WRIGHT STATE UNIVERSITY

Chapter-7
Color Image Processing

Color Image Processing

"It is only after years of preparation that the young artist should touch color – not color used descriptively, that is, but as a means of personal expression".

Henri Matisse

"For a long time I limited myself to one color – as a form of discipline".

Pablo Picasso

Color Image Processing Topics

- Color fundamentals
- Color models
- Pseudocolor image processing
- Color image smoothing and sharpening
- Color edge detection
- Noise in color images
- Color perception models

7.1 Color Fundamentals

- In 1666 Sir Isaac Newton discovered that when a beam of sunlight passes through a glass prism, the emerging beam is split into a spectrum of colors

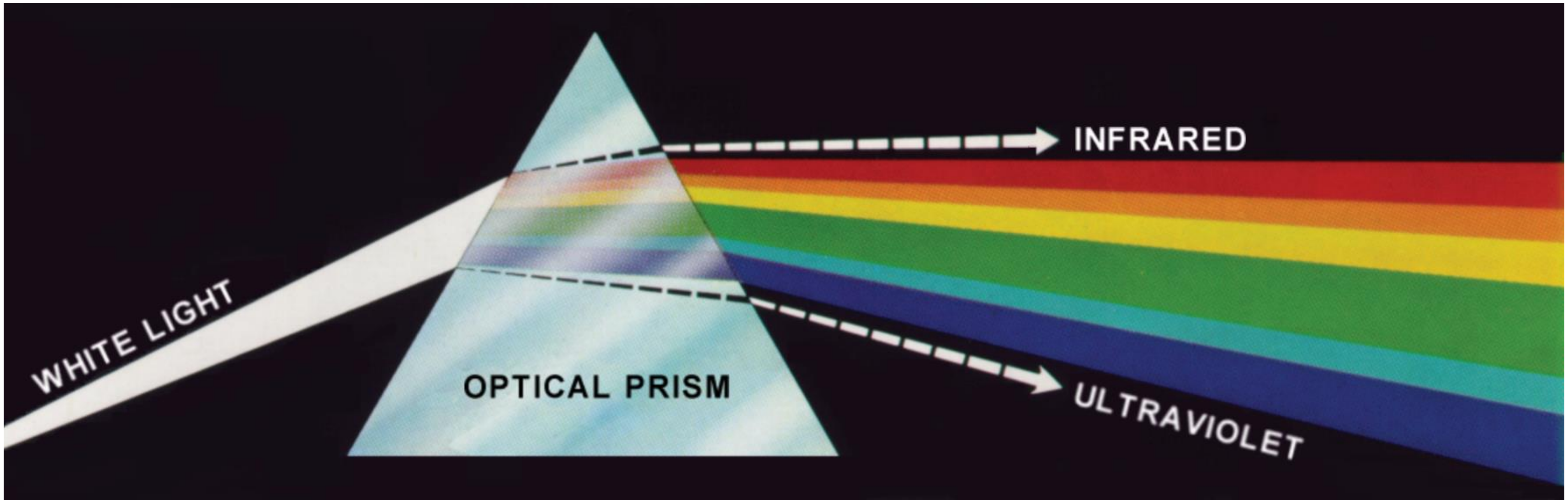
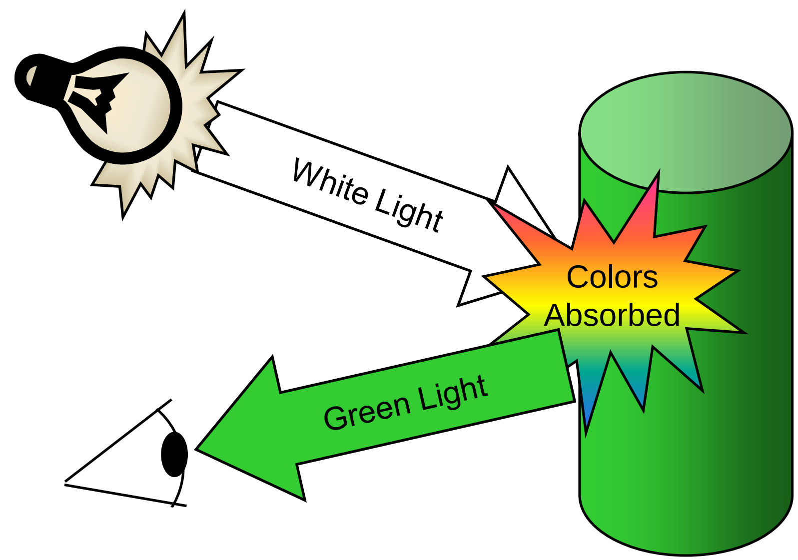


FIGURE 7.1 Color spectrum seen by passing white light through a prism. (Courtesy of the General Electric Co., Lighting Division.)

Color Fundamentals (cont…)

- The colors that humans and most animals perceive in an object are determined by the nature of the light reflected from the object
- For example, green objects reflect light with wavelengths primarily in the range of 500 - 570 nm while absorbing most of the other energy at other wavelengths.



Electromagnetic Spectrum

- Chromatic light (Visible by human eyes) spans the electromagnetic spectrum from around 400 to 700 nm.
- Human color vision is achieved through 6 to 7 million cones in each eye.
- For an achromatic (monochrome) light source, there is only one attribute to describe the quality: Intensity

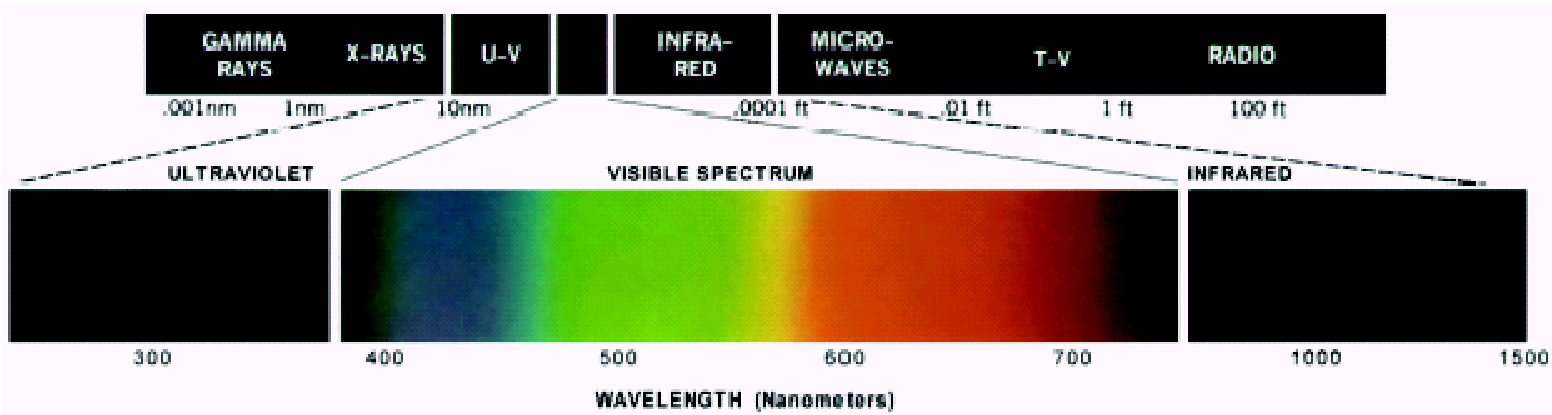


FIGURE 7.2 Wavelengths comprising the visible range of the electromagnetic spectrum. (Courtesy of the General Electric Co., Lighting Division.)

Color Fundamentals (cont…)

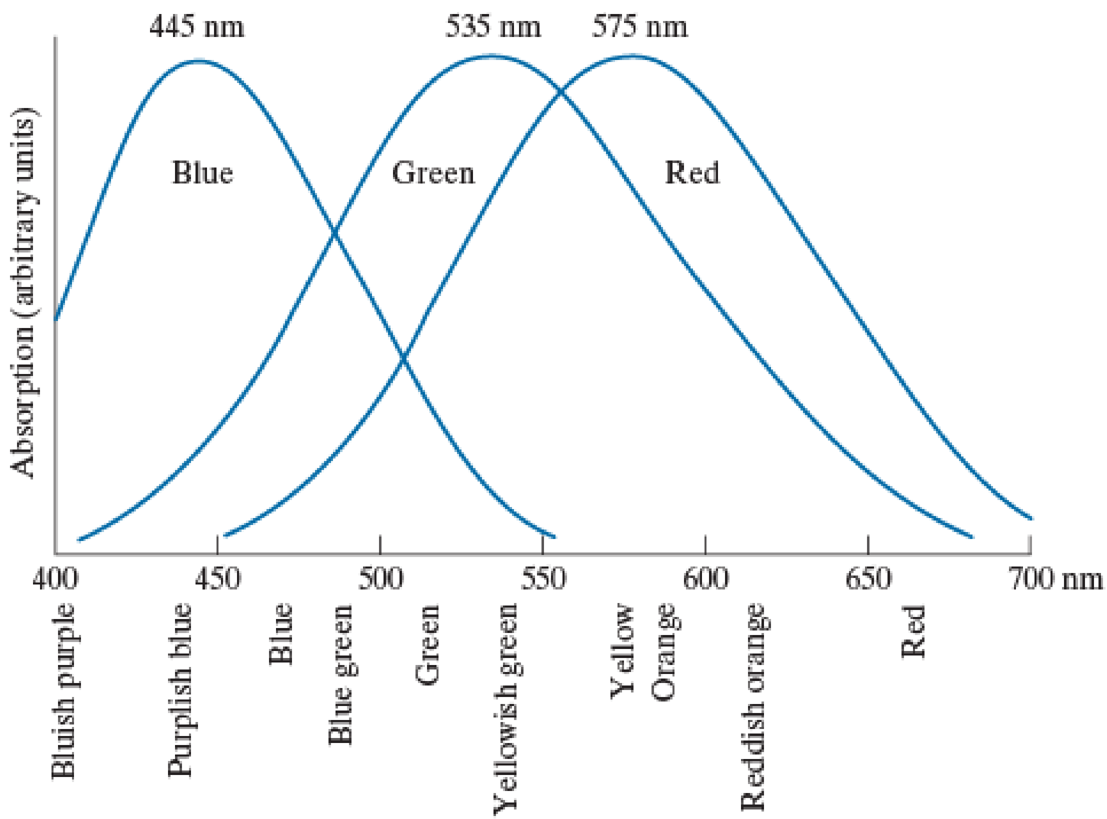
- For a chromatic light source, three attributes describe the quality :
  - **Radiance:** Total amount of energy that flows from the light source (measured in *watts*)
  - **Luminance:** Amount of energy an observer *perceives* from the light source (measured in *lumens*)
    - Note: We can have high radiance, but low luminance
  - **Brightness (= Intensity):** A subjective (practically unmeasurable) notion that embodies the achromatic notion of intensity of light

Color Fundamentals (cont…)

- 6-7 million cones in the human eye can be divided into three principal color sensing groups:
  - 66% of these cones are sensitive to red light
  - 33% to green light
  - 2% to blue light
- Absorption curves for the different cones have been determined experimentally
- Strangely, these do not match the CIE standards for red (700nm), green (546.1 nm) and blue (435.8 nm) light because the standards were developed well before the experiments could be done!

Color Fundamentals (cont…)

Note: The curves centered at Green and Red are very close



Primary colors:
Defined CIE in 1931
- Red = 700 nm
- Green = 546.1 nm
- Blue = 435.8 nm

FIGURE 7.3 Absorption of light by the red, green, and blue cones in the human eye as a function of wavelength.

CIE = Commission Internationale de l'Eclairage
(The International Commission on Illumination)

Color Fundamentals (cont…)

- **Wrong** : Linear combination of the three **primaries** (R, G, B) may produce all visible colors

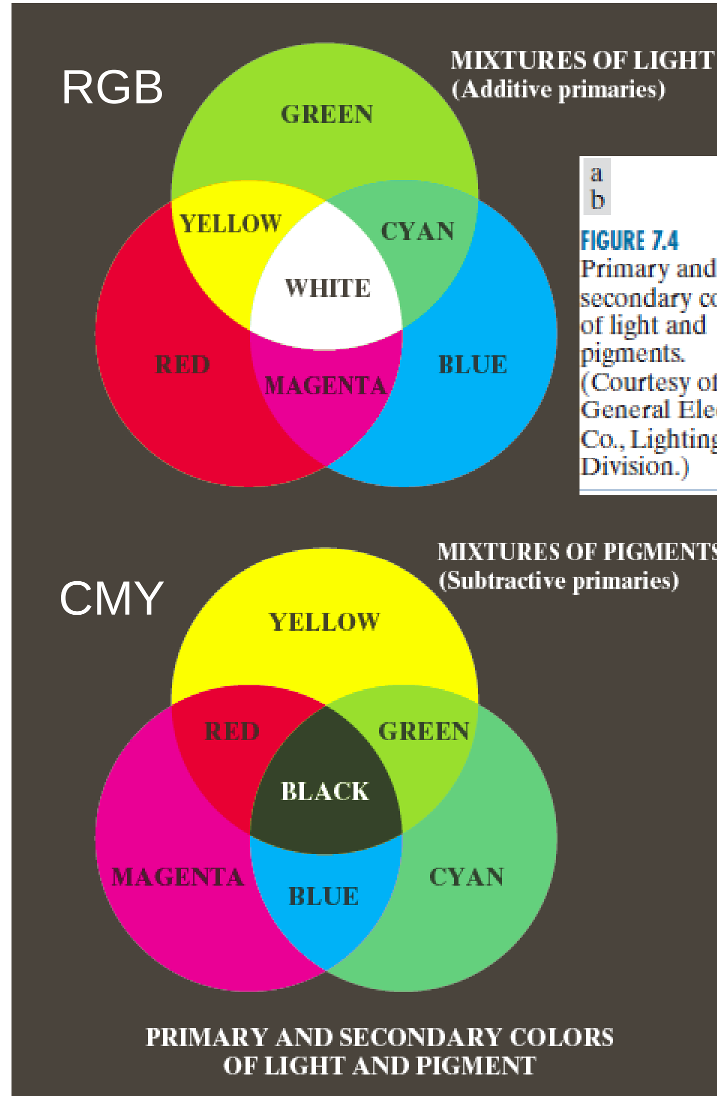
- This would be true only if the centers of the three curves are shifted
  - This means that the wavelengths must change but then we no longer have the same primaries!

- The three curves are not a basis

Additive, Subtractive, Primary and Secondary Colors

- **Additive primary colors**:
  - RGB used in case of light sources such as color monitors
  - The primary colors can be added to produce the secondary colors.
  - Mixing the three primaries (RGB) produces white.
  - Mixing a secondary with its opposite primary produces white (e.g. red+cyan).

- **Subtractive primary colors**: CMY used in the case of pigments in printing devices
  - e.g., Subtracting Green from White paper produces Magenta & so on



a
b

FIGURE 7.4 Primary and secondary colors of light and pigments. (Courtesy of the General Electric Co., Lighting Division.)

Color Fundamentals (cont…)

Important difference:

- Additive Primary colors (red, green, blue)
- Subtractive Primary colors of pigments (colorants)
  - A color that subtracts or absorbs a primary color of light and reflects the other two.
    - e.g., Magenta = Red + Blue (Green is subtracted from White)
  - Pigment Colors: Cyan, Magenta and Yellow (CMY)
  - Proper combination of pigment primaries produces black.
  - Save Money: No pigment needed to produce white

WRIGHT STATE UNIVERSITY

Color Fundamentals (cont…)

The characteristics generally used to distinguish one color from another are :

- **Brightness**: The achromatic notion of intensity
- **Hue: Dominant wavelength** in a mixture of light waves
  - Dominant color perceived by an observer,
  - When we say, "red" or "orange" object → That's Hue
- **Saturation**: Relative purity or the amount of white light mixed with a hue. (inversely proportional to amount of white light added)
  - Pure colors → Fully saturated, e.g., red or blue … but
  - Pink (red + white) → Less saturated
  - Lavender (violet + white) → Less saturated

Color Characterization

- Hue and Saturation taken together: *Chromaticity*

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

- Therefore, any color is characterized by its
  - Brightness and
  - Chromaticity.
- The amounts of red, green and blue needed to form a particular color are called Tristimulus values, denoted by X, Y and Z.

Color Fundamentals (cont…)

- A color is then specified by its *Trichromatic coefficients*:

RED GREEN BLUE

$$x = \frac{X}{X+Y+Z}, \; y = \frac{Y}{X+Y+Z}, \; z = \frac{Z}{X+Y+Z}$$

$$x + y + z = 1$$

- For any visible wavelength, the tristimulus values needed to produce specific color corresponding to particular wavelength are obtained by curves compiled by extensive experimentation.

CIE Chromaticity Diagram

- Specifying colors systematically can be achieved using the CIE **chromaticity diagram.**

- On this diagram the ***x*-axis** represents the proportion of Red and the ***y*-axis** represents the proportion of Green used to produce a specific color.

- The proportion of blue used in a color is calculated as:

$$z = 1 - (x + y)$$

CIE Chromacity Diagram (cont…)

- Shows color composition as a function of x (red) and y (green)
- Point marked "Green"
  - 62% green, 25% red and 13% blue. (sum=1)
- Point marked "Red"
  - 32% green, 67% red and 1% blue. (sum=1)
- The diagram is useful for color mixing.

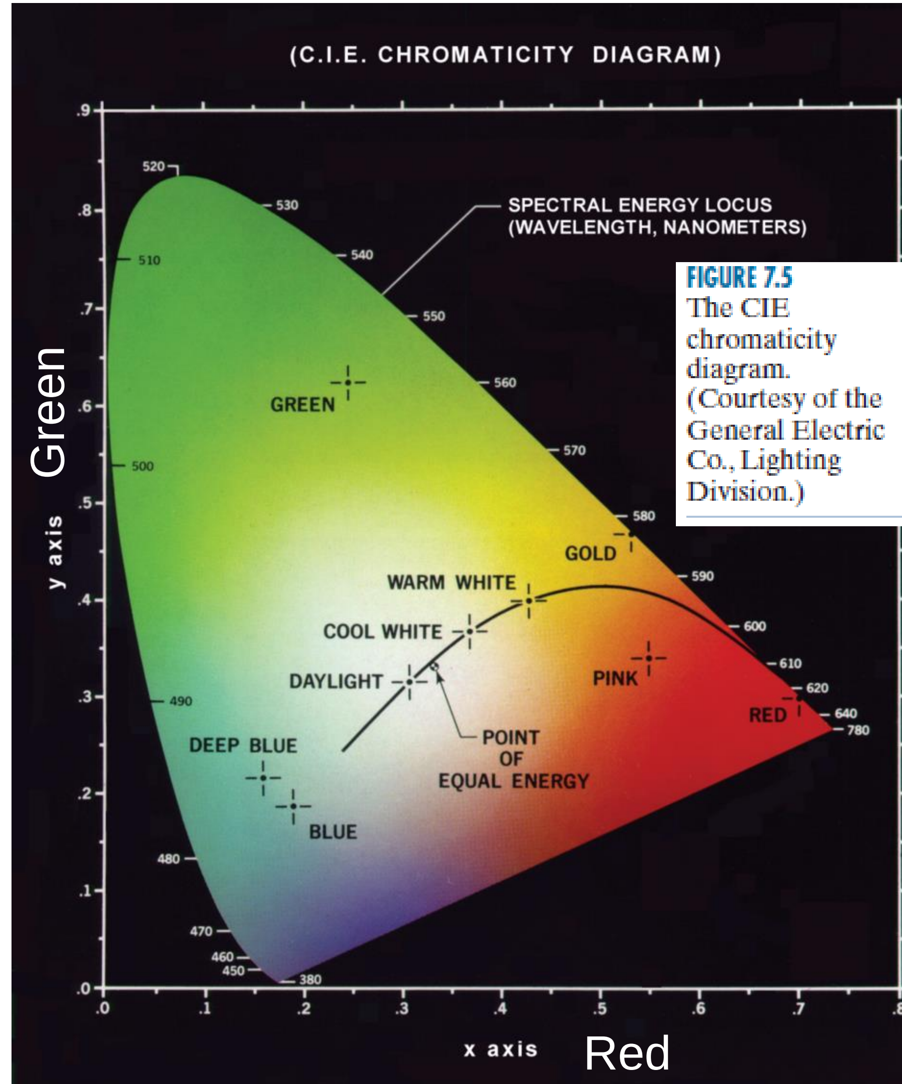
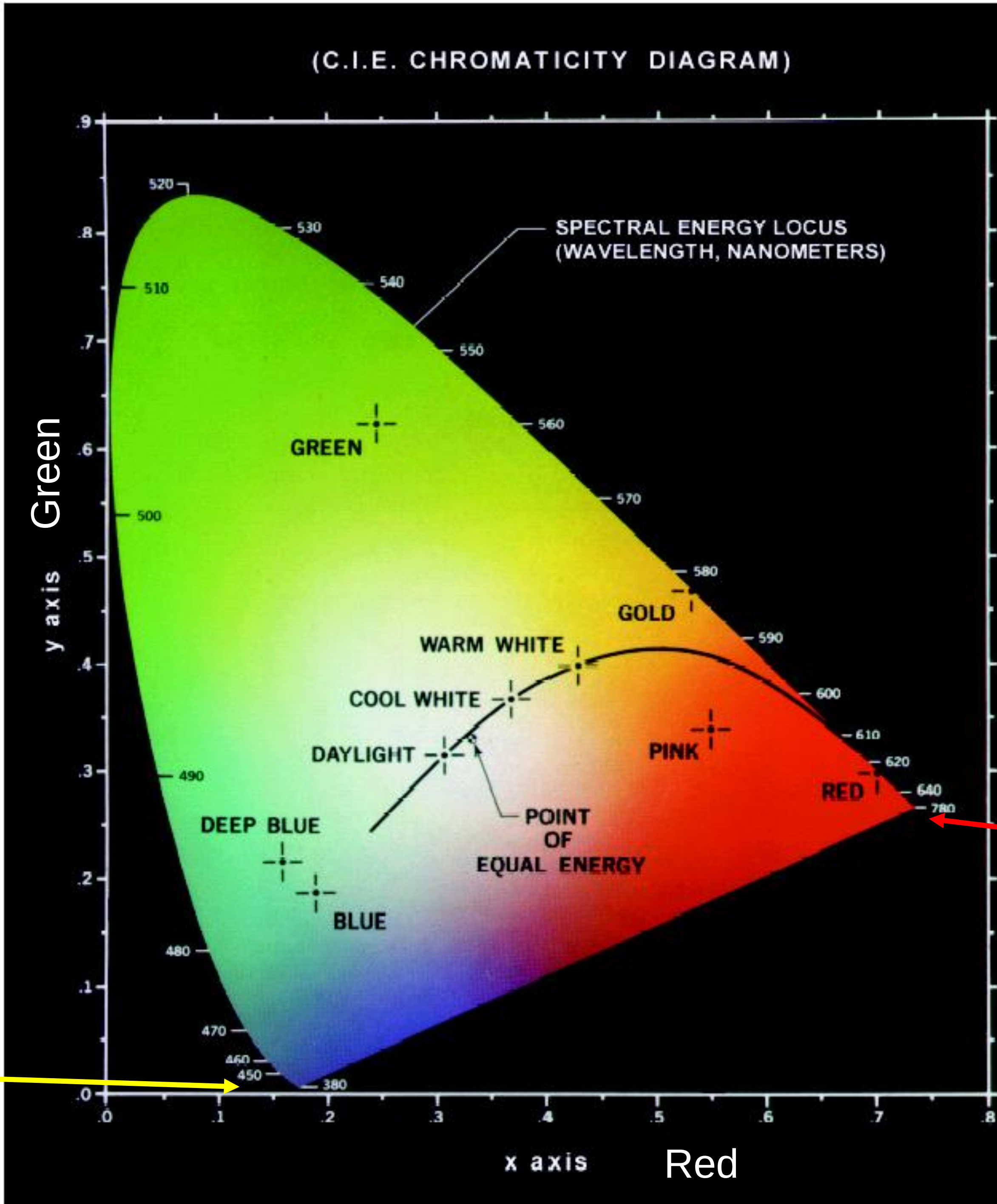


FIGURE 7.5 The CIE chromaticity diagram. (Courtesy of the General Electric Co., Lighting Division.)

