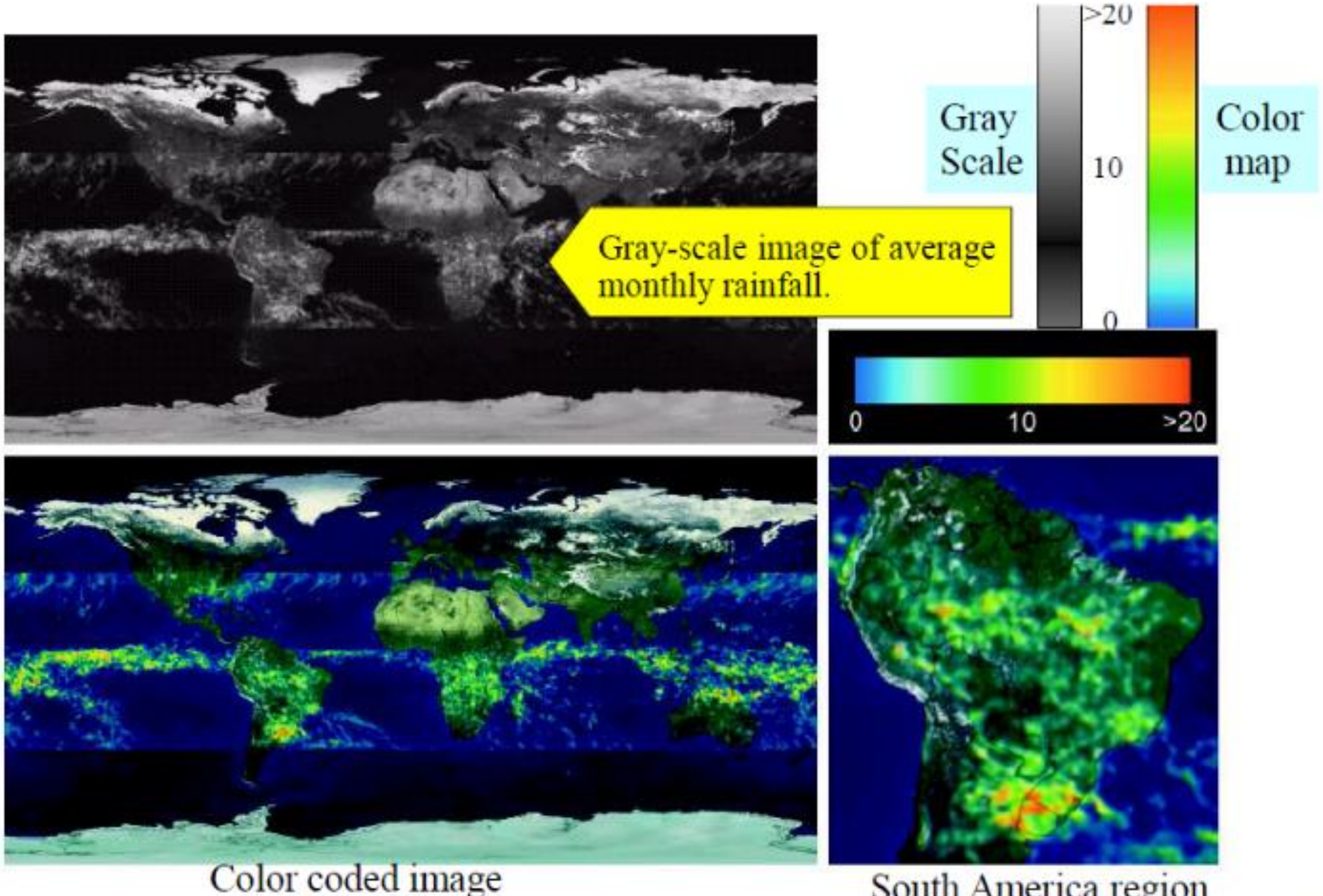
An X-ray image of the Picker Thyroid Phantom.