Boundary: Different pure spectrum colors → saturated



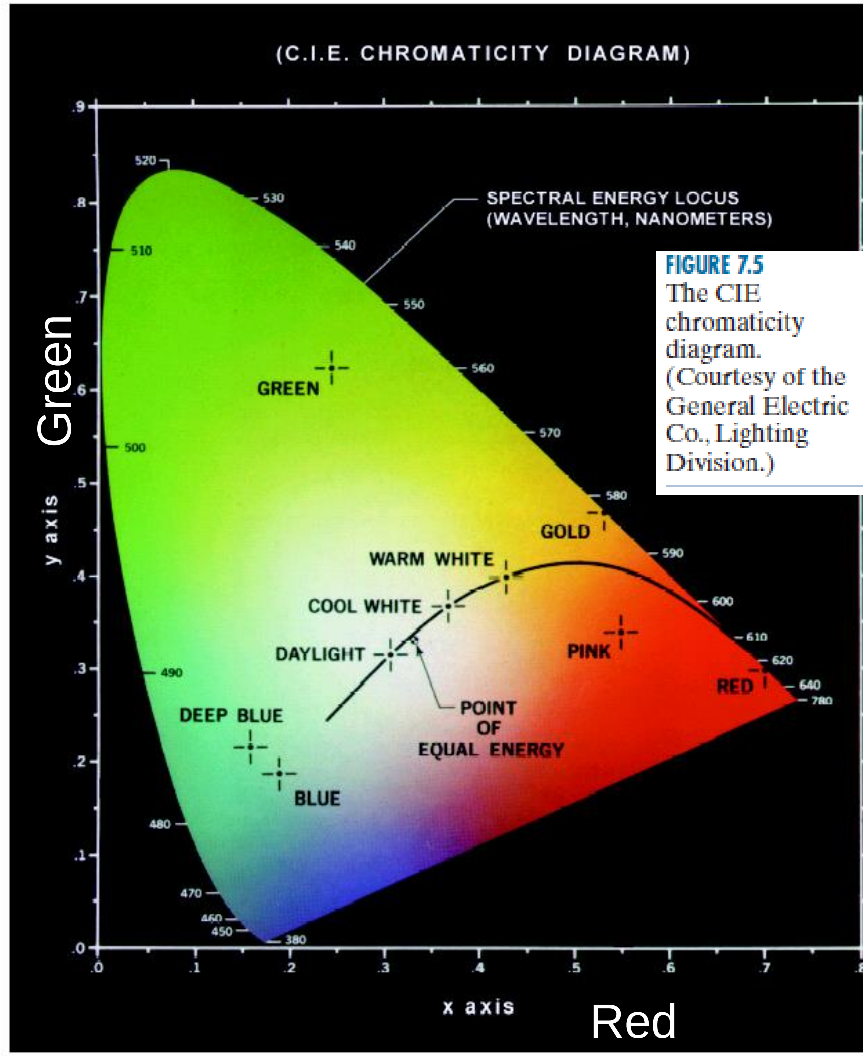
Violet: 380nm

Red: 720nm

FIGURE 7.5 The CIE chromaticity diagram. (Courtesy of the General Electric Co., Lighting Division.)

CIE Chromacity Diagram (cont…)

- Any color located on the boundary of the chromaticity chart is fully saturated (*Pure colors*).
- The point of equal energy (PEE) has equal amounts of red, green and blue. (= pure white)
  - It is the CIE standard for pure white.



CIE Chromacity Diagram (cont…)

- Any straight line joining two points in the diagram defines all the different colors that can be obtained by combining these two colors additively.
- A line drawn from the PEE to any point on the boundary defines all the shades of that particular color.

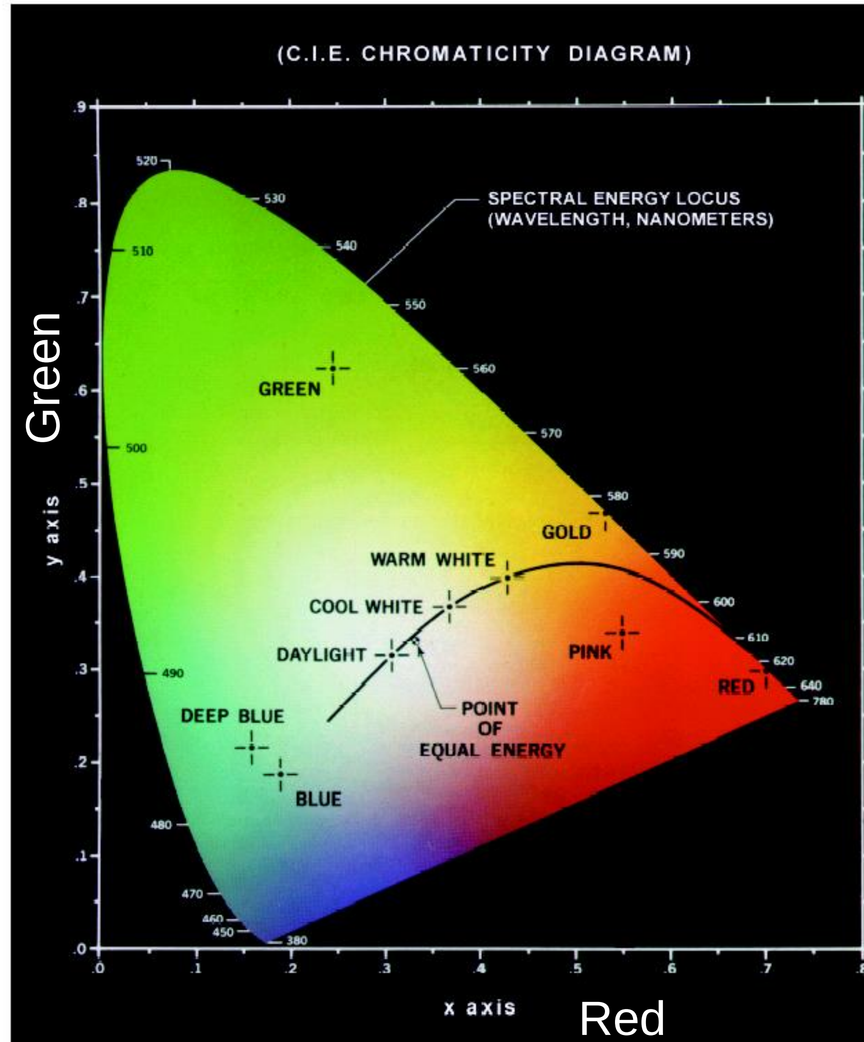


FIGURE 7.5 The CIE chromaticity diagram. (Courtesy of the General Electric Co., Lighting Division.)

CIE Chromacity Diagram (cont…)

- By combining any three given colors we may obtain the colors enclosed in the triangle defined by the three initial colors.

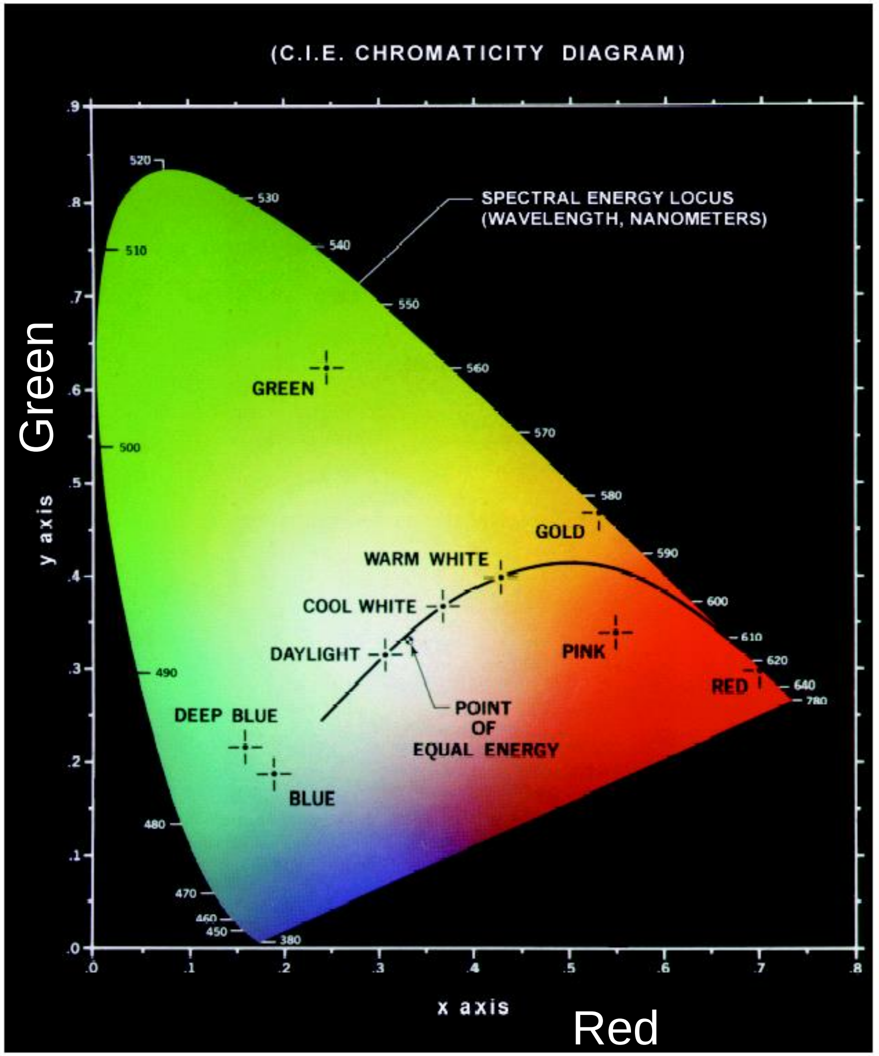


FIGURE 7.5 The CIE chromaticity diagram. (Courtesy of the General Electric Co., Lighting Division.)

CIE Chromacity Diagram (cont…)

- A triangle with vertices at any three *fixed pure colors* can not enclose the entire color region.

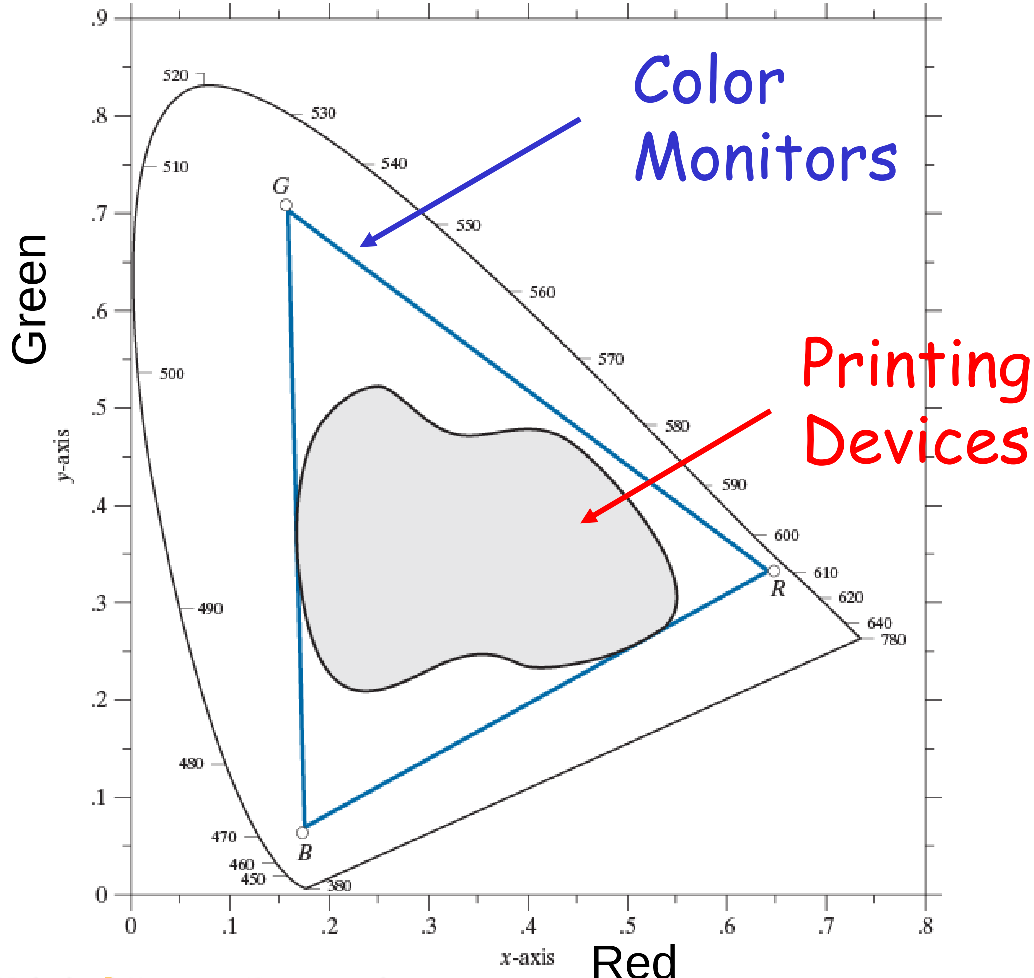


FIGURE 7.6 Illustrative color gamut of color monitors (triangle) and color printing devices (shaded region).

Color gamut of Color Monitors and Printing Devices

- The triangle shows the typical color gamut produced by RGB monitors.
- The entire color range cannot be displayed based on any three colors.
- The irregular shape is the gamut achieved by high quality color printers.

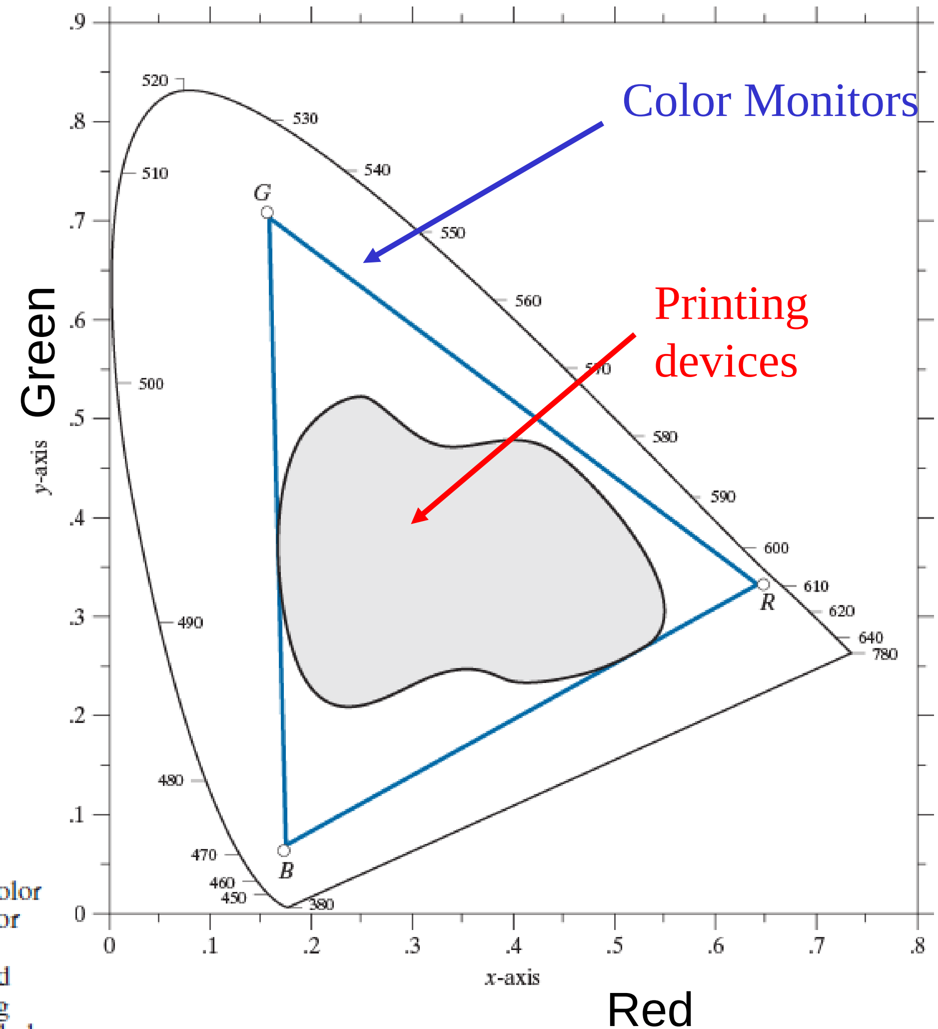


FIGURE 7.6 Illustrative color gamut of color monitors (triangle) and color printing devices (shaded region).

Color gamut of Color Monitors and Printing Devices

- The boundary of the printing gamut is irregular because printing is a combination of additive and subtractive color mixing.
- This is a more difficult process to control than that of displaying colors on a monitor using additive RGB primaries.

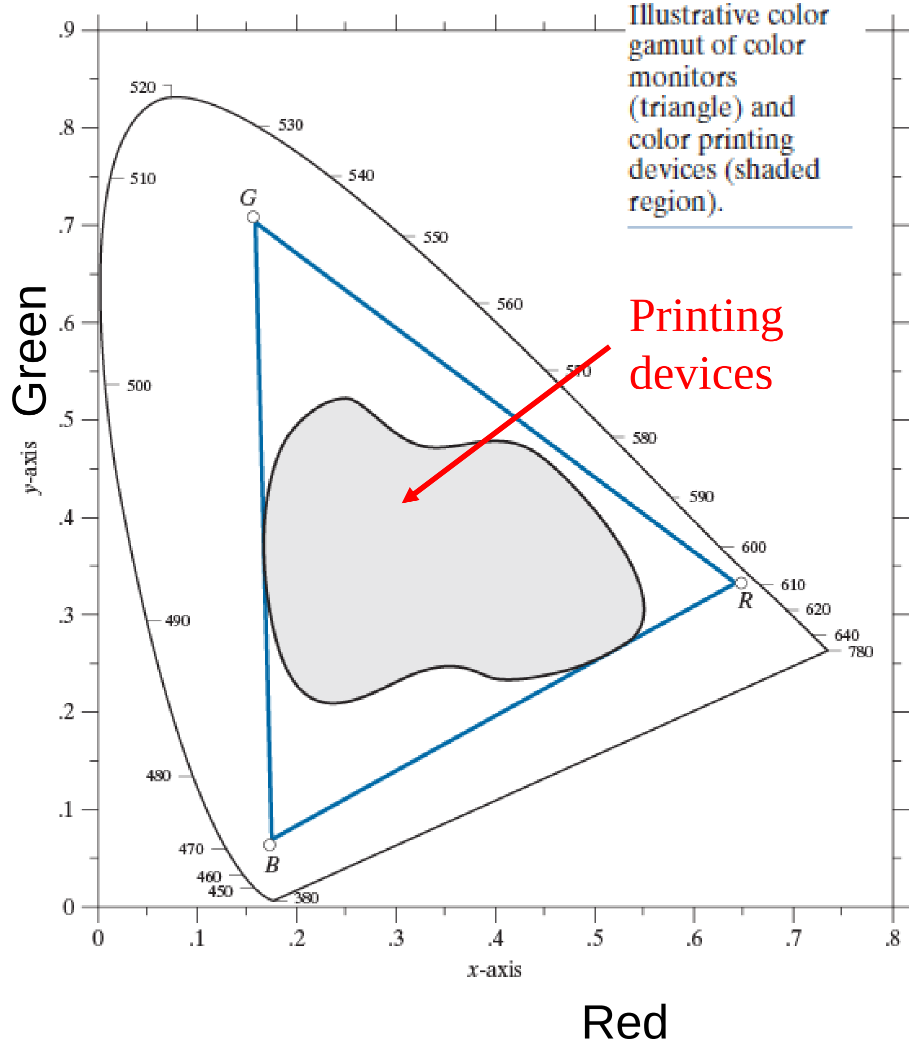


FIGURE 7.6 Illustrative color gamut of color monitors (triangle) and color printing devices (shaded region).

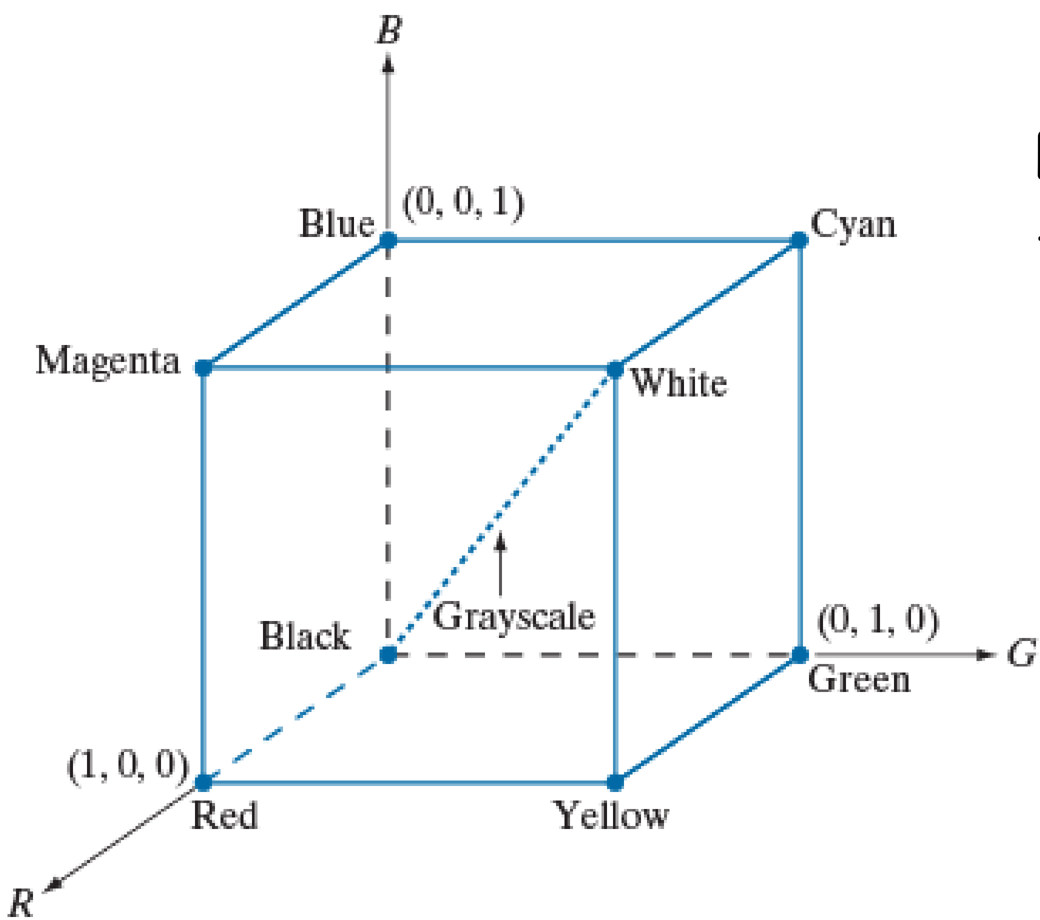
7.2 Color Models

- Purpose of color models: To facilitate specification of colors in some standard
- From previous discussion it is obvious that there are different ways to model color.
- We will consider two most popular models used in color image processing:

  - RGB (**R**ed **G**reen **B**lue)
  - HSI (**H**ue **S**aturation **I**ntensity)

7.2 RGB – For Monitors

- In the RGB model each color appears in its primary spectral components of red, green and blue.



RGB color models:
- Based on Cartesian coordinate system

FIGURE 7.7 Schematic of the RGB color cube. Points along the main diagonal have gray values, from black at the origin to white at point (1, 1, 1).

RGB

- RGB values are at 3 corners on primary axis.
- Cyan, Magenta and Yellow are at three other corners of the cube.
- Black is at the origin.
- White is the corner furthest from the origin.
- Different colors are points on or inside the cube represented by RGB vectors.

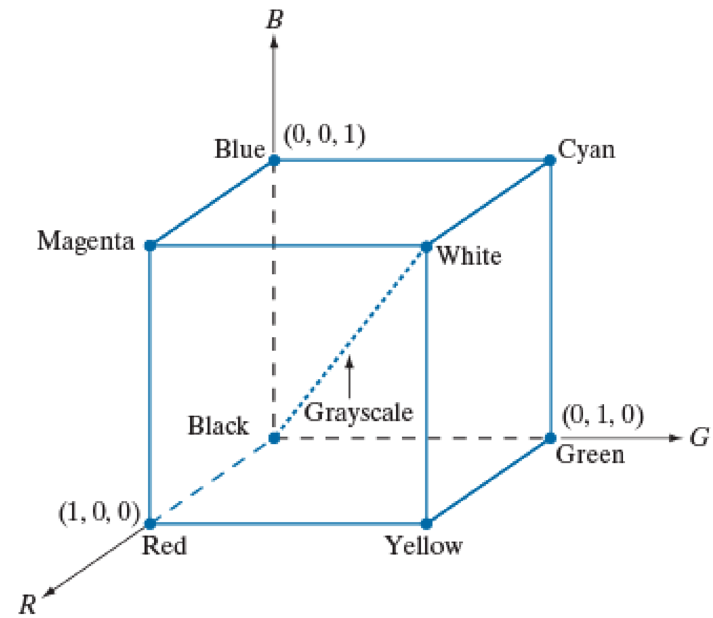


FIGURE 7.7 Schematic of the RGB color cube. Points along the main diagonal have gray values, from black at the origin to white at point (1, 1, 1).

FIGURE 7.8 A 24-bit RGB color cube.

RGB (cont…)

- Images represented in the RGB color model consist of three component images – one for each primary color.
- When fed into a monitor these images are combined to create a composite color image.
- The number of bits used to represent each pixel is referred to as the color depth.
- A 24-bit image is often referred to as a full-color image as it allows $\left(2^8\right)^3$ = 16,777,216 colors.

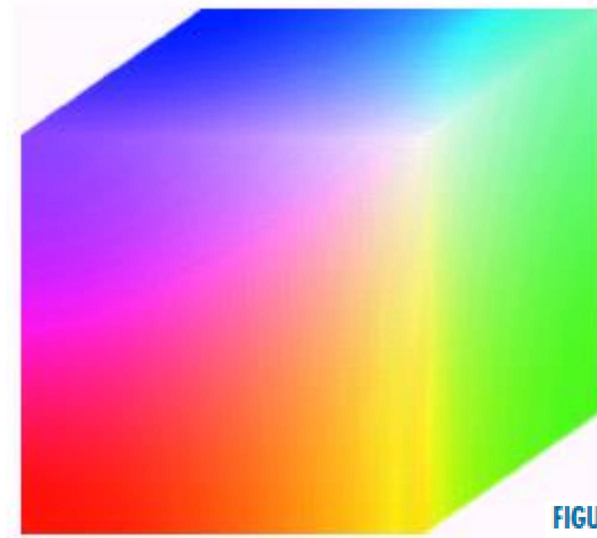


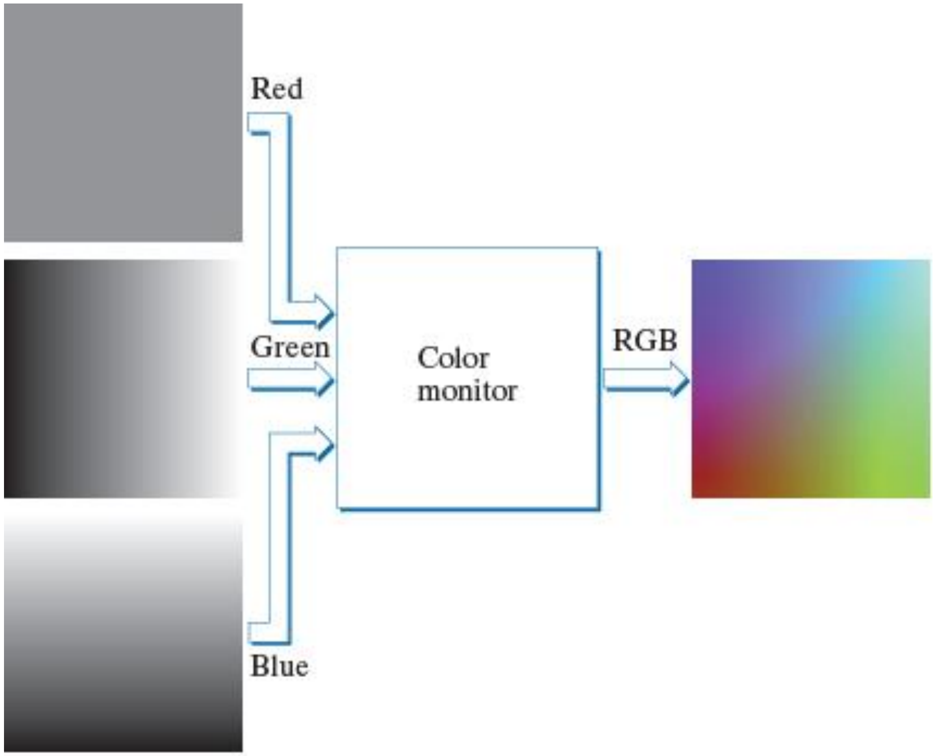
FIGURE 7.8
A 24-bit RGB color cube.

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

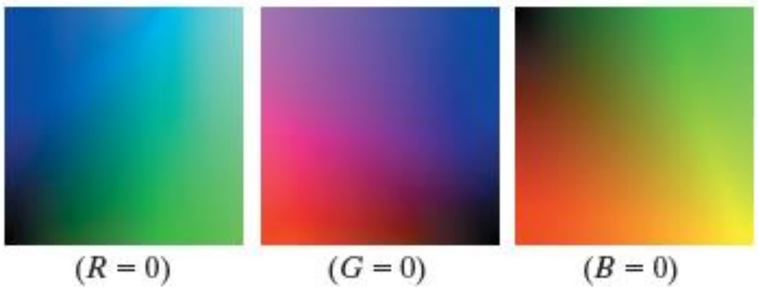
RGB (cont...)

a
b

FIGURE 7.9
(a) Generating the RGB image of the cross-sectional color plane (127, G, B).
(b) The three hidden surface planes in the color cube of Fig. 7.8.

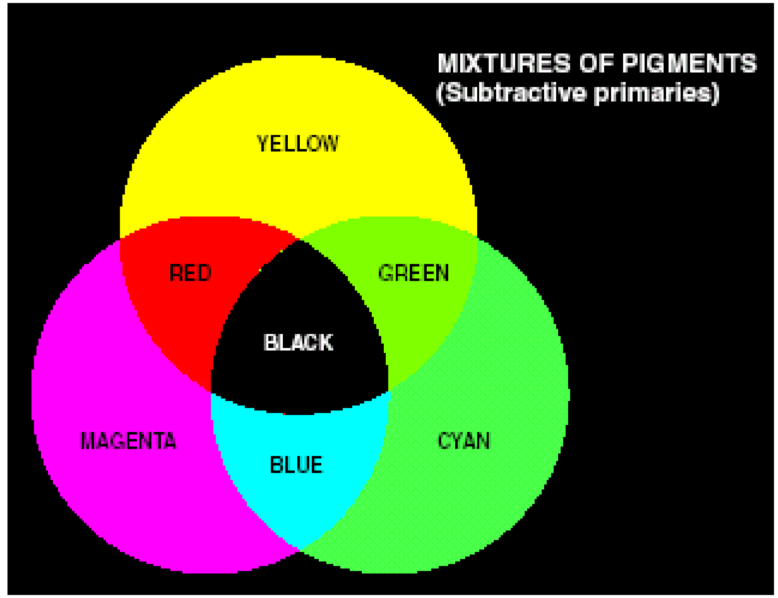


These are scalar valued pixels



CMY and CMYK Color Models

Used in
Color Printers



C = Cyan (Subtracts Red from white=1)
M = Magenta (Subtracts Green from white=1)
Y = Yellow (Subtracts Blue from white=1)
K = Black = C + M + Y (instead, use Black pigment to save money)

$$\begin{bmatrix} C \\ M \\ Y \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

Conversion of CMY to CMYK

- Begin by, $K = \min(C, M, Y)$
- For K=1, we have pure black and no color, i.e.,

$$C = 0, \quad M = 0, \quad Y = 0$$

- Otherwise,

$$C = (C - K)/(1 - K)$$

$$M = (M - K)/(1 - K)$$

$$Y = (Y - K)/(1 - K)$$

All values are in the range [0,1]

- The conversion of CMYK back to CMY are,

$$C = C * (1 - K) + K$$

$$M = M * (1 - K) + K$$

$$Y = Y * (1 - K) + K$$

The HSI Color Model

- RGB is useful for hardware implementations and is serendipitously related to the way in which the human visual system works.
- However, RGB is not a particularly intuitive way in which to describe colors.
- Rather when people describe colors they tend to use **hue**, **saturation** and **brightness**.
- RGB is great for color generation, but HSI is great for color description.

The HSI Color Model (cont…)

Recall:

- **Hue:** A color attribute that describes a pure color (pure yellow, orange or red).
- **Saturation:** Measure of how much a pure color is diluted with white light.
- **Intensity: Brightness** is nearly impossible to measure because it is so subjective. Instead we use intensity. Intensity is the same achromatic notion that is seen in grey level images.

HSI Color Model

- RGB and CMY models are not good for human interpretation

HSI Color model:

Hue: Dominant color (i.e., Wavelength)

Saturation: Relative purity (inversely proportional to amount of white light added)

Color carrying information

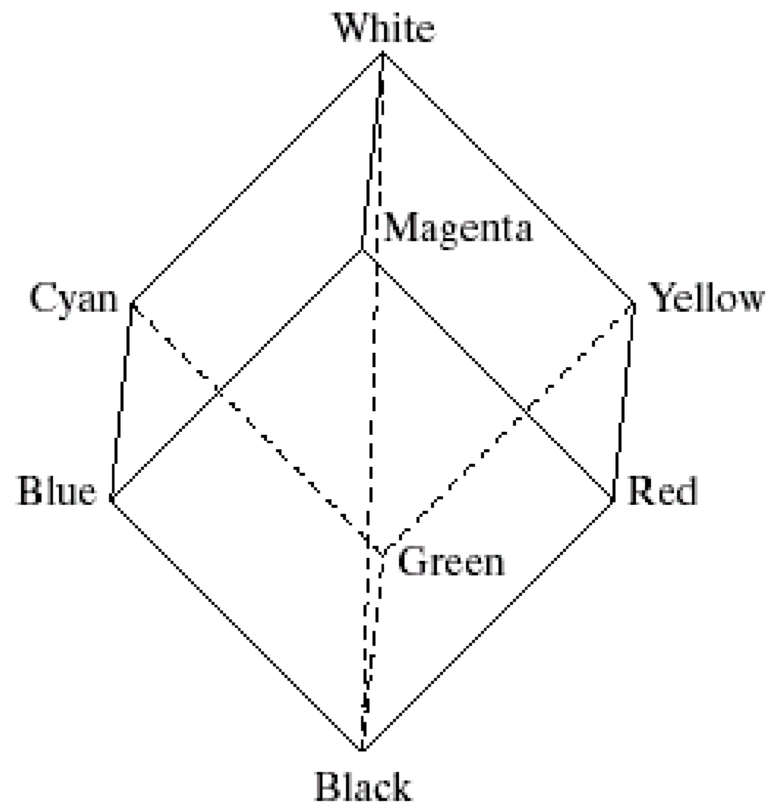
Intensity: Brightness

HSI, Intensity & RGB

- Intensity can be extracted from RGB images.
- However, human perception of color does not refer to percentages of RGB.
- Recall: The diagonal on the RGB color cube runs from black to white.
- Now consider if we stand this cube on the black vertex and position the white vertex directly above it.

Intensity Axis

- The intensity component of any color can be determined by passing a plane *perpendicular* to the intensity axis and containing the color point.
- The intersection of the plane with the intensity axis gives the intensity component of the color.

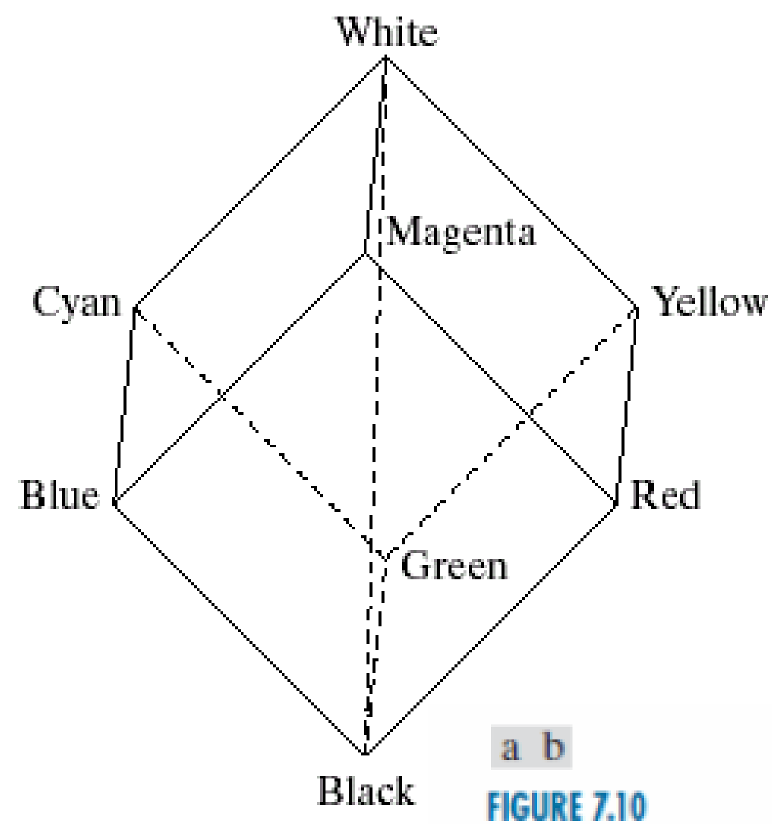


a b

FIGURE 7.10 Conceptual relationships between the RGB and HSI color models.

Saturation of a Color

- The saturation of a color (percentage of white missing from the color) increases as a function of distance from the intensity axis.
- We say, Pure or Saturated color at the edges



a b

FIGURE 7.10 Conceptual relationships between the RGB and HSI color models.

HSI, Hue & RGB

- In a similar way we can extract the hue (Color) from the RGB color cube.
- Consider a plane defined by the three points cyan, black and white.
- All points contained in this plane must have the same hue (cyan) as black and white cannot contribute hue information to a color.

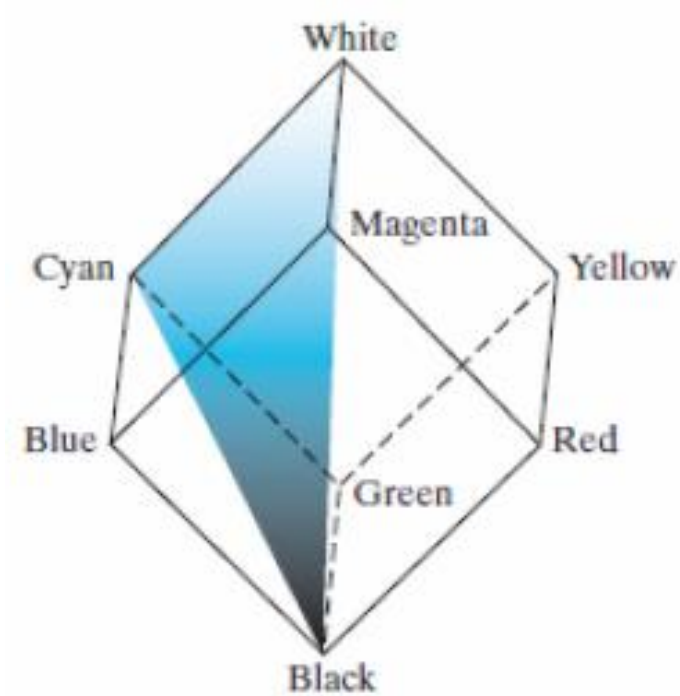
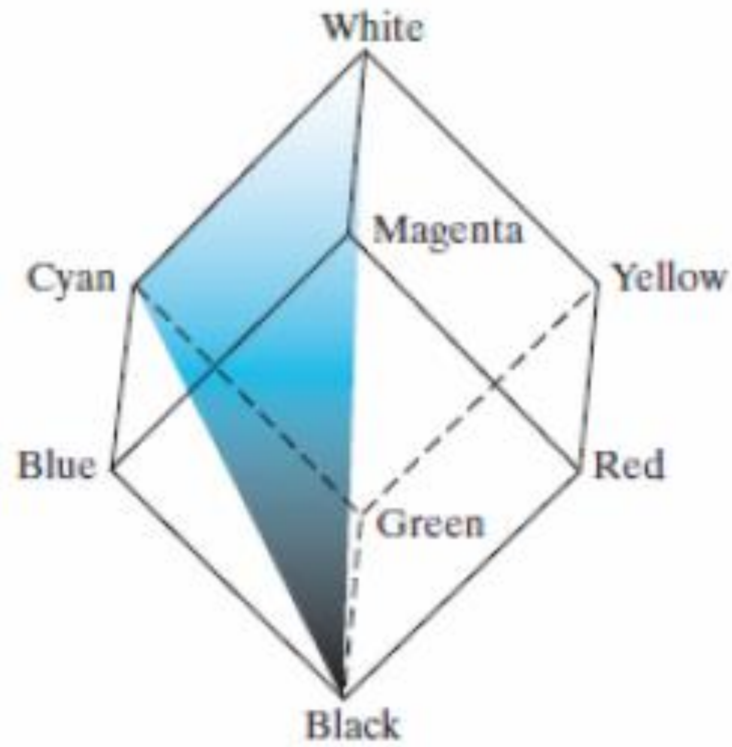


FIGURE 7.10 Conceptual relationships between the RGB and HSI color models.

HSI, Hue & RGB

- Rotating the shaded plane around the intensity axis we obtain different hues.
- Conclusion:
  - The HSI values can be obtained from the RGB values.
  - We need to work the geometric formulas.

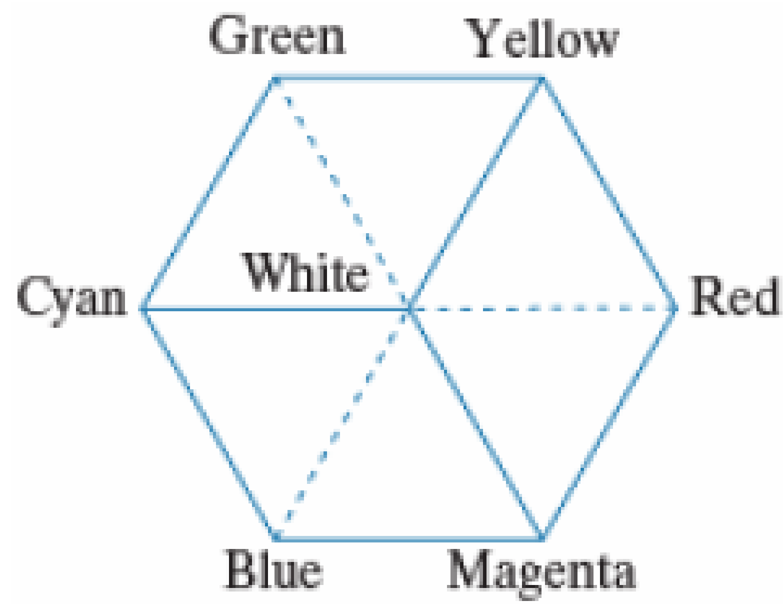


a b

FIGURE 7.10 Conceptual relationships between the RGB and HSI color models.