After density slicing into 8 colors

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

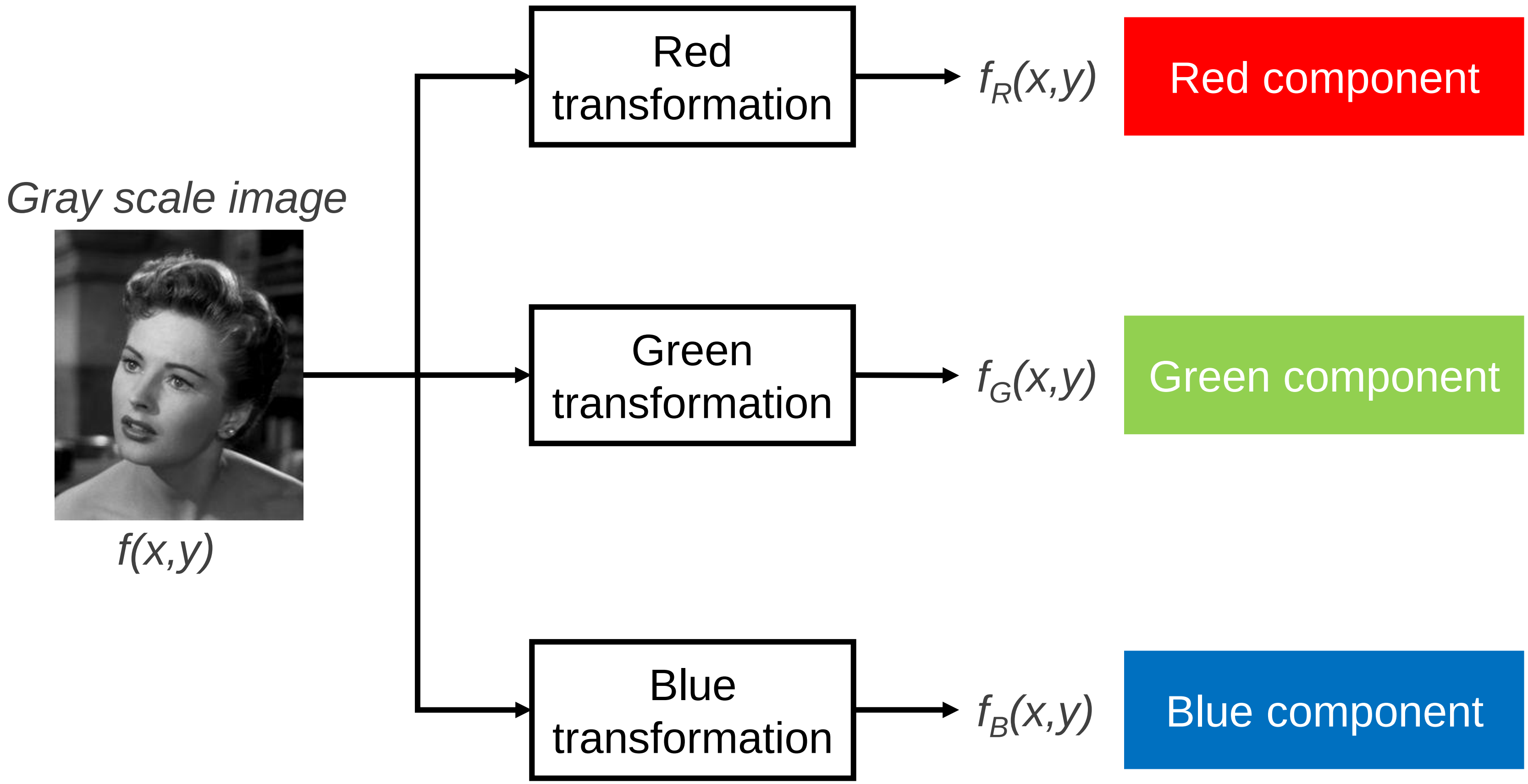
Color Coding Example



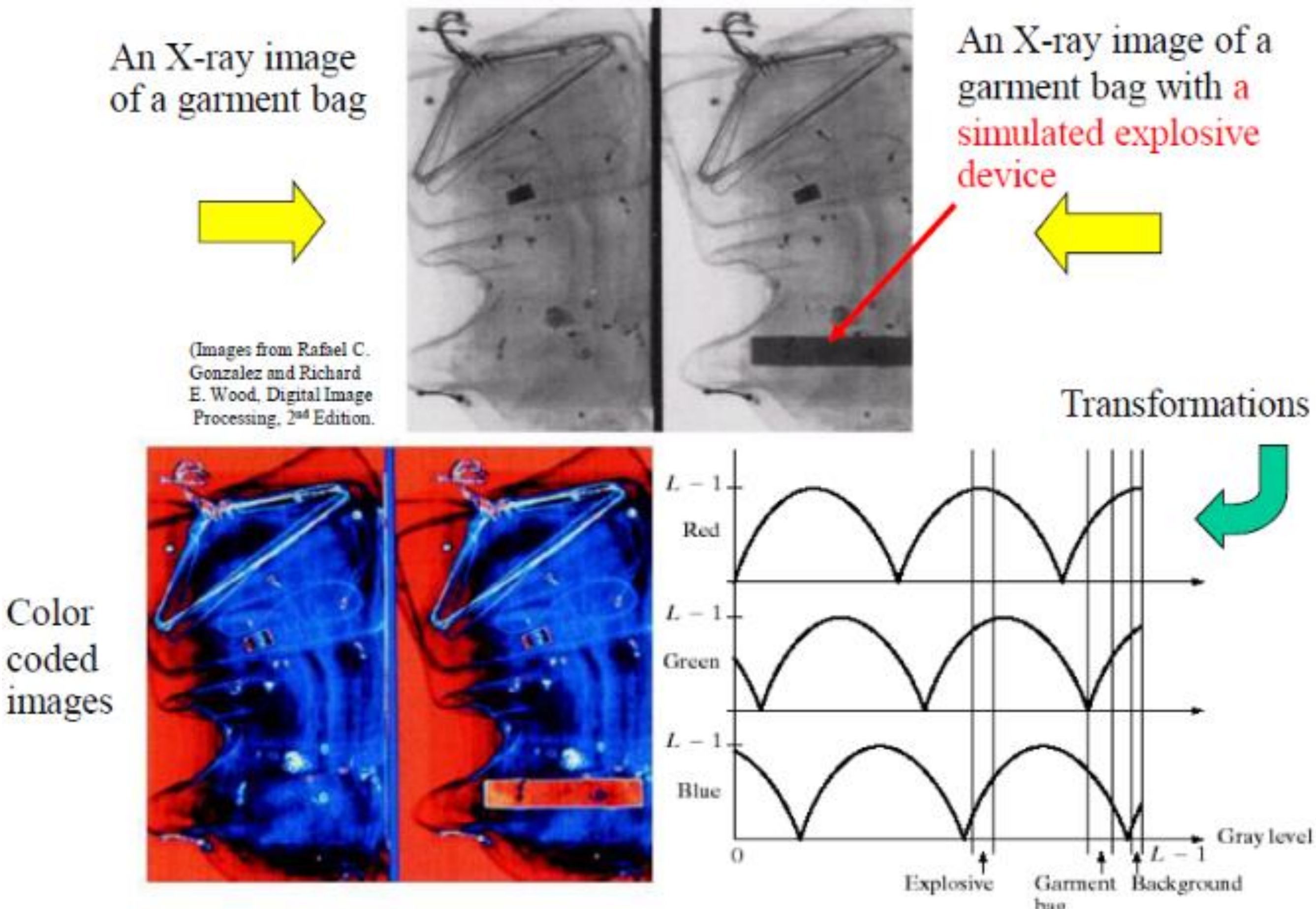
(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Gray Level to Color Transformation

- To assign colors to gray levels based on specific mapping functions



Gray Level to Color Transformation Example



https://slideplayer.com/slide/7577824/

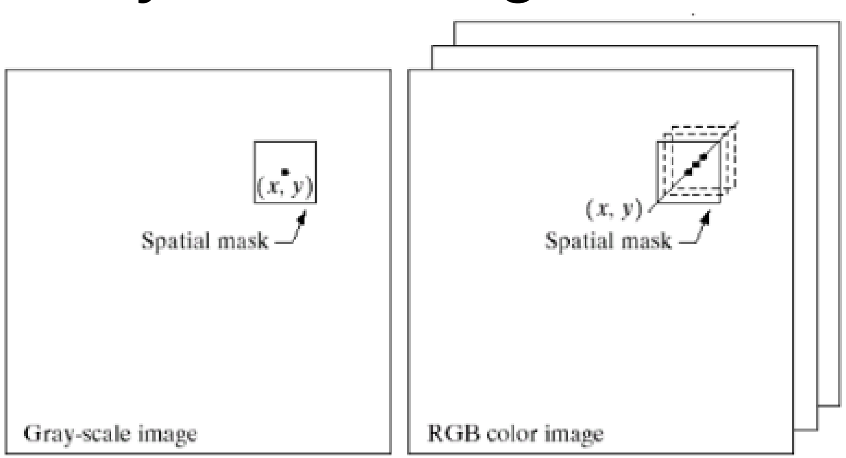
Basics of Full-Color Image Processing

2 Methods:

- Per-color-component processing: process each component separately.
- Vector processing: treat each pixel as a vector to be processed.

Examples of per-color-component processing: smoothing or sharpening an image

By smoothing each RGB component separately.



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2 Edition.

Color Transformation

- The treatment of colors
  - Transforming one color state to another color state

- Equation:

$$g(x, y) = T[f(x, y)]$$

where $f(x, y)$ = input color image, $g(x, y)$ = output color image
T = operation on f over a neighborhood of (x, y)

- Processing is done pixel by pixel, using:

$$s_i = T_i(r_1, r_2, \ldots, r_n) \qquad i = 1, 2, \ldots n$$

where r_i = color component of $f(x, y)$
where s_i = color component of $g(x, y)$