The HSI Color Model

- If looked straight down at the RGB cube as it was arranged previously a hexagonal shape can be seen with each primary color separated by 120° and secondary colors at 60° from the primaries.
- **HSI Model:** Composed of a vertical intensity axis and the locus of color points that lie on planes perpendicular to that axis.



The HSI Color Model (cont…)

- Hexagonal shape at an arbitrary color point
  - **Hue (H):** Determined by the angle from a reference point, usually red (H=0).
  - **Saturation (S):** Distance from the origin to the point.
  - **Intensity (I):** Determined by how far up the vertical intensity axis this hexagonal plane sits
    - ➢ Not apparent from this diagram

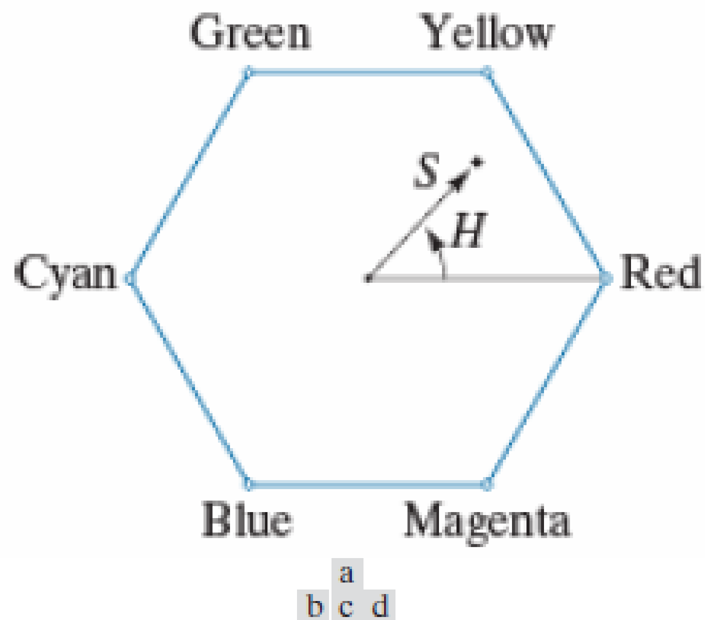


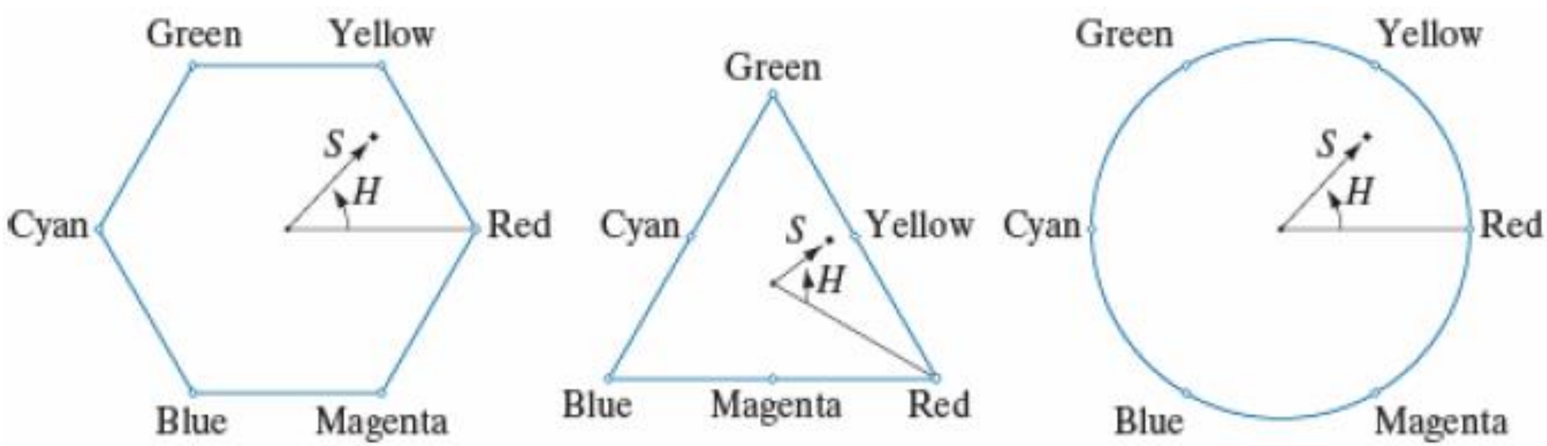
FIGURE 7.11 Hue and saturation in the HSI color model. The dot is any color point. The angle from the red axis gives the hue. The length of the vector is the saturation. The intensity of all colors in any of these planes is given by the position of the plane on the vertical intensity axis.

The HSI Color Model (cont…)

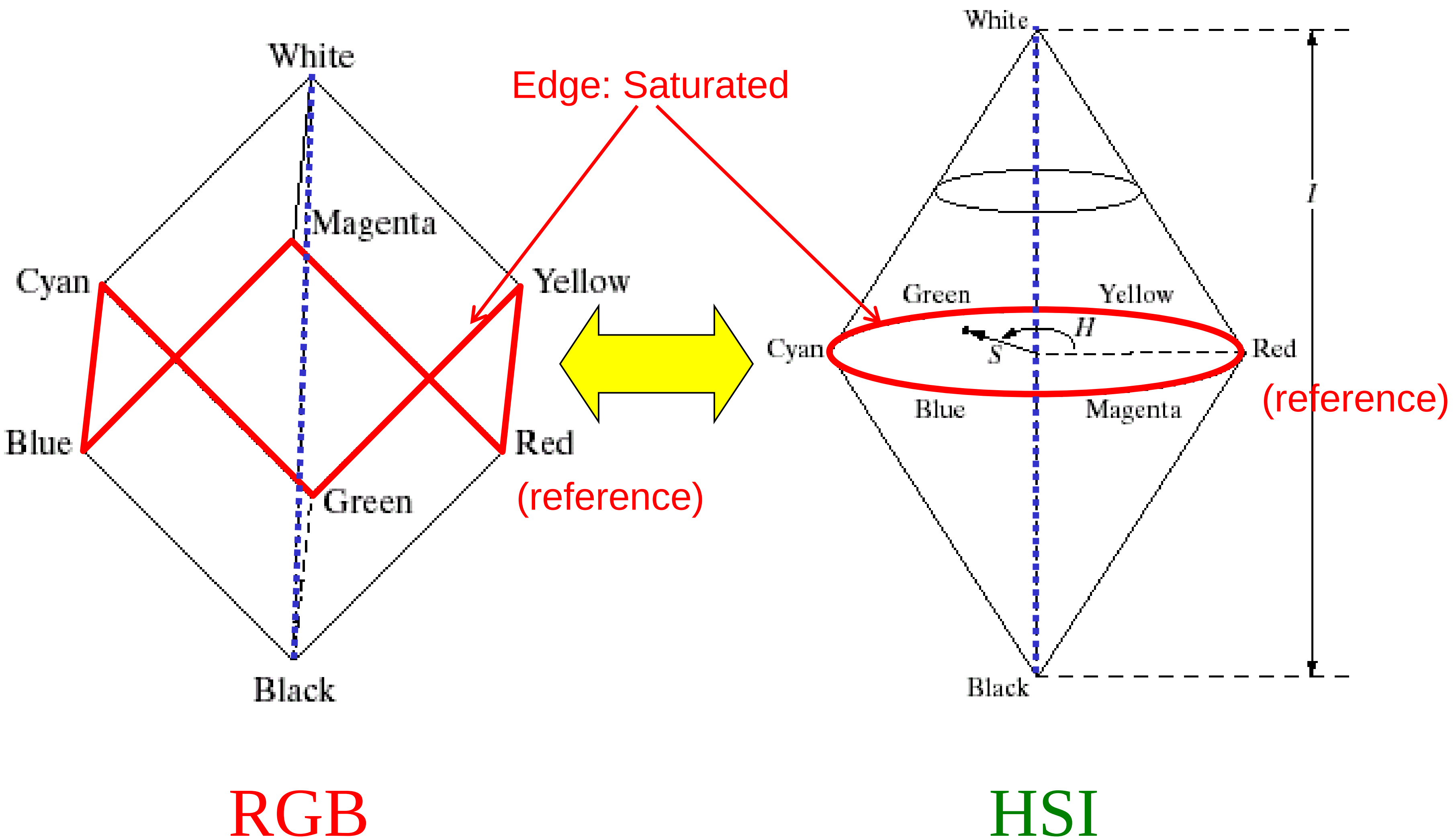
- Since the only important parameters are the angle and the length of the saturation vector this plane is also often represented as a circle or a triangle.

a
b c d

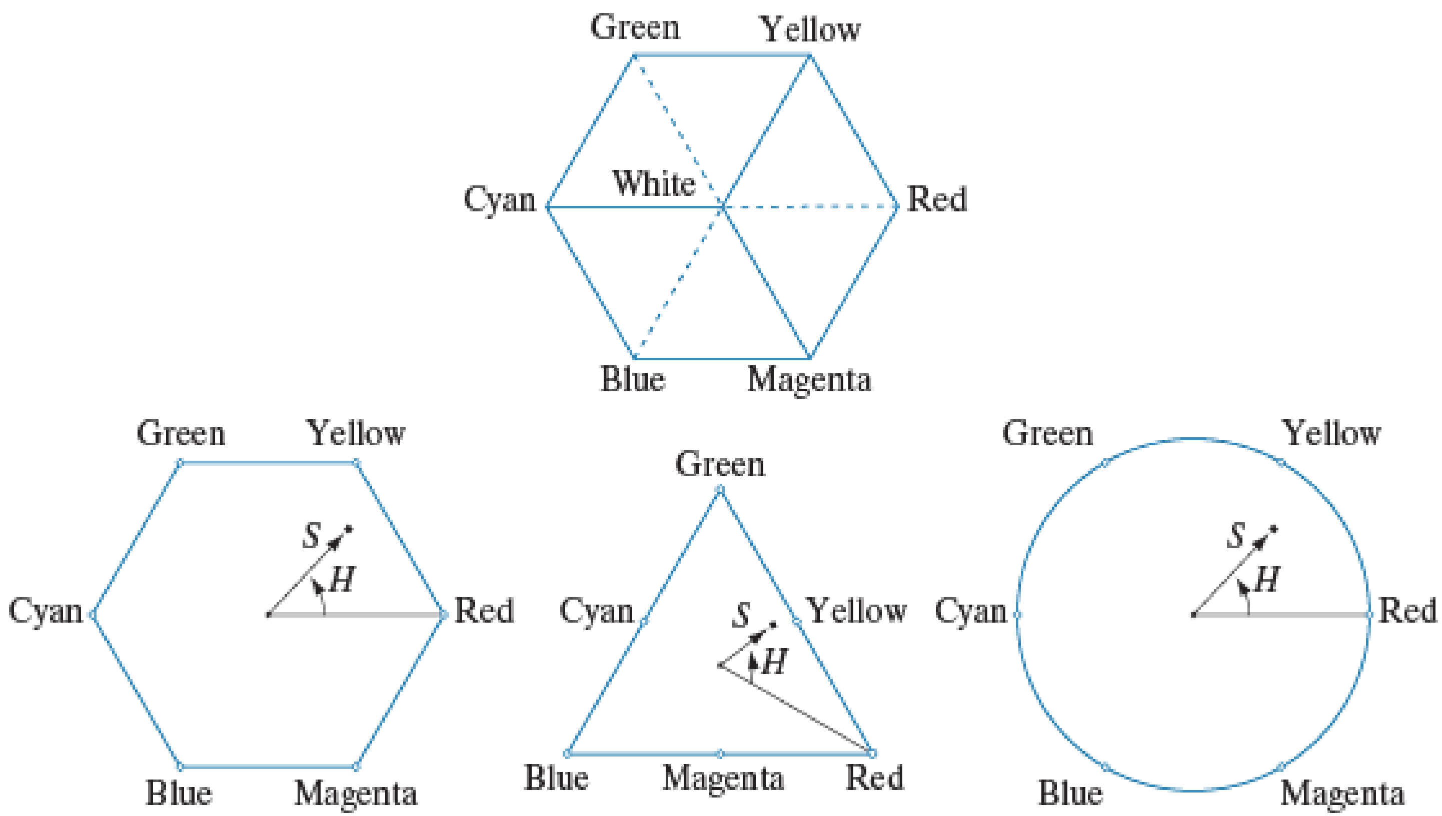
FIGURE 7.11 Hue and saturation in the HSI color model. The dot is any color point. The angle from the red axis gives the hue. The length of the vector is the saturation. The intensity of all colors in any of these planes is given by the position of the plane on the vertical intensity axis.



Relationship Between RGB and HSI Color Models



Hue and Saturation on Color Planes



a
b c d

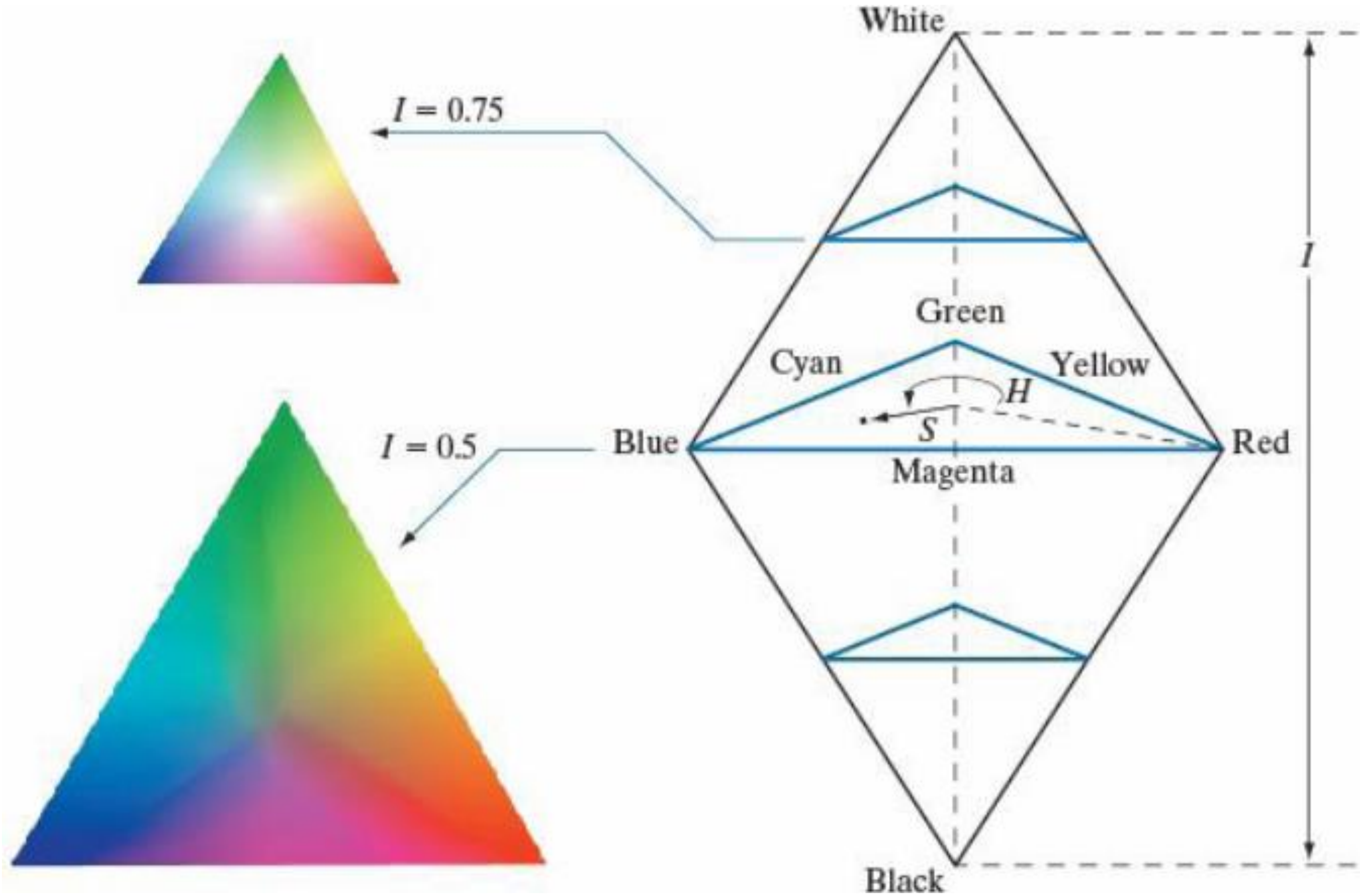
FIGURE 7.11 Hue and saturation in the HSI color model. The dot is any color point. The angle from the red axis gives the hue. The length of the vector is the saturation. The intensity of all colors in any of these planes is given by the position of the plane on the vertical intensity axis.

1. A Dot in the plane: An arbitrary color
2. Hue (Dominant color): An angle from the RED (reference)
3. Saturation: Distance to the Dot from Center (Max at edges)

HSI Color Model (contd..)

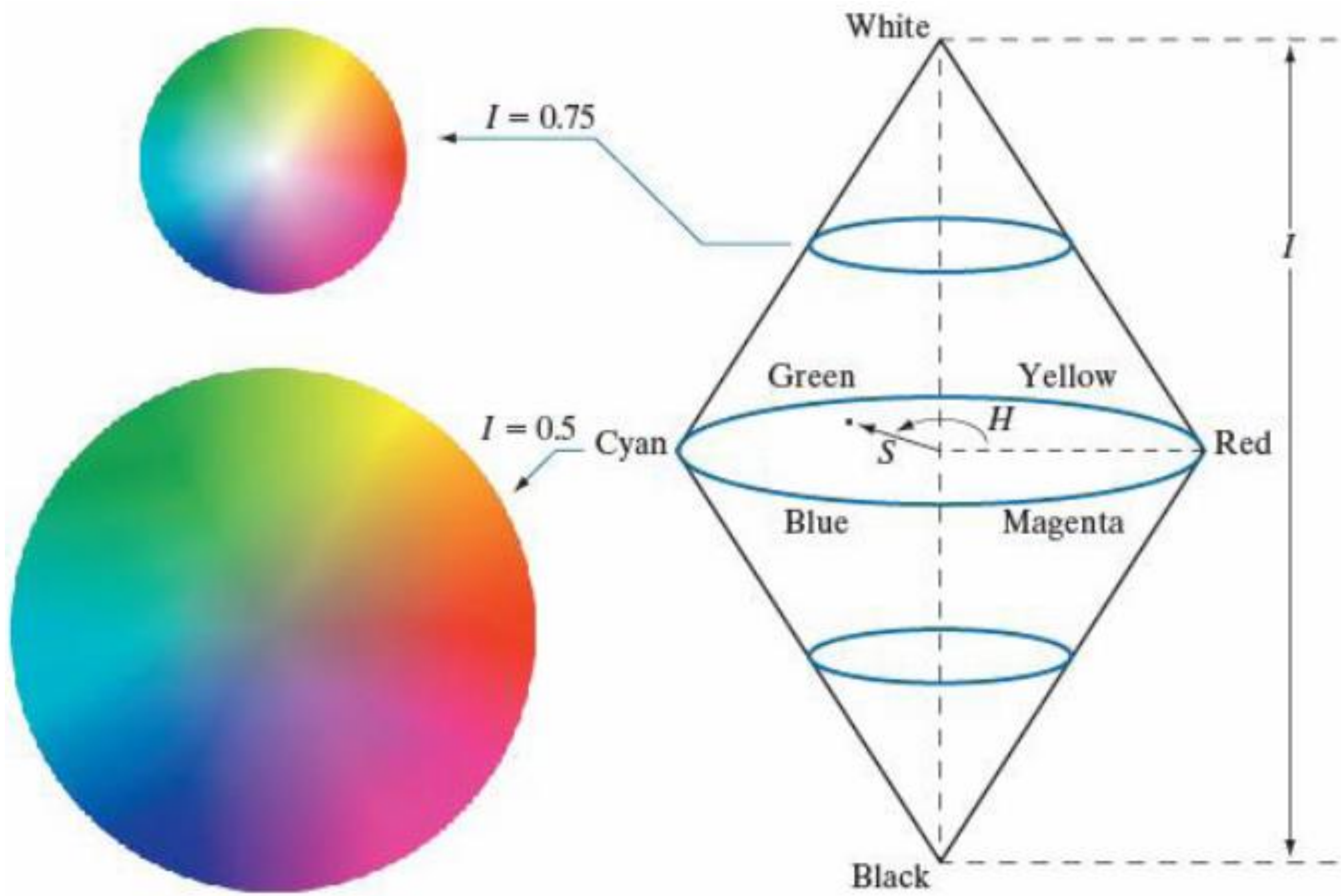
a
b

FIGURE 7.12 The HSI color model based on (a) triangular, and (b) circular color planes. The triangles and circles are perpendicular to the vertical intensity axis.



Intensity is given by a position on the vertical axis.

HSI Color Model (contd..)



a
b

FIGURE 7.12 The HSI color model based on (a) triangular, and (b) circular color planes. The triangles and circles are perpendicular to the vertical intensity axis.

Intensity is given by a position on the vertical axis.