- In RGB images, $n = 3$

Example: Color Transformation

RGB Transformation:

$$s_R(x,y) = kr_R(x,y)$$

$$s_G(x,y) = kr_G(x,y)$$

$$s_B(x,y) = kr_B(x,y)$$

HSI Transformation:

$$s_I(x,y) = kr_I(x,y)$$

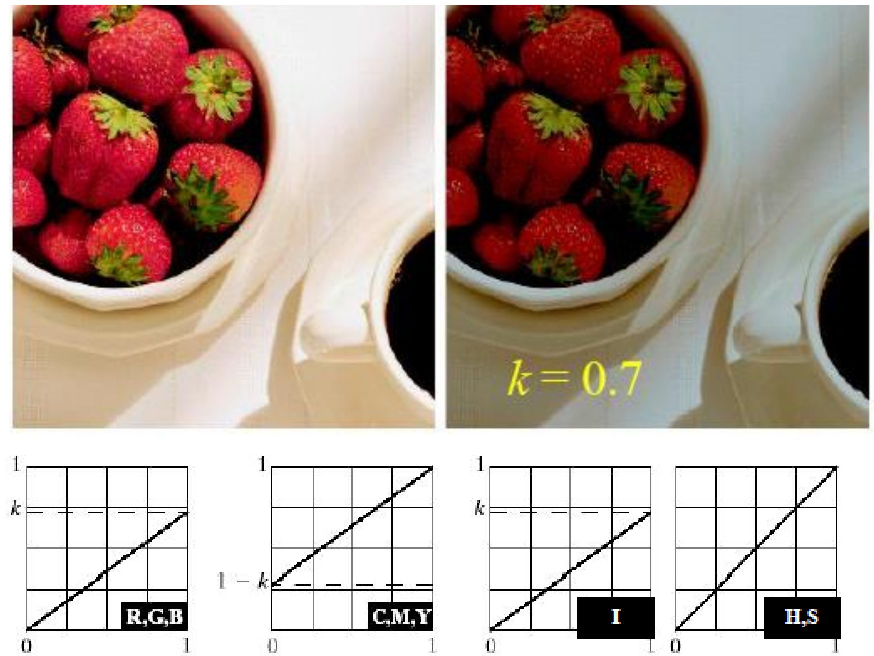
CMY Transformation:

$$s_C(x,y) = kr_C(x,y) + (1-k)$$

$$s_M(x,y) = kr_M(x,y) + (1-k)$$

$$s_Y(x,y) = kr_Y(x,y) + (1-k)$$

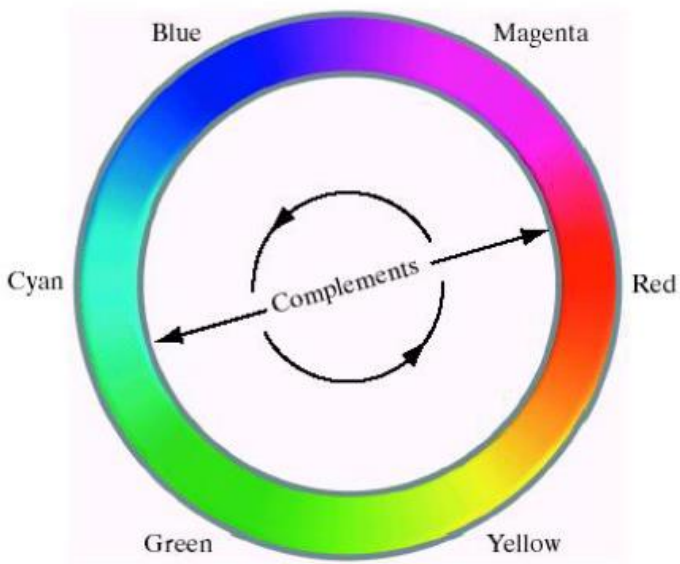
The above 3 transformation will yield similar results



Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition

Color Complements

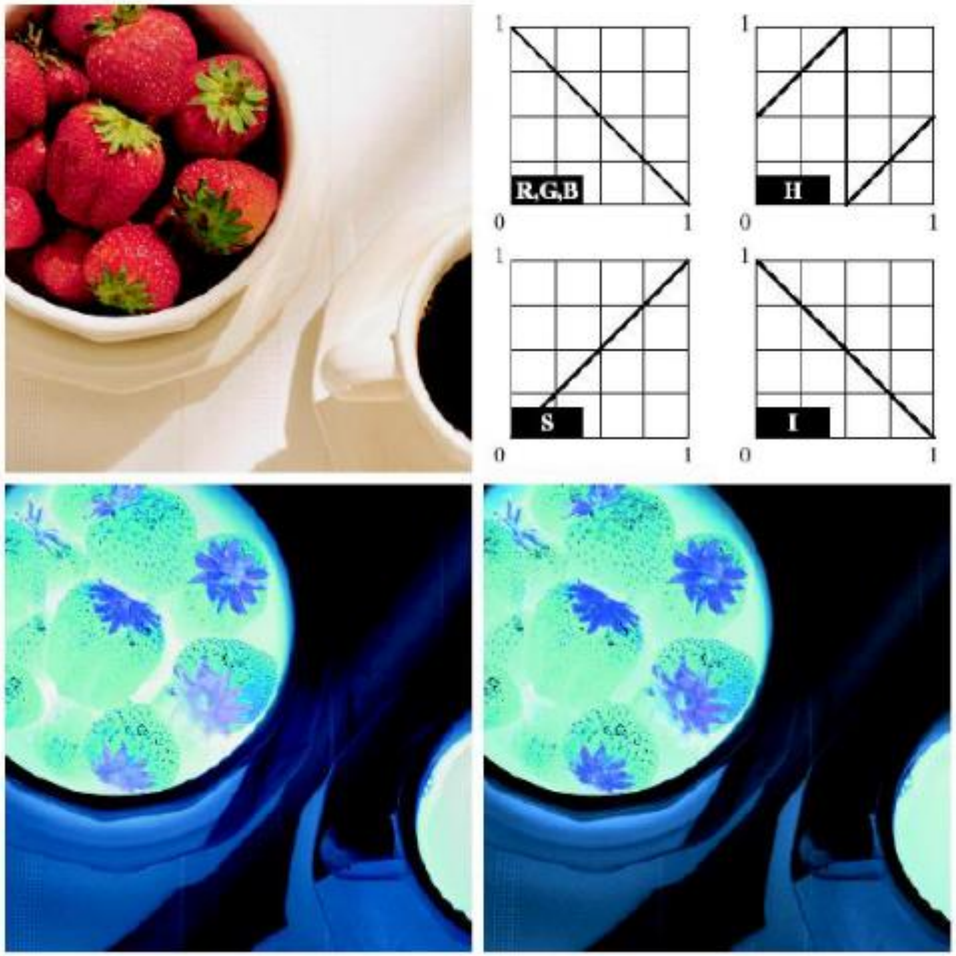
Color complement replaces each color with its opposite color in the color circle of the Hue component. This operation is analogous to image negative in a gray scale image.



Color circle

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Color Complement Transformation Example



a b
c d

FIGURE 6.33 Color complement transformations. (a) Original image. (b) Complement transformation functions. (c) Complement of (a) based on the RGB mapping functions. (d) An approximation of the RGB complement using HSI transformations.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Color Slicing Transformation

"Slicing" in color space: if the color of pixel is far from a desired color more than threshold distance, we set that color to some specific color such as gray, otherwise we keep the original color unchanged.

Method 1:

$$s_i = \begin{cases} 0.5 & \text{if } \left[\left| r_j - a_j \right| > \dfrac{W}{2} \right] \text{ For } 1 \le j \le n \\ r_i & \text{otherwise} \end{cases}$$

Set to Grey

Keep original color

Method 2:

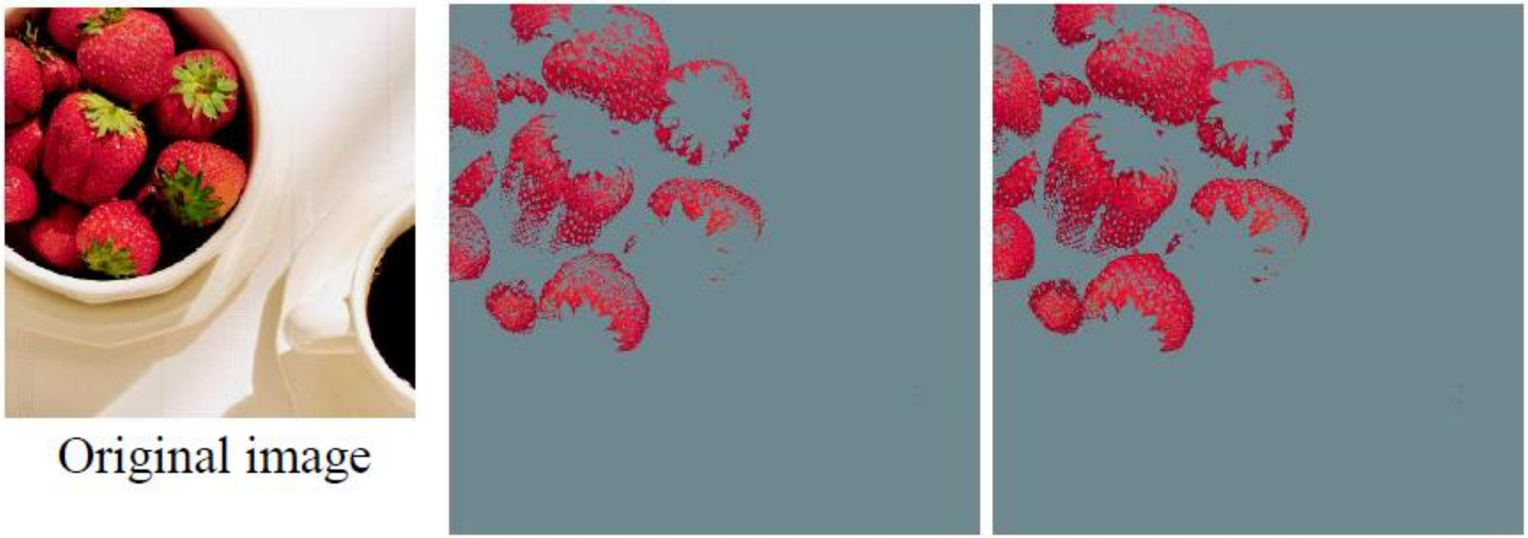
$$s_i = \begin{cases} 0.5 & \text{if } \sum_{j=1}^{n} (r_j - a_j)^2 > R_0^2 \\ r_i & \text{otherwise} \end{cases}$$