WRIGHT STATE UNIVERSITY

Converting From RGB To HSI

- Given a color as R, G, and B, the H, S, and I values are calculated as follows:

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

$$S = 1 - \frac{3}{(R+G+B)}\left[\min(R,G,B)\right]$$

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

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

Converting 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: $240 \leq H \leq 360$

$$H = H - 240$$

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

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

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

Given a color as H, S, and I the R, G, and B values are found using these equations

WRIGHT STATE UNIVERSITY

HSI & RGB

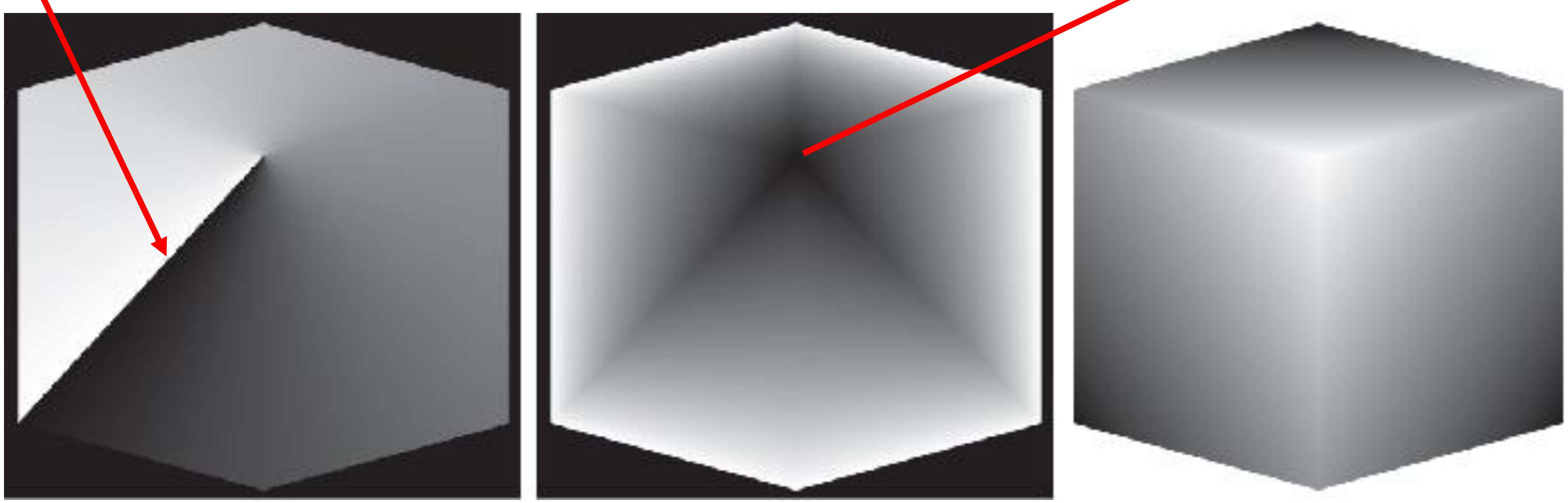
H, S, and I Components of RGB Color Cube

FIGURE 7.8 A 24-bit RGB color cube.

RGB Color Cube

Cross-over from 0° (white) to 360° (black)

Low → Less saturated



Hue Saturation Intensity

a b c

FIGURE 7.13 HSI components of the image in Fig. 7.8: (a) hue, (b) saturation, and (c) intensity images.

WRIGHT STATE UNIVERSITY

HSI & RGB

H, S, and I Components of RGB Color Cube

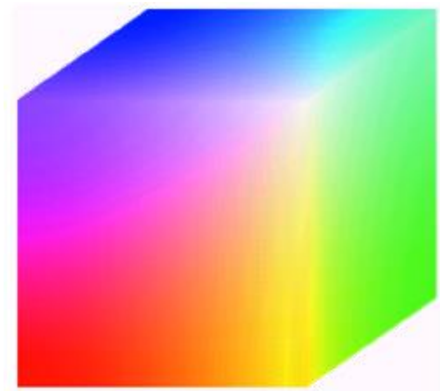
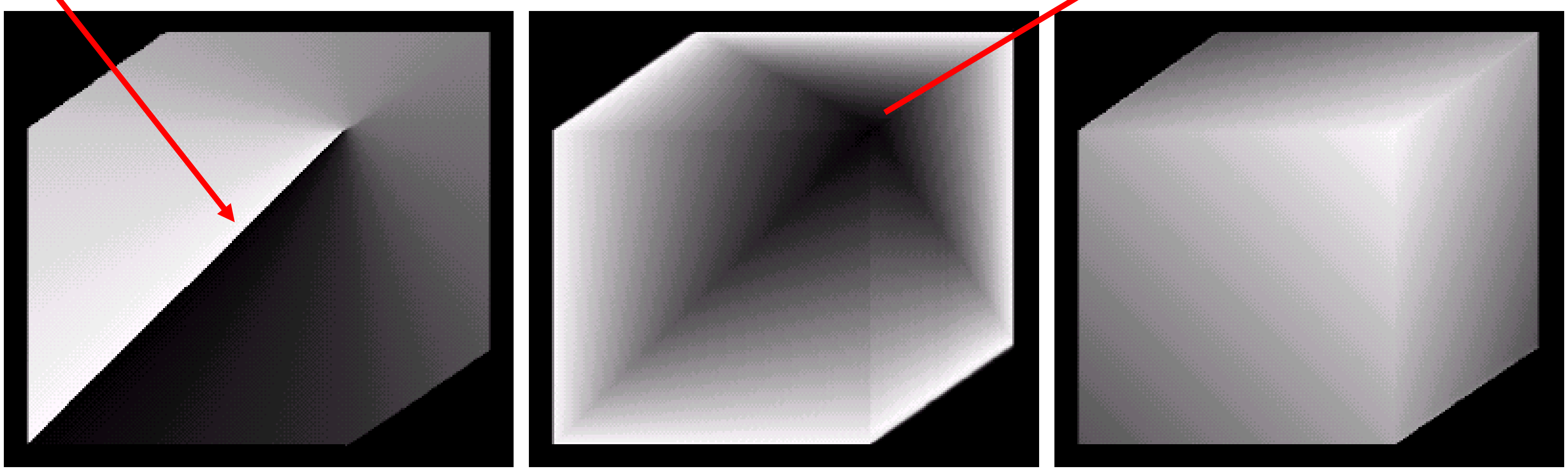


FIGURE 6.8 RGB 24-bit color cube.

RGB Color Cube

Cross-over from 0° (white) to 360° (black)

Low → Less saturated



Hue | Saturation | Intensity

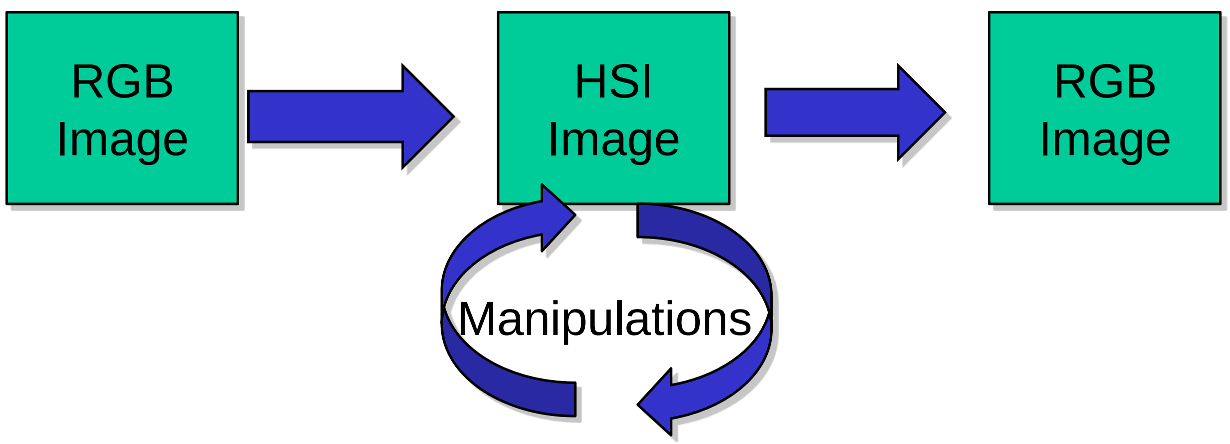
a b c

3rd Edition

FIGURE 6.15 HSI components of the image in Fig. 6.8. (a) Hue, (b) saturation, and (c) intensity images.

Manipulating Images in the HSI Model

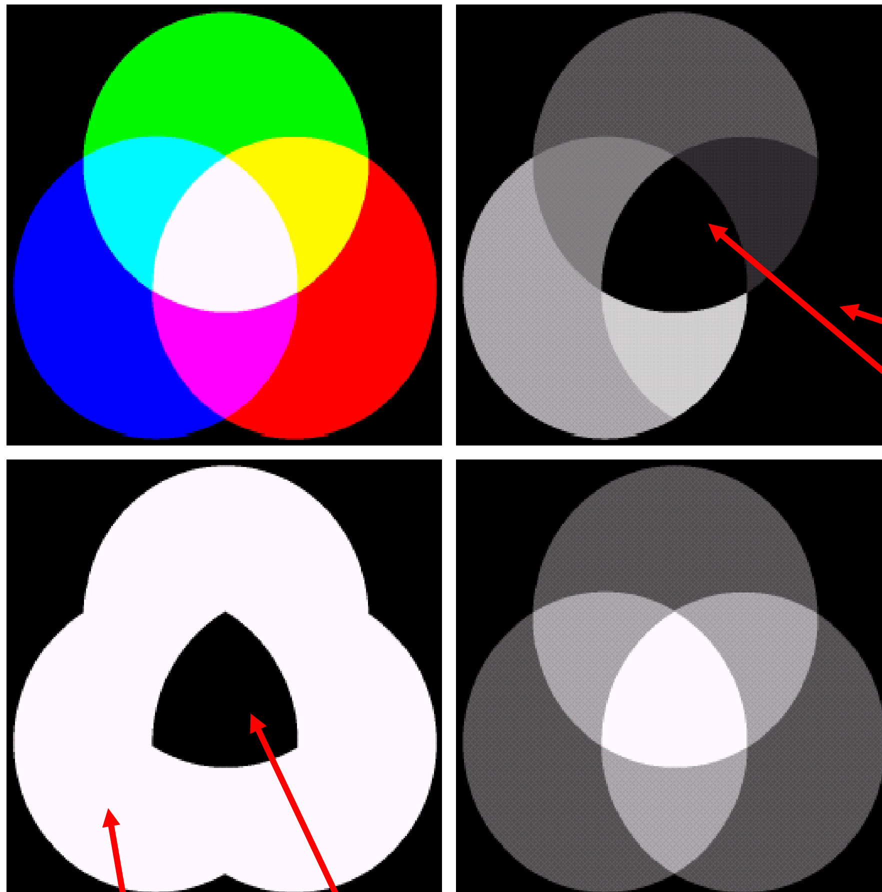
- Steps for manipulating an image under the HSI model:
  - First convert image from RGB to HSI
  - Perform manipulations under HSI as desired
  - Finally convert the image back from HSI to RGB



RGB -> HSI -> RGB

RGB Image: Primary and Secondary Colors

Hue (angles from Red)



0° for Red and peak

Saturation (distance from origin)

Intensity (distance from pure black at bottom)

Max at edges

0 at top peak

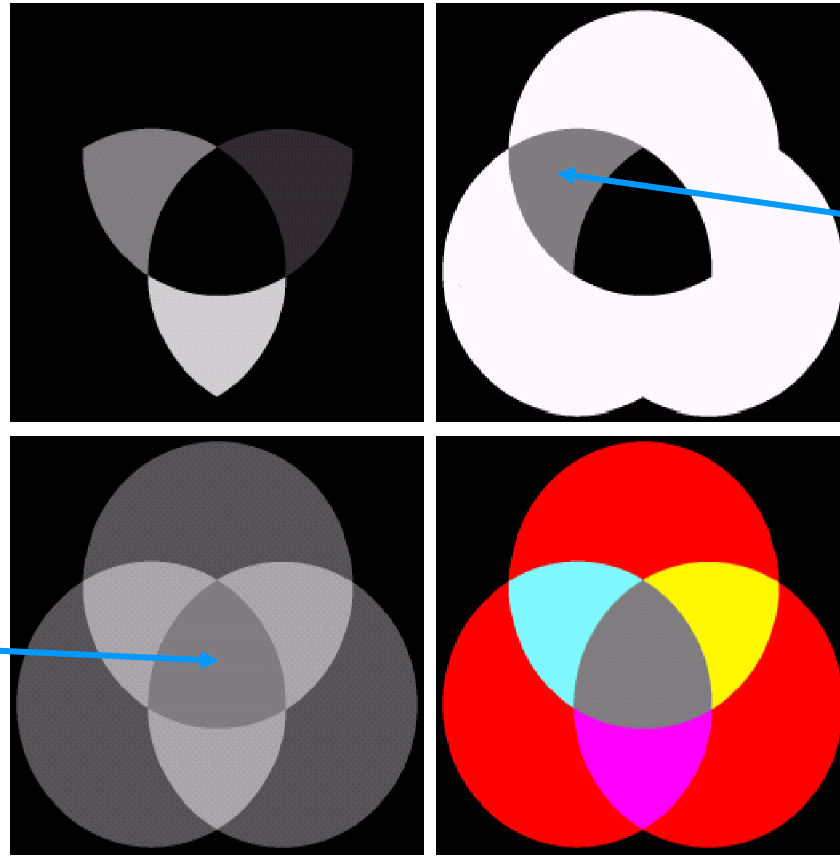
a b
c d

FIGURE 7.14 (a) RGB image and the components of its corresponding HSI image: (b) hue, (c) saturation, and (d) intensity.

Manipulating HSI Components of Images

Hue component: Change blue and green pixels to 0 in Fig 7.14(b) → converts to Red

Saturation Component: Reduce by half the saturation of the cyan [1 → 0.5] region in Fig 7.14(c)



Intensity component: Reduce by half the intensity of the central white region [1 → 0.5] in Fig 7.14(d)

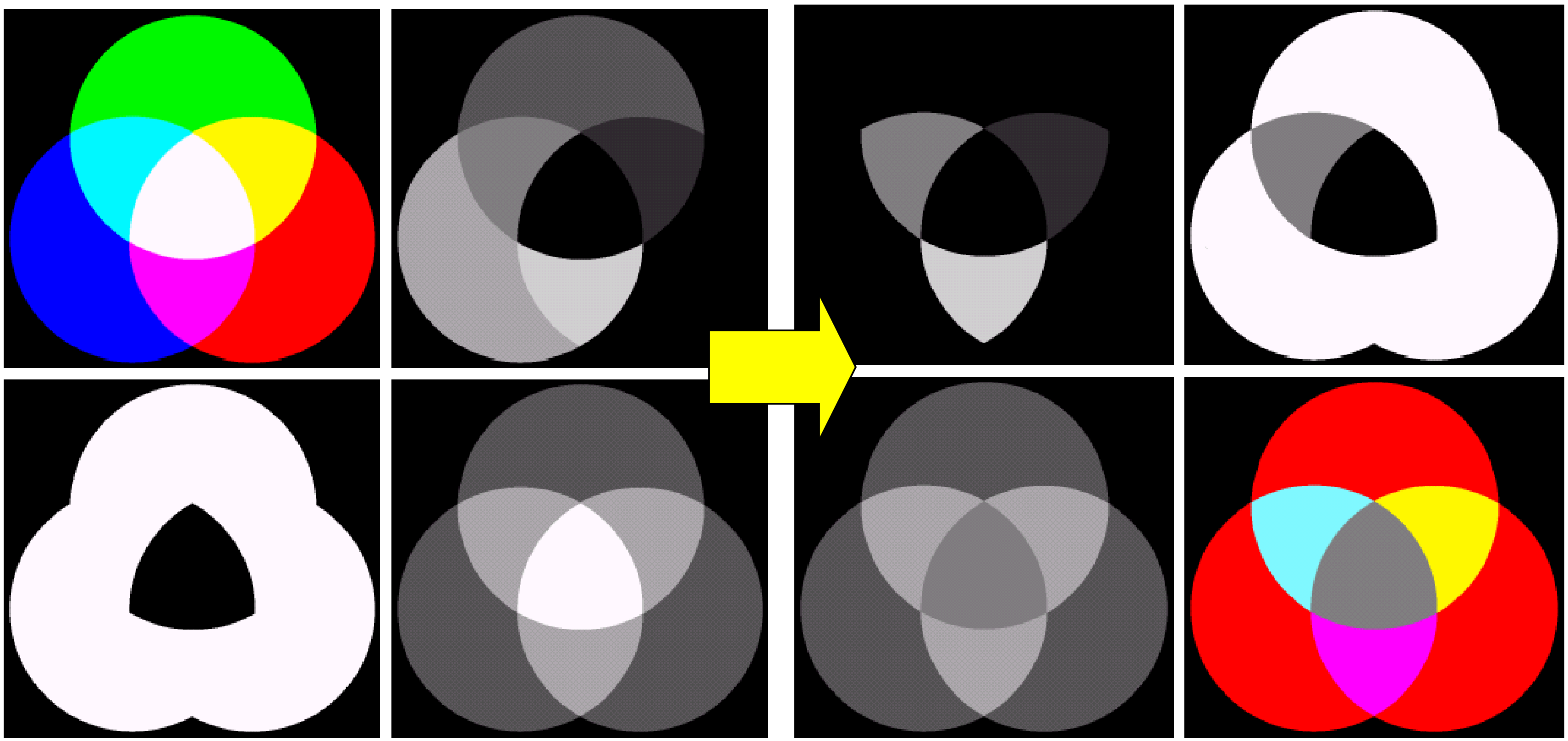
The modified result image

a b
c d

FIGURE 7.15
(a)-(c) Modified HSI component images.
(d) Resulting RGB image. (See Fig. 7.14 for the original HSI images.)

Example: Manipulating HSI Components

RGB Image | Hue | Hue | Saturation



Saturation | Intensity | Intensity | RGB Image

Color Image Processing

Two types of color image processing

- **Pseudocolor image processing:** Assign colors to gray values based on a specific criterion
  - Gray scale images to be processed may be a single image or multiple bands in images such as multispectral or hyperspectral images.
- **Full color image processing:** Manipulate real color images such as color photographs

7.3 Pseudocolor Image Processing

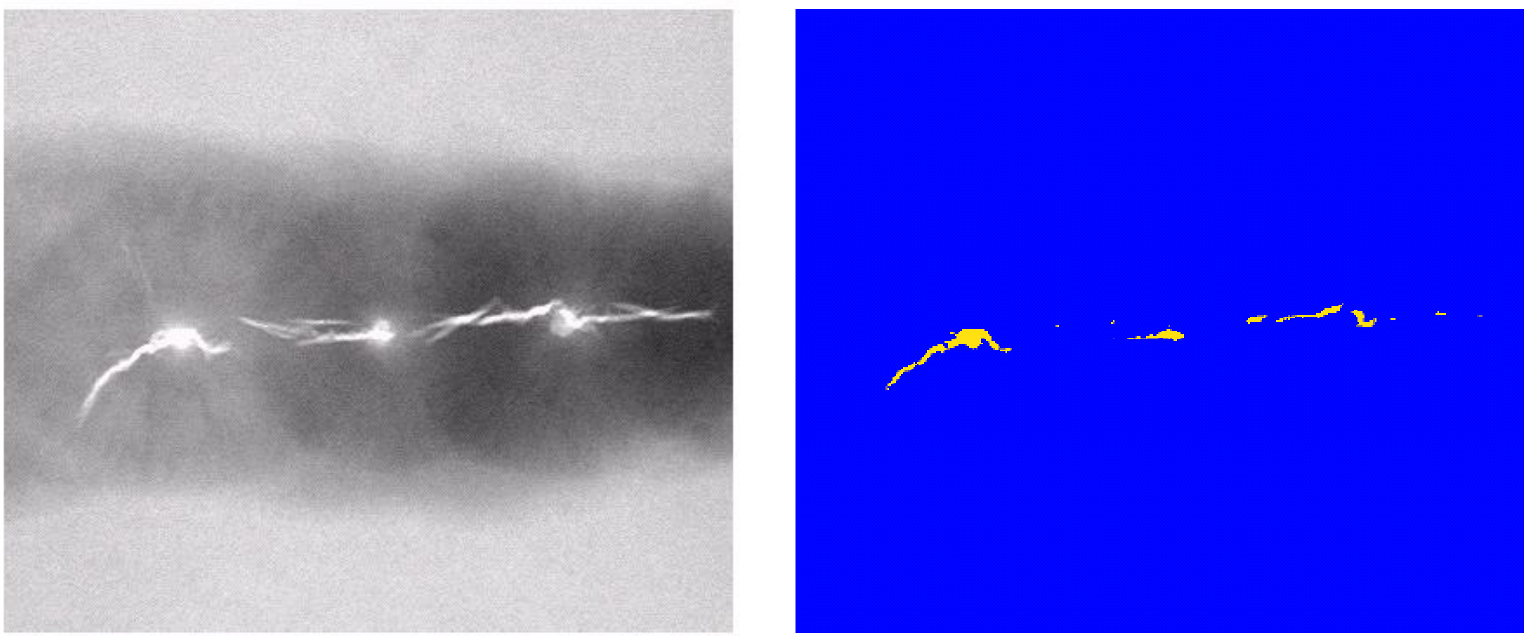
Pseudo color = false color : In some cases there is no "color" concept for a gray scale image. But "false" colors can be assigned to an image.

Why we need to assign colors to gray scale image?

Answer: Human eyes can distinguish different colors better than multiple shades of gray.

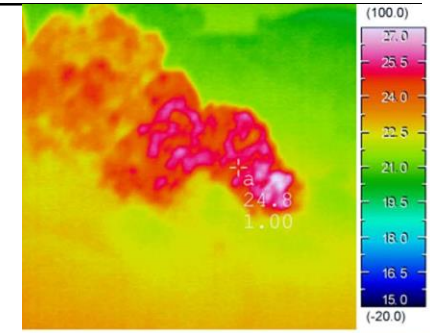
a b

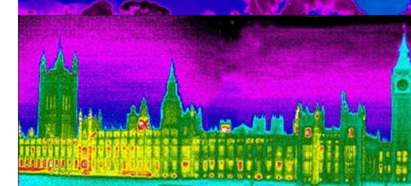
FIGURE 7.19
(a) X-ray image of a weld.
(b) Result of color coding. (Original image courtesy of X-TEK Systems, Ltd.)



7.3 Pseudocolor Image Processing

- Pseudocolor (also called false color) image processing consists of assigning colors to gray values based on a specific criterion.
- The principle use of Pseudocolor image processing is for human visualization.
  - Humans can discern between thousands of color shades and intensities, compared to only about two dozen or so shades of gray





7.3 Pseudo Color Image Processing – Intensity Slicing

- Intensity Slicing and Color Coding: Simplest kind of pseudocolor image processing.
- First consider an image as a 3D function mapping spatial coordinates to intensities
  - Can be considered as heights:
    - ✓ Black → Minimum intensity
    - ✓ White → Maximum intensity
- Consider placing planes at certain levels parallel to the coordinate plane.
- If a value is on one side of such a plane it is rendered in one color, but a different color if on the other side.