Set to Grey

Keep original color

Color Slicing Transformation Example

After color slicing



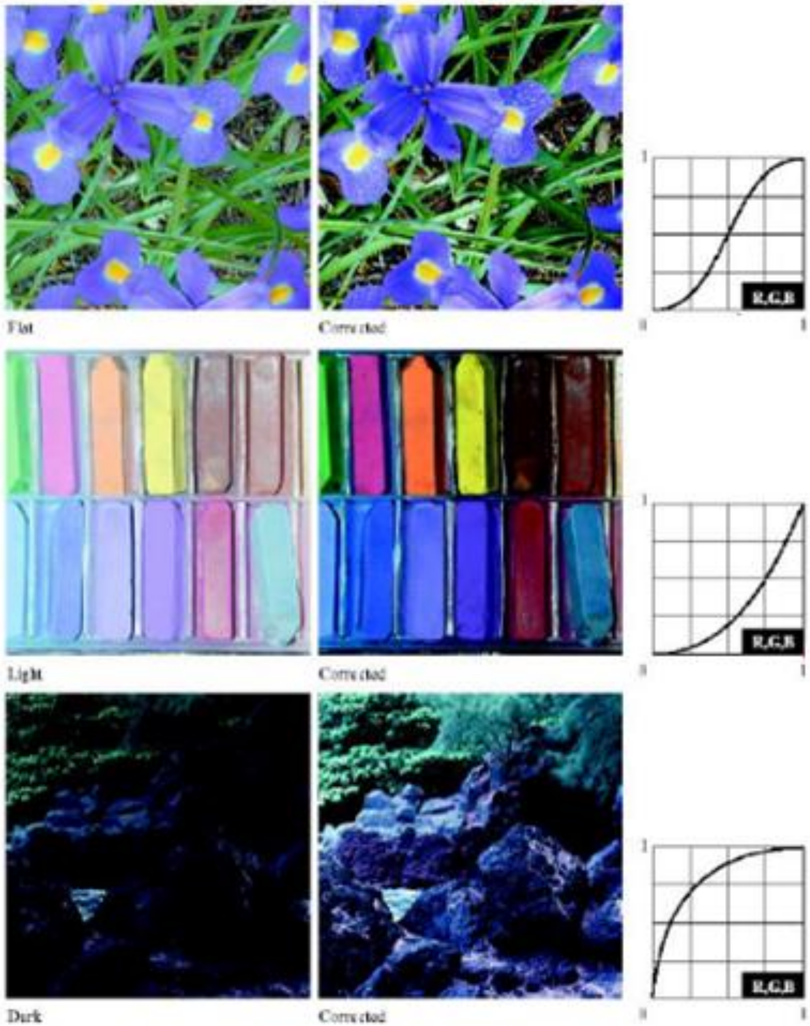
Original image

a b

FIGURE 6.34 Color slicing transformations that detect (a) reds within an RGB cube of width $W = 0.2549$ centered at $(0.6863, 0.1608, 0.1922)$, and (b) reds within an RGB sphere of radius 0.1765 centered at the same point. Pixels outside the cube and sphere were replaced by color $(0.5, 0.5, 0.5)$.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Tonal Correction Examples



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

- For the shown examples on the left, only brightness and contrast are adjusted while color is kept constant.
- Achieved through same transformation for all RGB components