Intensity Slicing or Density Slicing

Formula:

$$f(x,y)=\begin{cases} c_1 & \text{if } f(x,y)\le l_1 \\ c_2 & \text{if } f(x,y)> l_1 \end{cases}$$

$\rightarrow c_1$ = Color No. 1

$\rightarrow c_2$ = Color No. 2

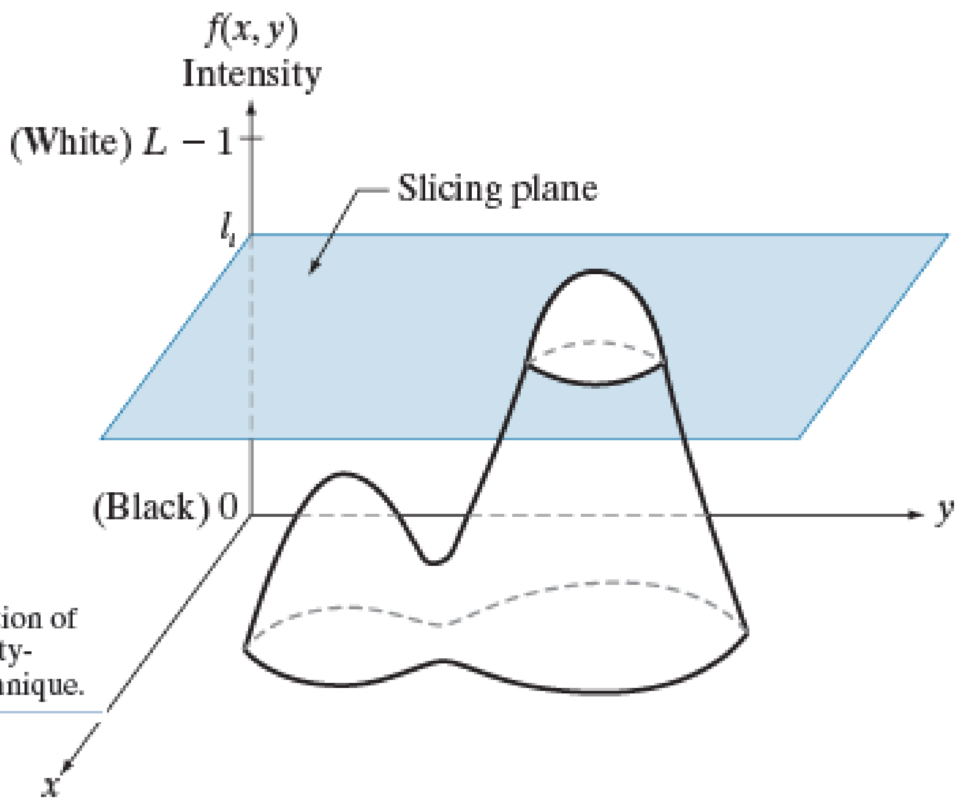


FIGURE 7.16 Graphical interpretation of the intensity-slicing technique.

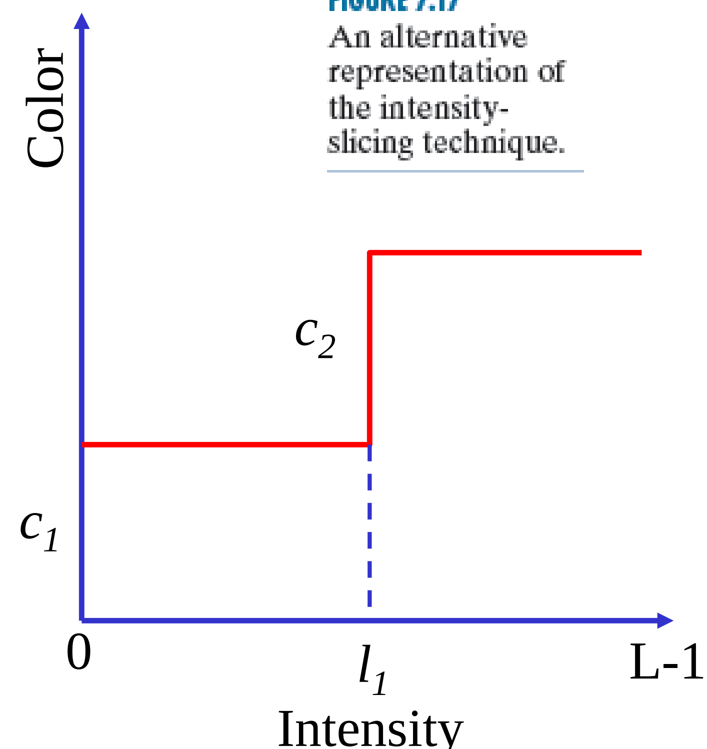
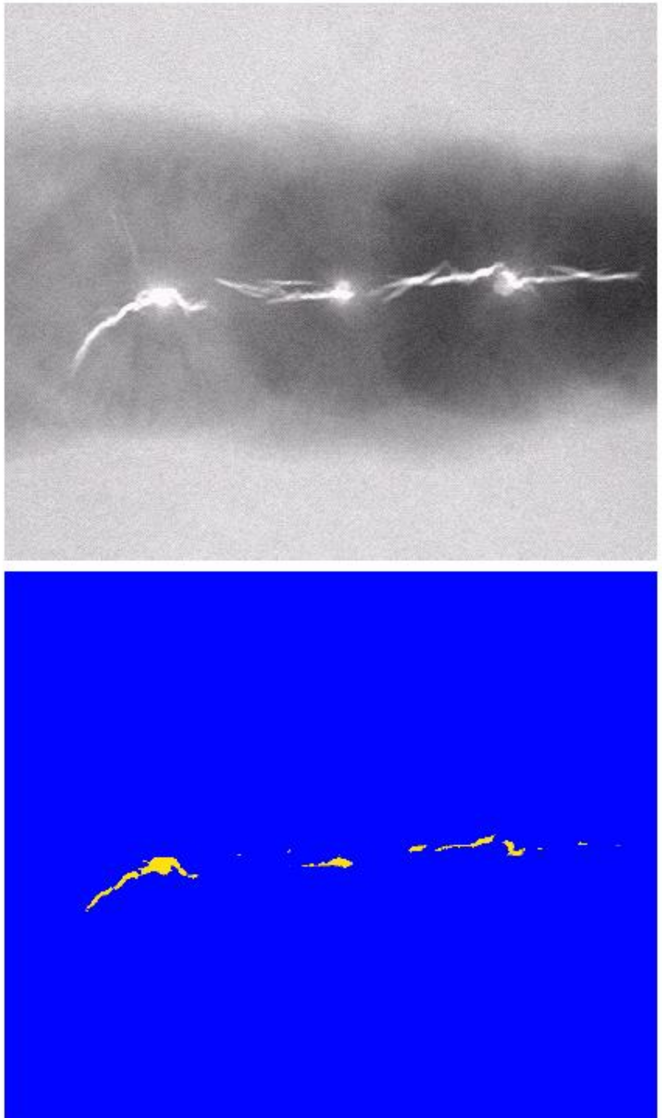


FIGURE 7.17 An alternative representation of the intensity-slicing technique.

A gray scale image viewed as a 3D surface.

Intensity Slicing Example



An X-ray image of a weld with cracks

a b

FIGURE 7.19
(a) X-ray image of a weld.
(b) Result of color coding. (Original image courtesy of X-TEK Systems, Ltd.)

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

Pseudocolor Image Processing – Intensity Slicing (cont…)

Summary of more general Intensity Slicing :

- Let $[0, L\text{-}1]$ represent the grey scale
- Let l_0 represent Black $[f(x, y) = 0]$ and let $l_{L\text{-}1}$ represent White $[f(x, y) = L\text{-}1]$
- Suppose P planes perpendicular to the intensity axis are defined at levels $l_1, l_2, \ldots, l_P$
- Assume $0 < P < L\text{-}1$: P-planes partition the grey scale into $P + 1$ intervals $V_1, V_2, \ldots, V_{P+1}$

Pseudocolor Image Processing – Intensity Slicing (cont…)

- Grey level color assignments can then be made according to the relation:

$$f(x, y) = c_k \qquad \text{if } f(x, y) \in V_k$$

where,

- c_k : Color associated with the k^{th} intensity level V_k

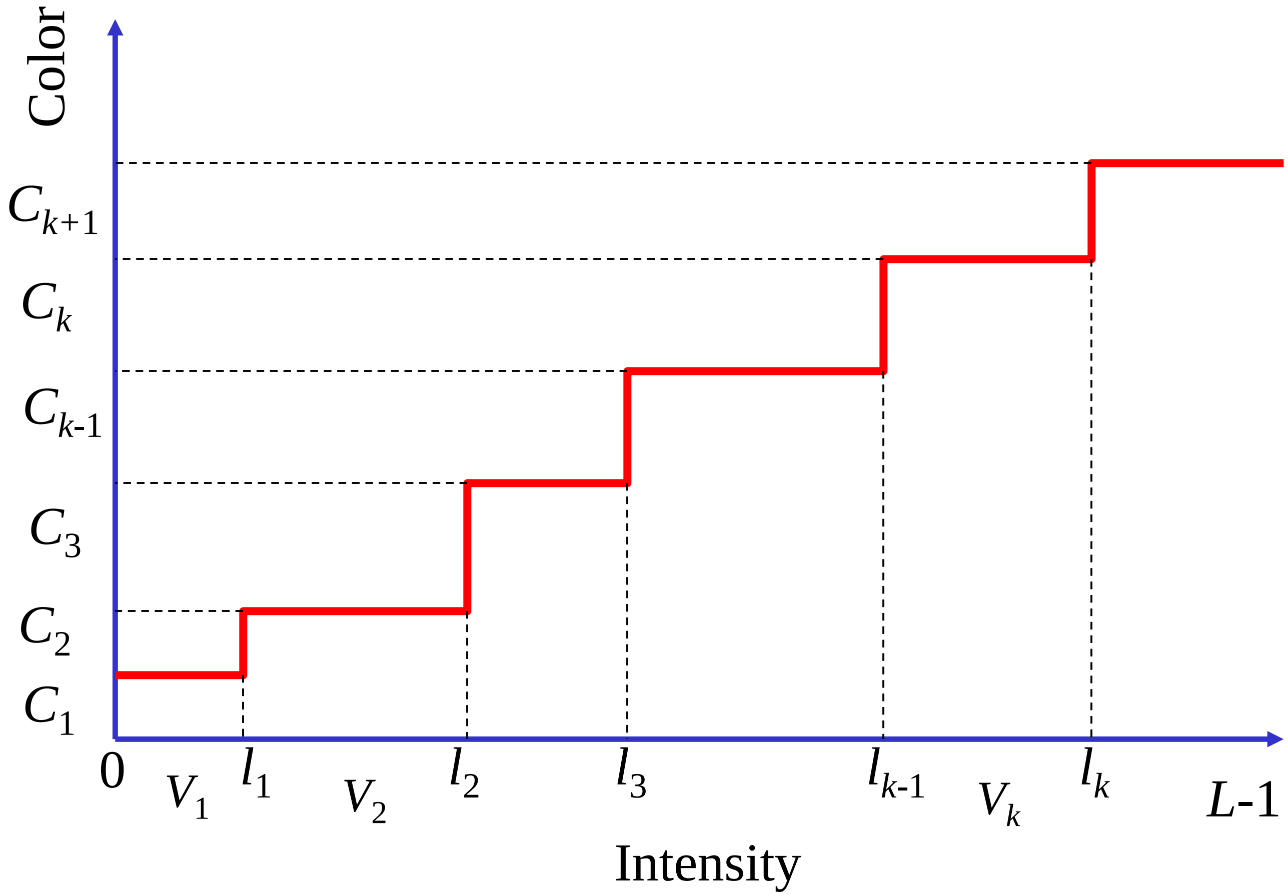
- V_k : Defined by the partitioning planes between $l = k - 1$ and $l = k$

Multi-Level Intensity Slicing

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

C_k = Color No. k

l_k = Threshold level k



Multi Level Intensity Slicing Example

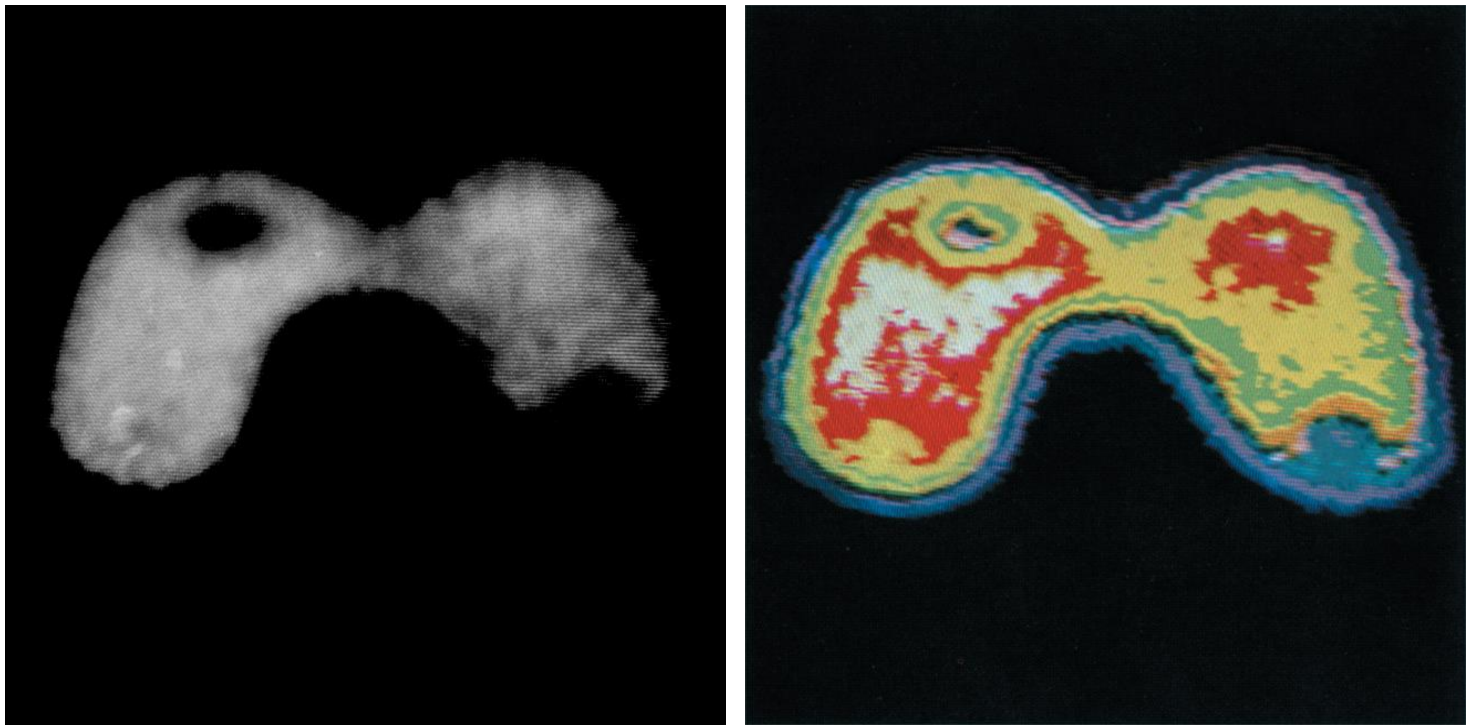
$$g(x,y) = C_k \qquad \text{for } l_{k-1} < f(x,y) \le l_k$$

C_k = Color No. k

l_k = Threshold level k

a b

FIGURE 7.18
(a) Grayscale image of the Picker Thyroid Phantom.
(b) Result of intensity slicing using eight colors.
(Courtesy of Dr. J. L. Blankenship, Oak Ridge National Laboratory.)

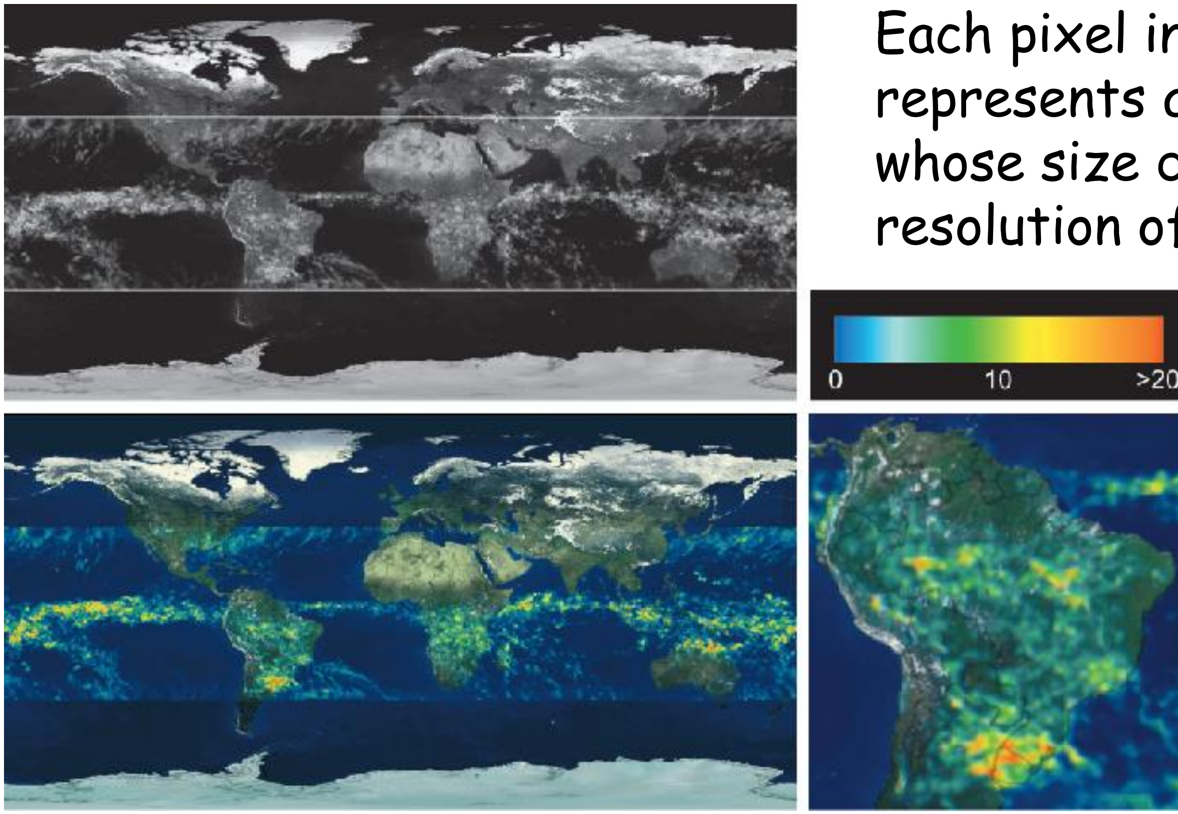


An X-ray image of the Picker Thyroid Phantom

After density slicing into 8 colors

Pseudocolor Image Processing for Rainfall Detection

Each pixel in these images represents a physical land area whose size depends on the resolution of the sensors.



Gray-scale image of average monthly rainfall.

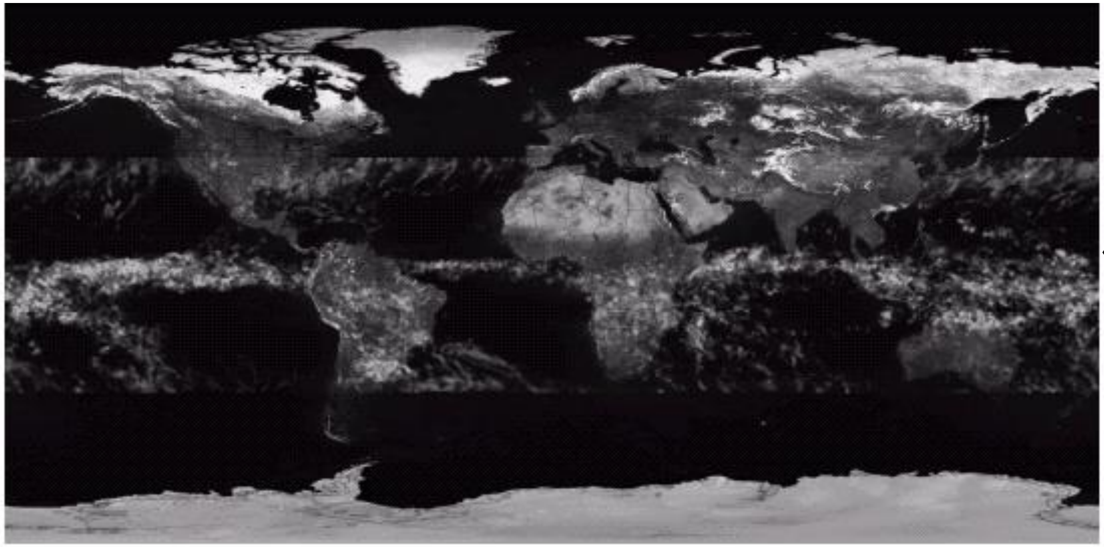
South America region

Color coded image

a b
c d

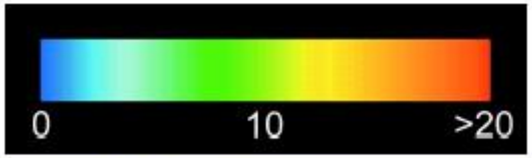
FIGURE 7.20 (a) Grayscale image in which intensity (in the horizontal band shown) corresponds to average monthly rainfall. (b) Colors assigned to intensity values. (c) Color-coded image. (d) Zoom of the South American region. (Courtesy of NASA.)

Color Coding Example

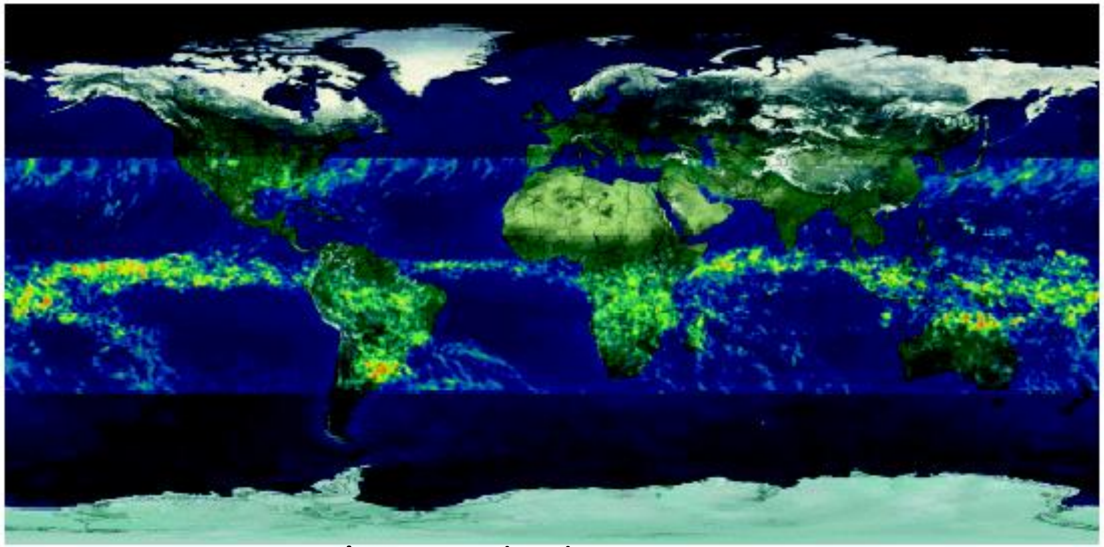


A unique color is assigned to each intensity value.

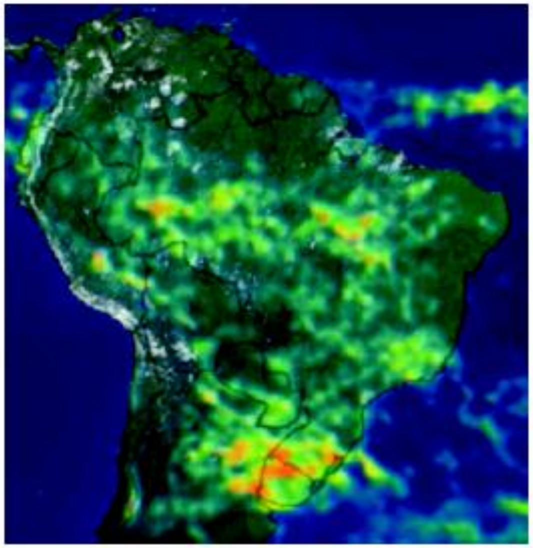
Gray-scale image of average monthly rainfall.



Color map



Color coded image



South America region

WRIGHT STATE UNIVERSITY

7.3 Gray Level to Color Transformation

Assigning colors to gray levels based on specific mapping functions

Gray scale image

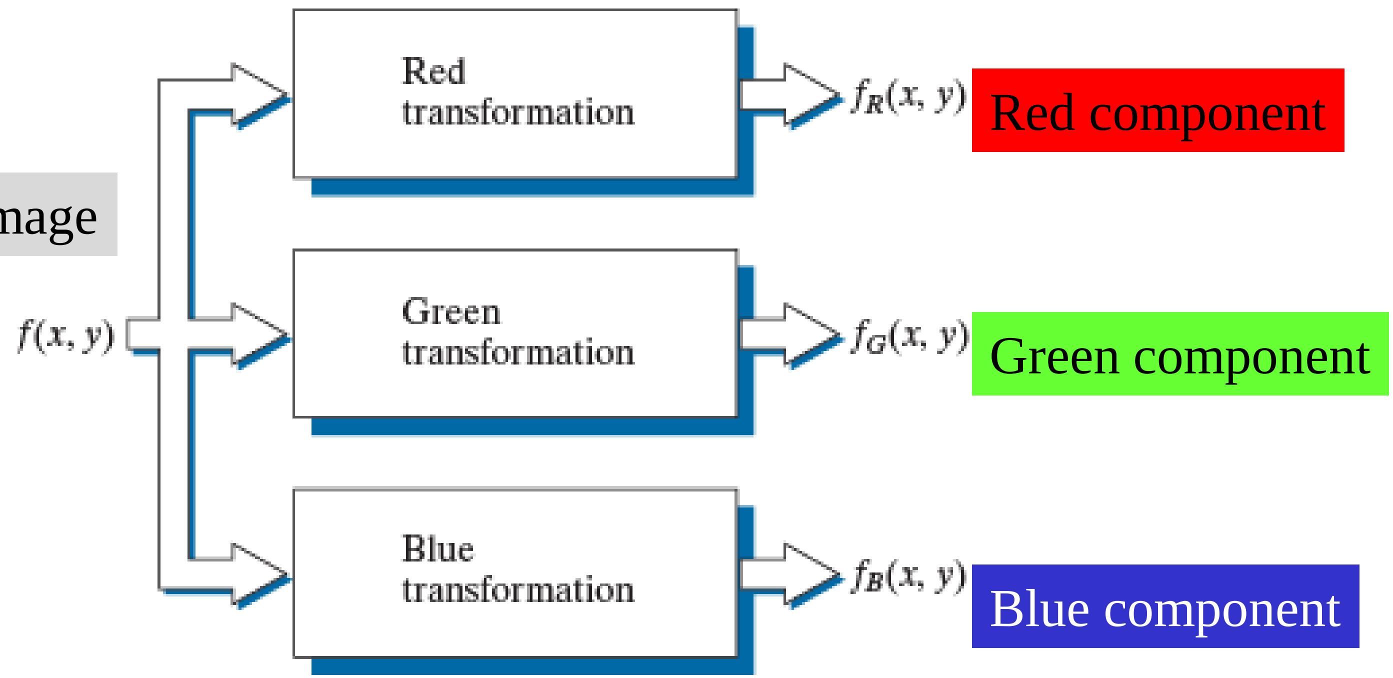
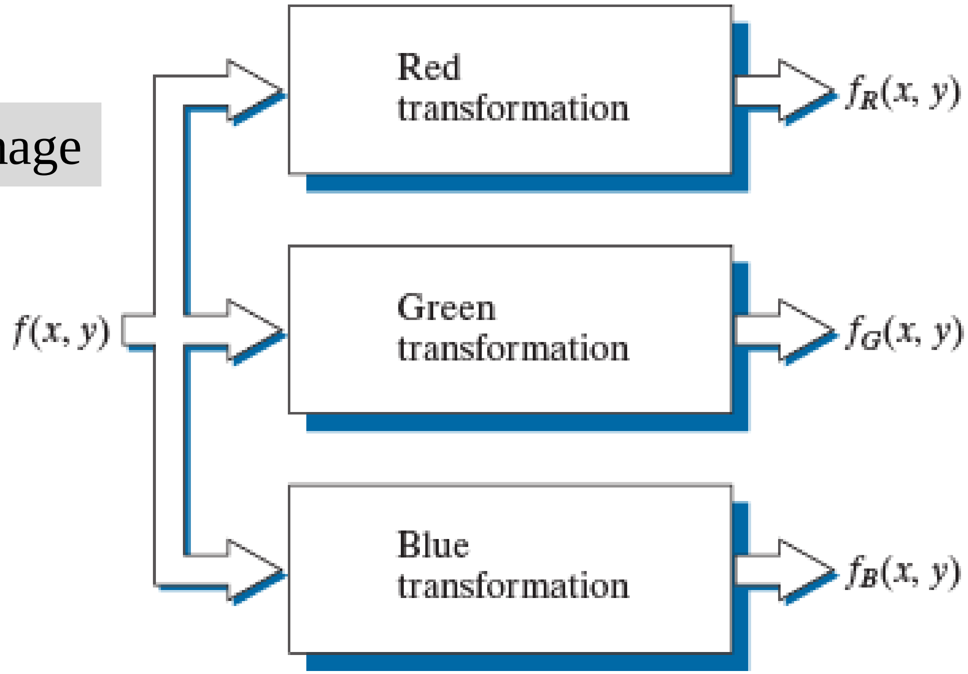


FIGURE 7.21 Functional block diagram for pseudocolor image processing. Images f_R, f_G, and f_B are fed into the corresponding red, green, and blue inputs of an RGB color monitor.

- Gray level to color transformation for achieving a wider range of pseudocolor enhancement results.

7.3 Pseudocolor Image Processing Intensity to Color Transformation

Gray scale image



Red component

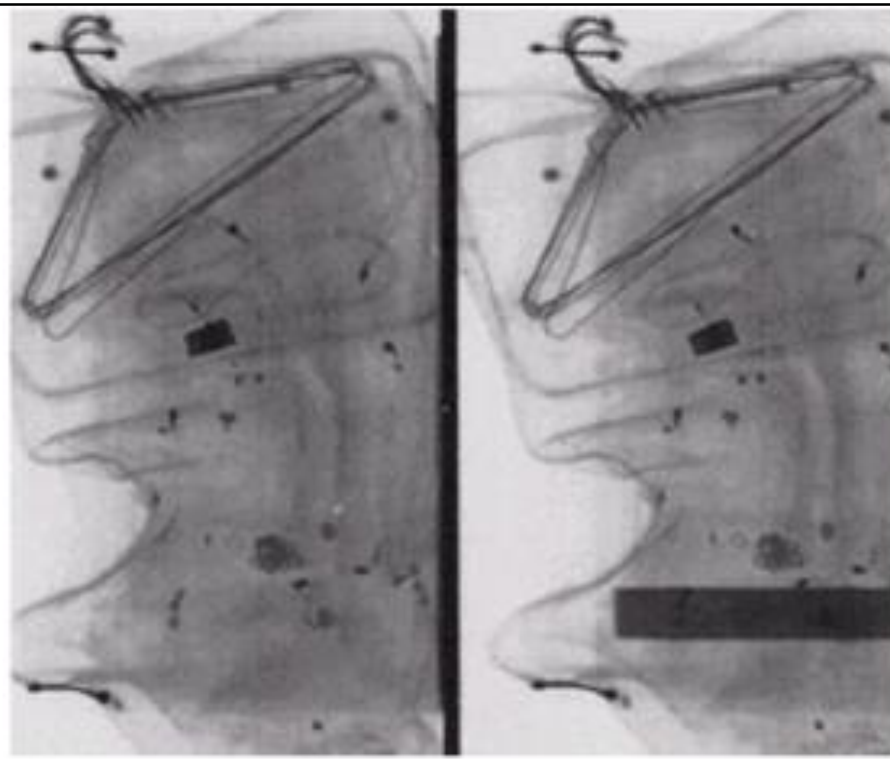
Green component

Blue component

FIGURE 7.21 Functional block diagram for pseudocolor image processing. Images f_R, f_G, and f_B are fed into the corresponding red, green, and blue inputs of an RGB color monitor.

- Three *independent transformations* of intensity
- Results fed into R, G, B channels *separately*
- Resulting composite image highlights certain image parts

Pseudocolor Image Processing
Intensity to Color Transformation (cont.)



a
b c

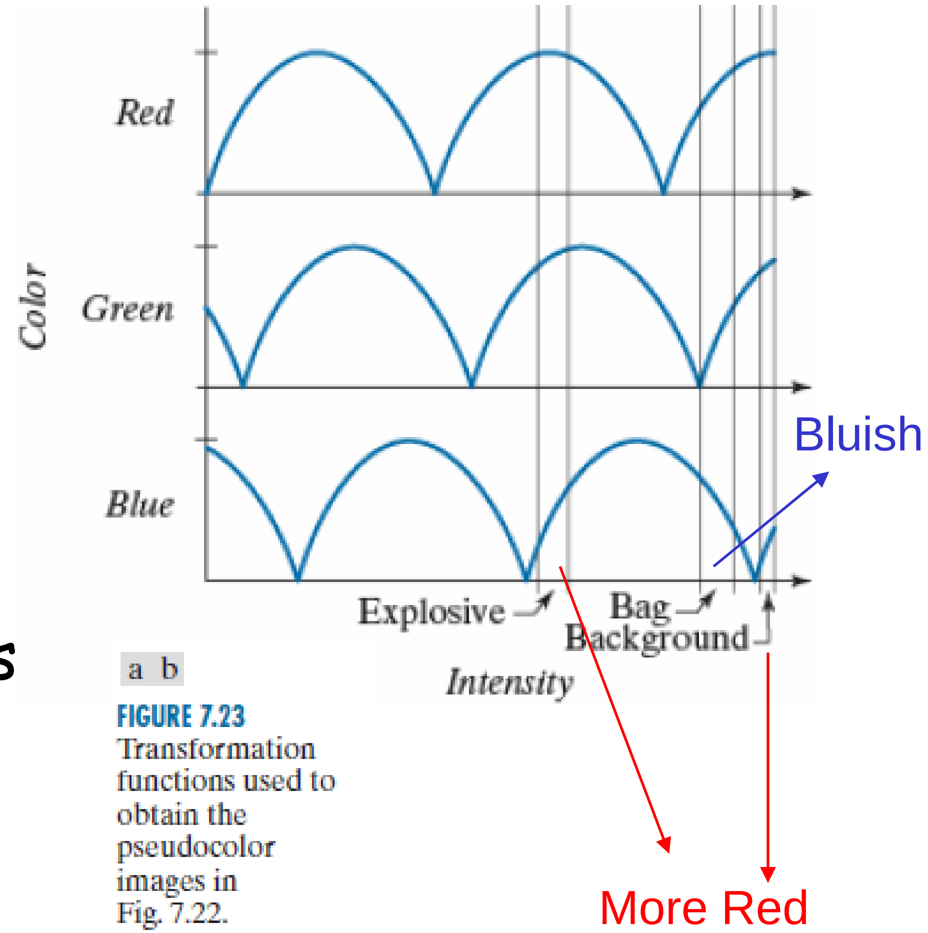
FIGURE 7.22 Pseudocolor enhancement by using the gray level to color transformations in Fig. 7.23. (Original image courtesy of Dr. Mike Hurwitz, Westinghouse.)

- X-ray images from airport scanning system
- The image on the right contains *simulated plastic explosives*

Pseudocolor Image Processing
Intensity to Color Transformation (cont.)

- Sinusoidal transformation functions
- Changing the phase or the frequency of the transformation functions can emphasize ranges in the gray scale:
  - A small change in the phase between the transformations assigns a strong color to the pixels with intensities in the valleys.

3 Independent Transformations



a b

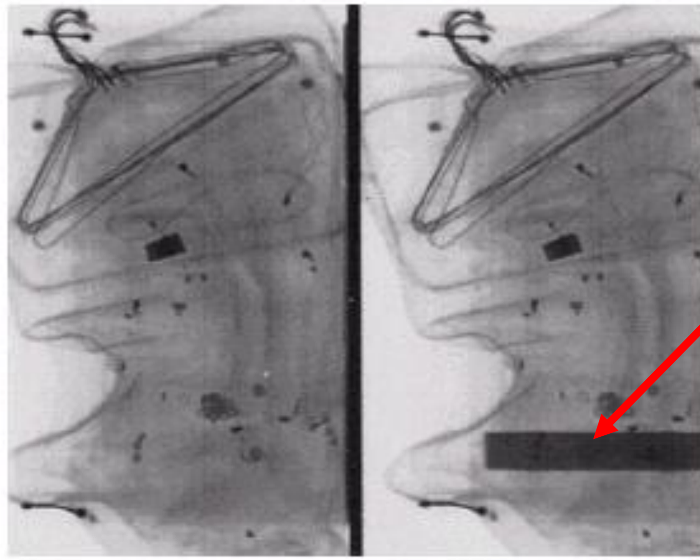
FIGURE 7.23 Transformation functions used to obtain the pseudocolor images in Fig. 7.22.

Gray Level to Color Transformation-1

An X-ray image of a garment bag

a
b c

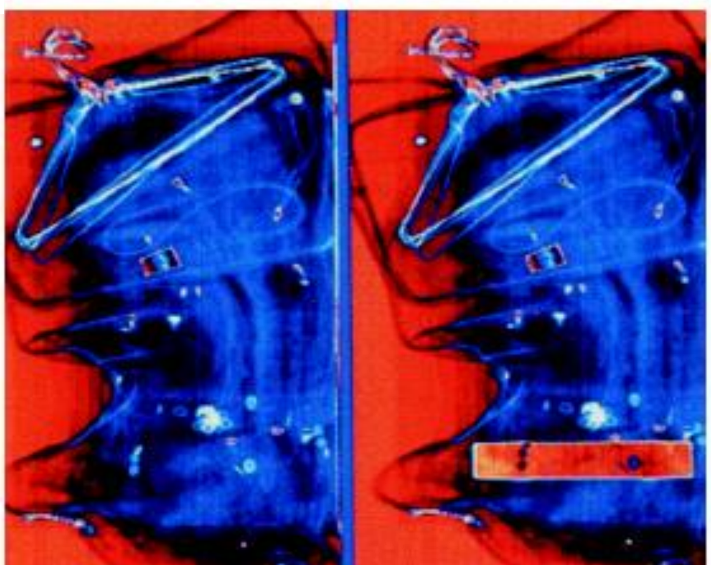
FIGURE 7.22 Pseudocolor enhancement by using the gray level to color transformations in Fig. 7.23. (Original image courtesy of Dr. Mike Hurwitz, Westinghouse.)

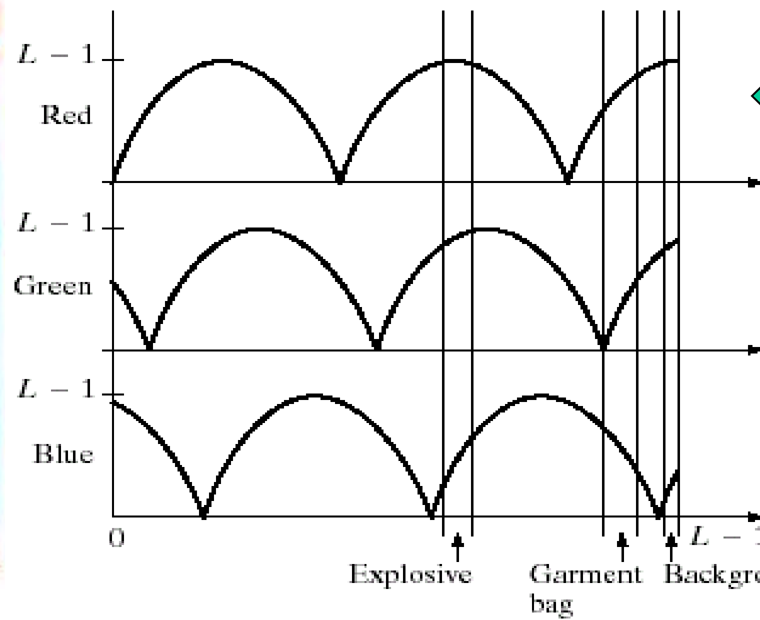


An X-ray image of a garment bag with a simulated explosive device

3 Independent Transformations

Color coded images



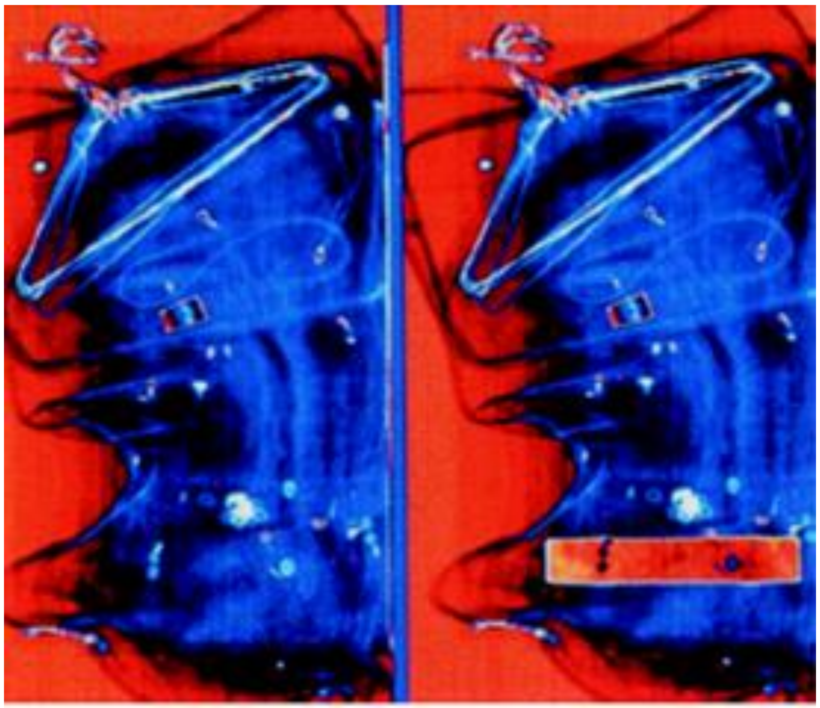


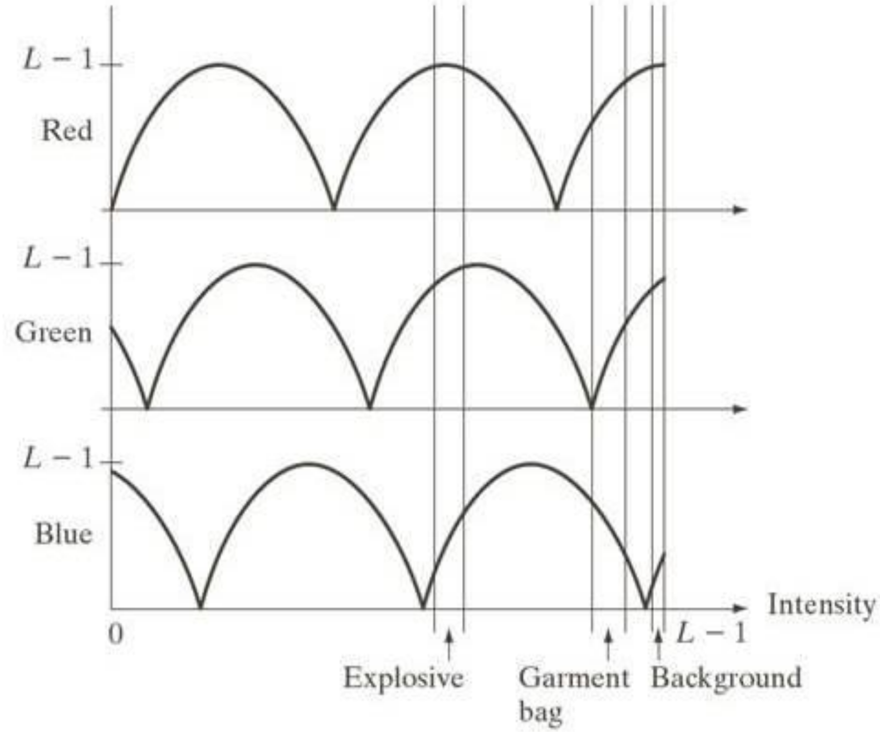
a b

FIGURE 7.23 Transformation functions used to obtain the pseudocolor images in Fig. 7.22.

Pseudocolor Image Processing
Intensity to Color Transformation (cont.)