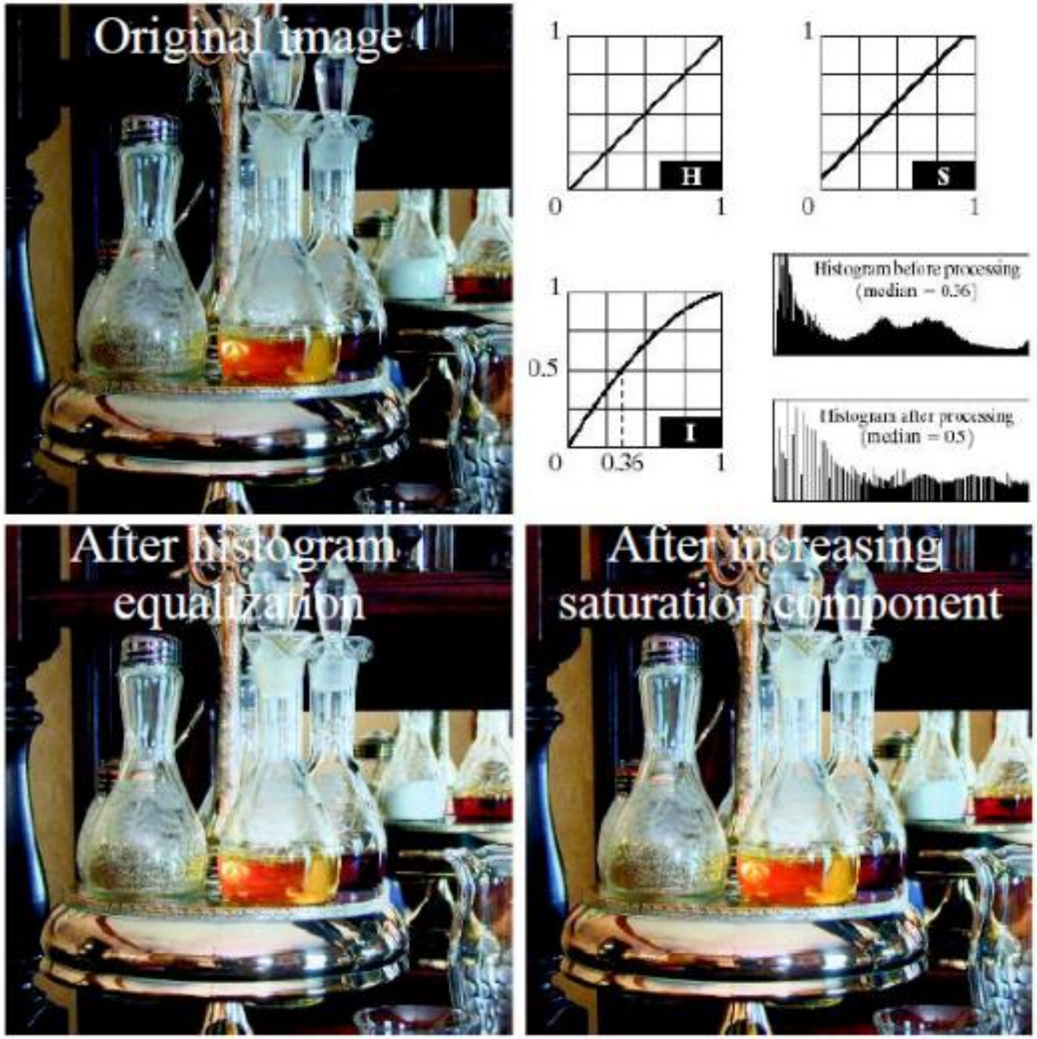
Histogram Equalization of a Full-Color Image

- Histogram equalization of a color image can be performed by adjusting the intensity of each color component uniformly throughout.
- The HSI model is suitable for histogram equalization where only Intensity (I) component is equalized.

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

where *r* and *s* are intensity components of input and output color image.

Histogram Equalization of a Full-Color Image



a b
c d

FIGURE 6.37 Histogram equalization (followed by saturation adjustment) in the HSI color space.

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Color Image Smoothing

2 Methods:

- Per-color-plane method: for RGB, CMY color models

  Smooth each color plane using moving averaging and

  the combine back to RGB

$$\bar{c}(x,y) = \frac{1}{K}\sum_{(x,y)\in S_{xy}} c(x,y) = \begin{bmatrix} \frac{1}{K}\sum_{(x,y)\in S_{xy}} R(x,y) \\ \frac{1}{K}\sum_{(x,y)\in S_{xy}} G(x,y) \\ \frac{1}{K}\sum_{(x,y)\in S_{xy}} B(x,y) \end{bmatrix}$$

- Smooth only Intensity component of a HSI image while leaving H and S unmodified.

Color Image Sharpening

We can do in the same manner as color image smoothing:

- Per-color-plane method for RGB,CMY images
- Sharpening only I component of a HSI image

Sharpening all RGB components

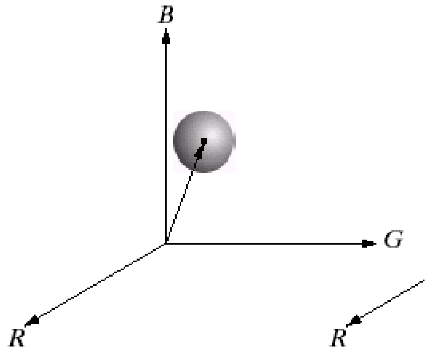
Sharpening only I component of HSI

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

Color Segmentation

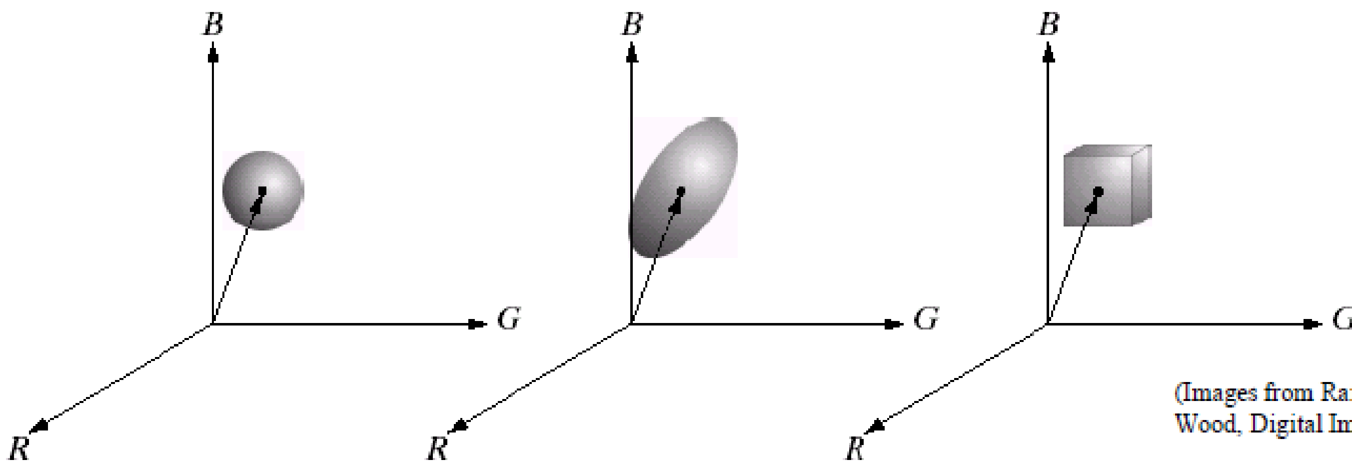
2 Methods:

- Segmented in HSI color space:
  - A thresholding function based on color information in H and S Components. We rarely use I component for color image segmentation.
- Segmentation in RGB vector space:
  - A thresholding function based on distance in a color vector space.



(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition

Color Segmentation in RGB Vector Space



a b c

FIGURE 6.43 Three approaches for enclosing data regions for RGB vector segmentation.

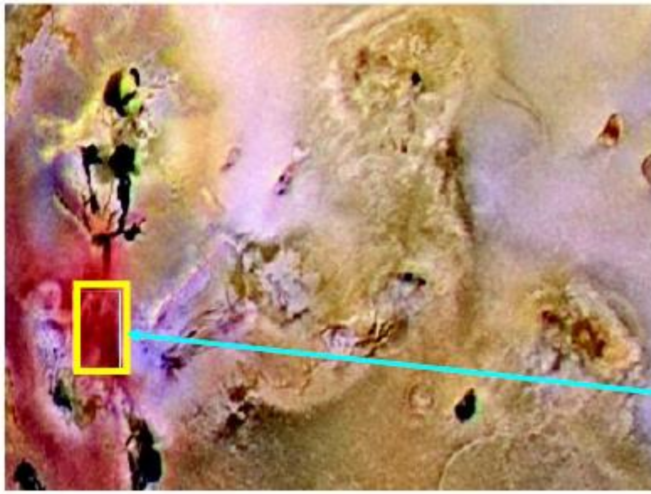
(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

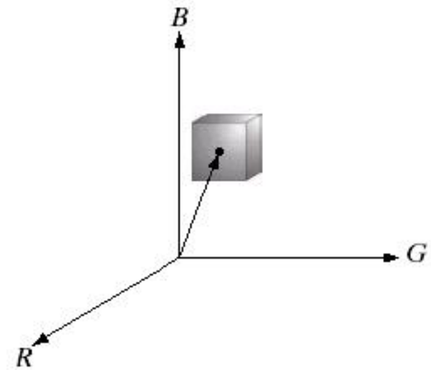
- Each pixel's RGS coordinate in the vector space represents a color
- Segmentation is based on distance thresholding in a vector space

$$g(x,y) = \begin{cases} 1 & \text{if} \quad D(c(x,y), c_T) \leq T \\ 0 & \text{if} \quad D(c(x,y), c_T) > T \end{cases}$$

where *D(u,v)* = distance function
c_T = color to be segmented
c(x,y) = RGB vector at pixel *(x,y)*

Example: Segmentation in RGB Vector Space





Color image

Reference color $\mathbf{c}_T$ to be segmented

$\mathbf{c}_T$ = average color of pixel in the box

Results of segmentation in RGB vector space with Threshold value

T = 1.25 times the SD of R,G,B values In the box

(Images from Rafael C. Gonzalez and Richard E. Wood, Digital Image Processing, 2nd Edition.

End of Module 2