3 Independent Transformations



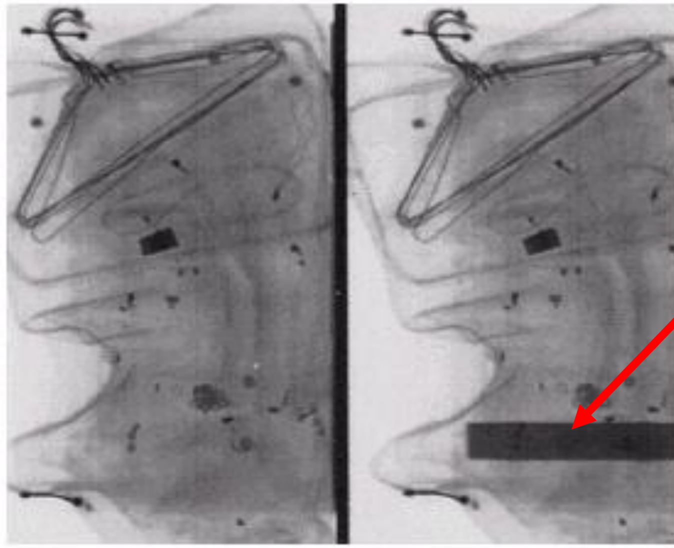


- Background and explosives are coded with approximately the same color although they differ
- This is due to the periodicity of the sine waves
- Done on a case-by-case basis

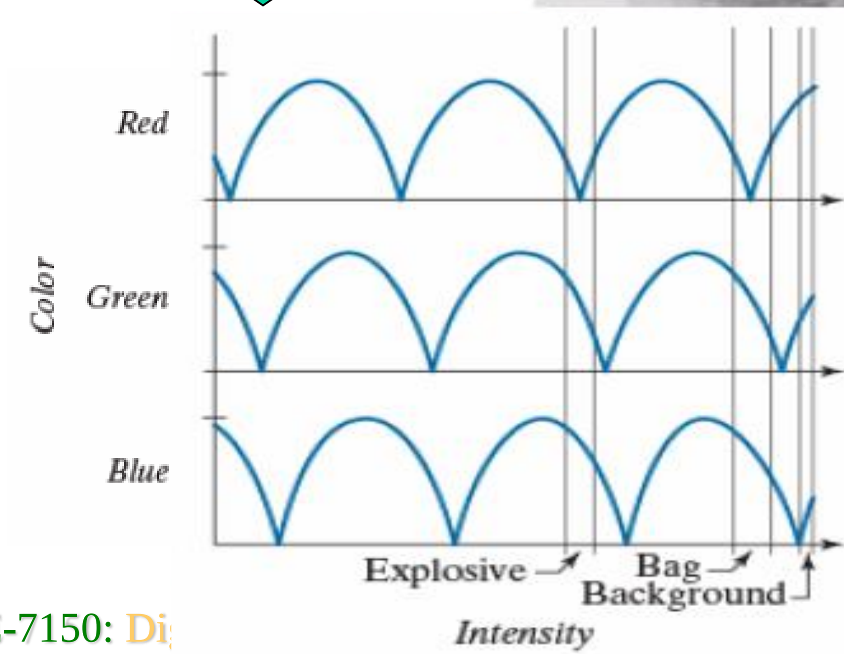
Gray Level to Color Transformation-2

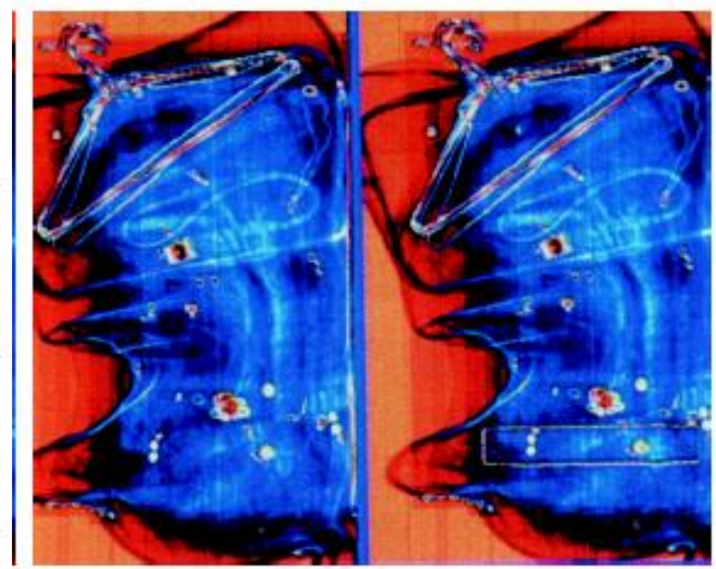
An X-ray image of a garment bag

An X-ray image of a garment bag with a simulated explosive device



Transformations

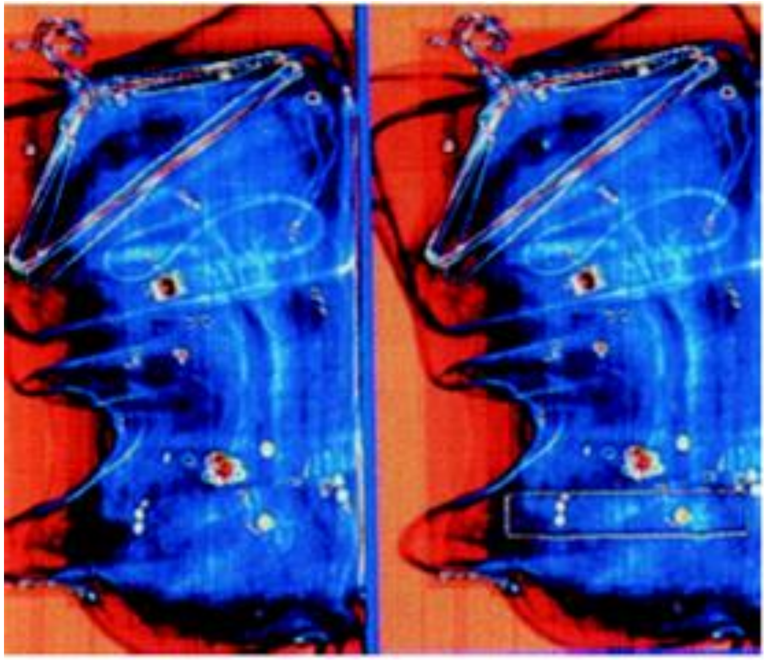


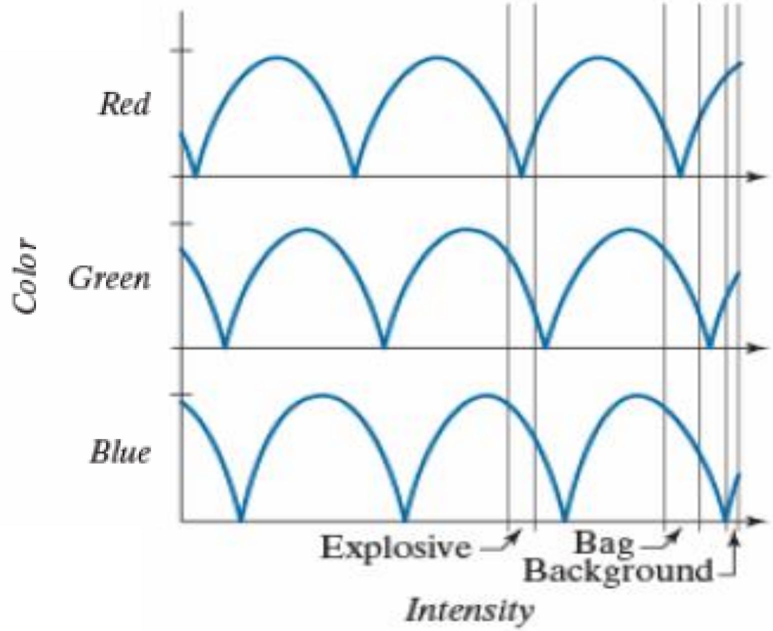


Color coded images

Pseudocolor Image Processing
Intensity to Color Transformation (cont.)

3 Independent Transformations





a b

FIGURE 7.23 Transformation functions used to obtain the pseudocolor images in Fig. 7.22.

- Explosives and bag content are mapped by similar transformations and were assigned to the same color.
- The observer may "see through" the explosives and not mistake them for the background.

Pseudocolor Coding

- Used if there are multiple monochrome images such as multispectral satellite images
  - Sensors collecting at different spectral bands

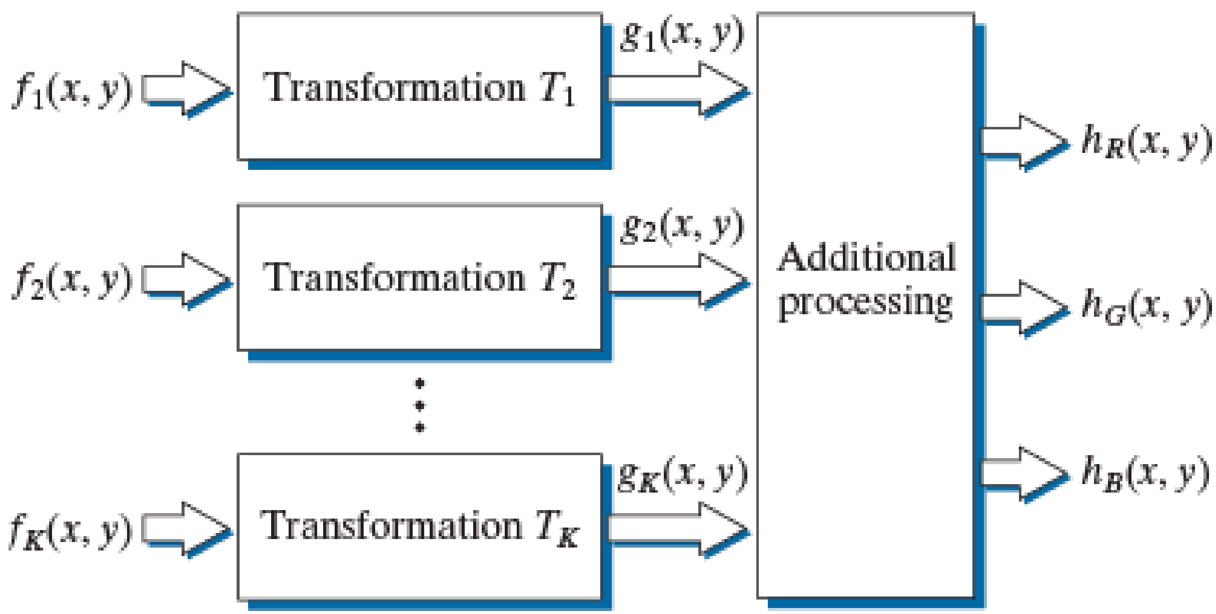
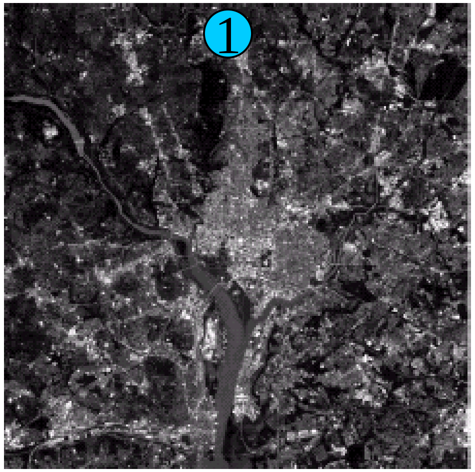


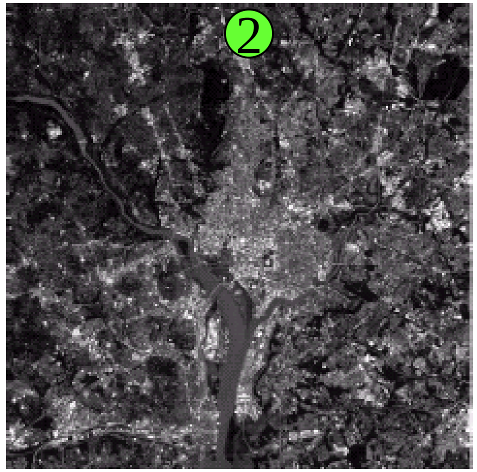
FIGURE 7.24 A pseudocolor coding approach using multiple grayscale images. The inputs are grayscale images. The outputs are the three components of an RGB composite image.

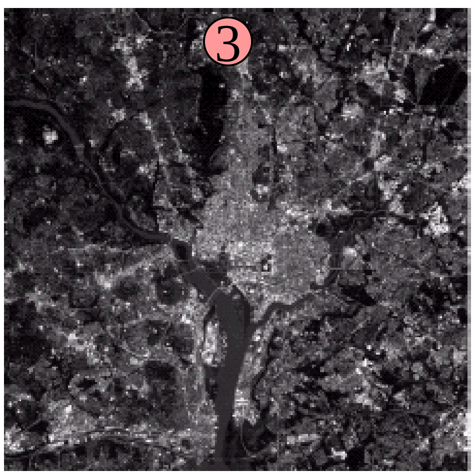
Pseudocolor Coding Example

Visible blue
λ= 0.45-0.52 μm
Max water penetration

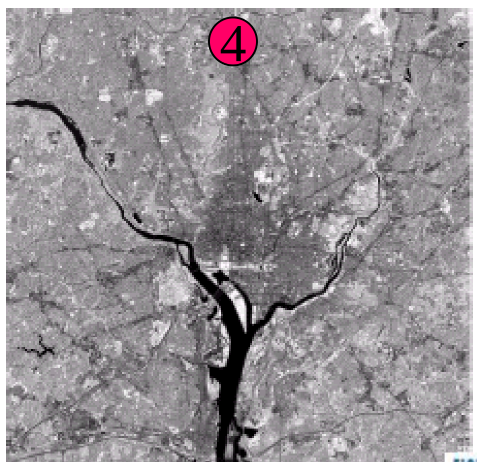


Visible green
λ= 0.52-0.60 μm
Measuring plant





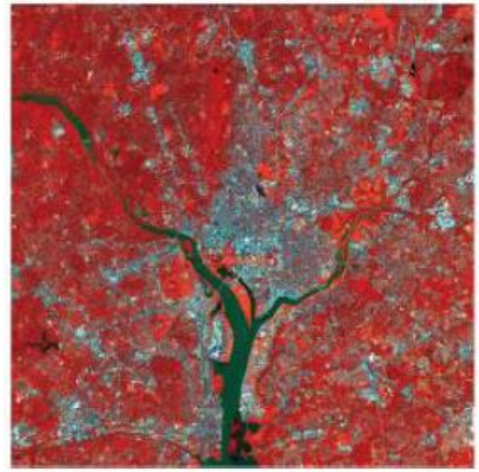
Visible red
λ= 0.63-0.69 μm
Plant discrimination



Near infrared
λ= 0.76-0.90 μm
Biomass and shoreline mapping

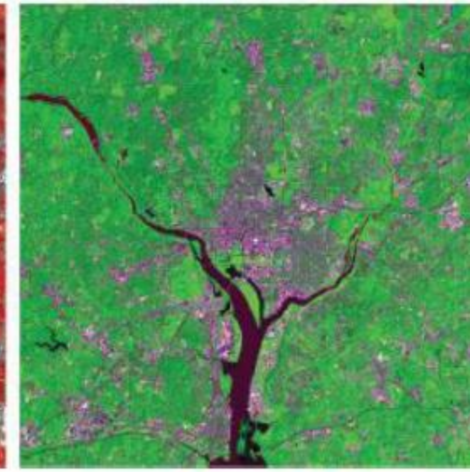
Washington D.C. area

Color composite images



Red = 4
Green = 2
Blue = 3

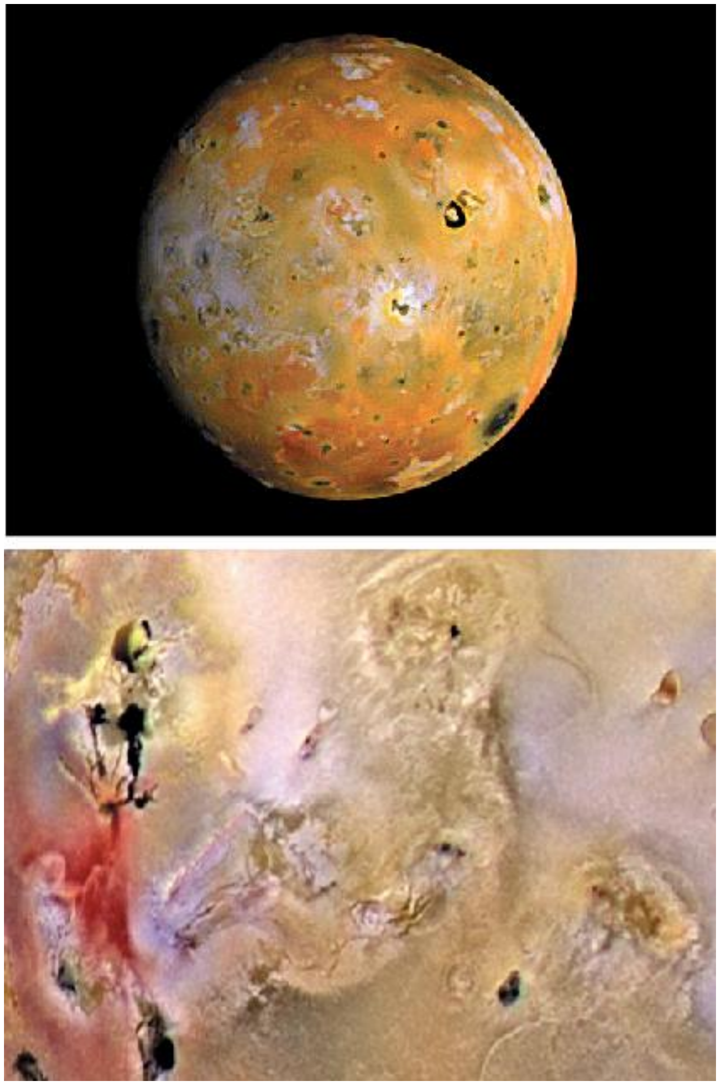
Shows difference between Biomass and man-made objects



Red = 3
Green = 4
Blue = 1

FIGURE 7.25 (a)–(d) Red (R), green (G), blue (B), and near-infrared (IR) components of a LANDSAT multispectral image of the Washington, D.C. area. (e) RGB color composite image obtained using the IR, G, and B component images. (f) RGB color composite image obtained using the R, IR, and B component images. (Original multispectral images courtesy of NASA.)

Pseudocolor Coding Example



Psuedocolor rendition of Jupiter moon Io

Yellow areas = Older sulfur deposits
Red areas = Material ejected from active volcanoes

A close-up

a
b

FIGURE 7.26 (a) Pseudocolor rendition of Jupiter Moon Io. (b) A close-up. (Courtesy of NASA.)

7.4 Basics of Full-Color Image Processing

Full color image processing: Manipulate real color images such as color photographs

Two Processing Methods:

1. **Per-color-component processing:** Process each component (R or G or B) image separately and then form composite
2. **Vector processing**: Treat each pixel as a vector (with RGB bands) to be processed

Full Color Image Processing

- Two processing methods:

  (1) Process each channel (or color component) separately, as if the color image were three gray scale images

  (2) Work with color pixels directly represented as a vector and process all 3 channels for each pixel

- In order for per-color-component and vector-based processing to be equivalent, two conditions have to be satisfied
  - The process has to be applicable to both vectors and scalars.
  - The operation on each component of a vector must be independent of the other components.

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

Basics of Full-Color Image Processing

- Example of per-color-component processing: Smoothing an image by smoothing each RGB component separately.
- Both approaches used in later sections

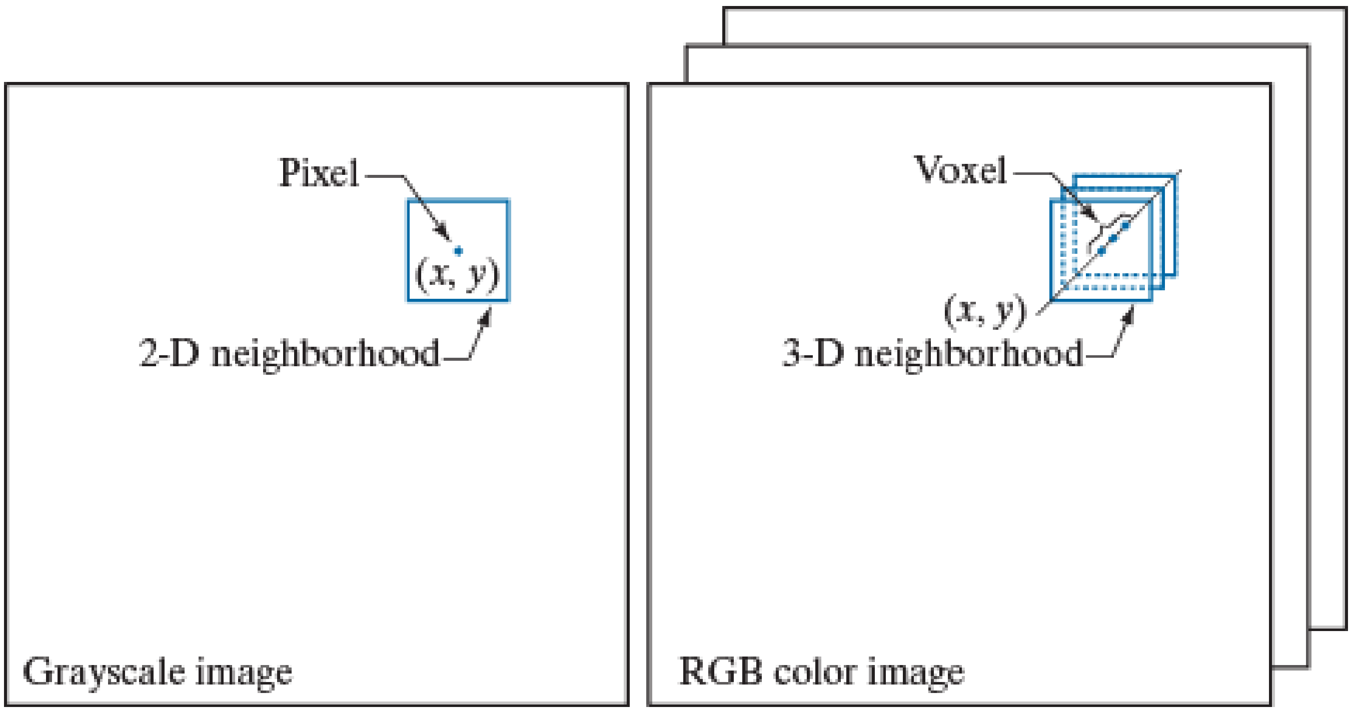


FIGURE 7.27 Spatial neighborhoods for grayscale and RGB color images. Observe in (b) that a *single* pair of spatial coordinates, (x, y), addresses the same spatial location in all three images.

Basics of Full-Color Image Processing

Let c represent an arbitrary vector in RGB color space:

$$c = \begin{bmatrix} c_R \\ c_G \\ c_B \end{bmatrix} = \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

At coordinates (x, y),

$$c(x,y) = \begin{bmatrix} c_R(x,y) \\ c_G(x,y) \\ c_B(x,y) \end{bmatrix} = \begin{bmatrix} R(x,y) \\ G(x,y) \\ B(x,y) \end{bmatrix}$$

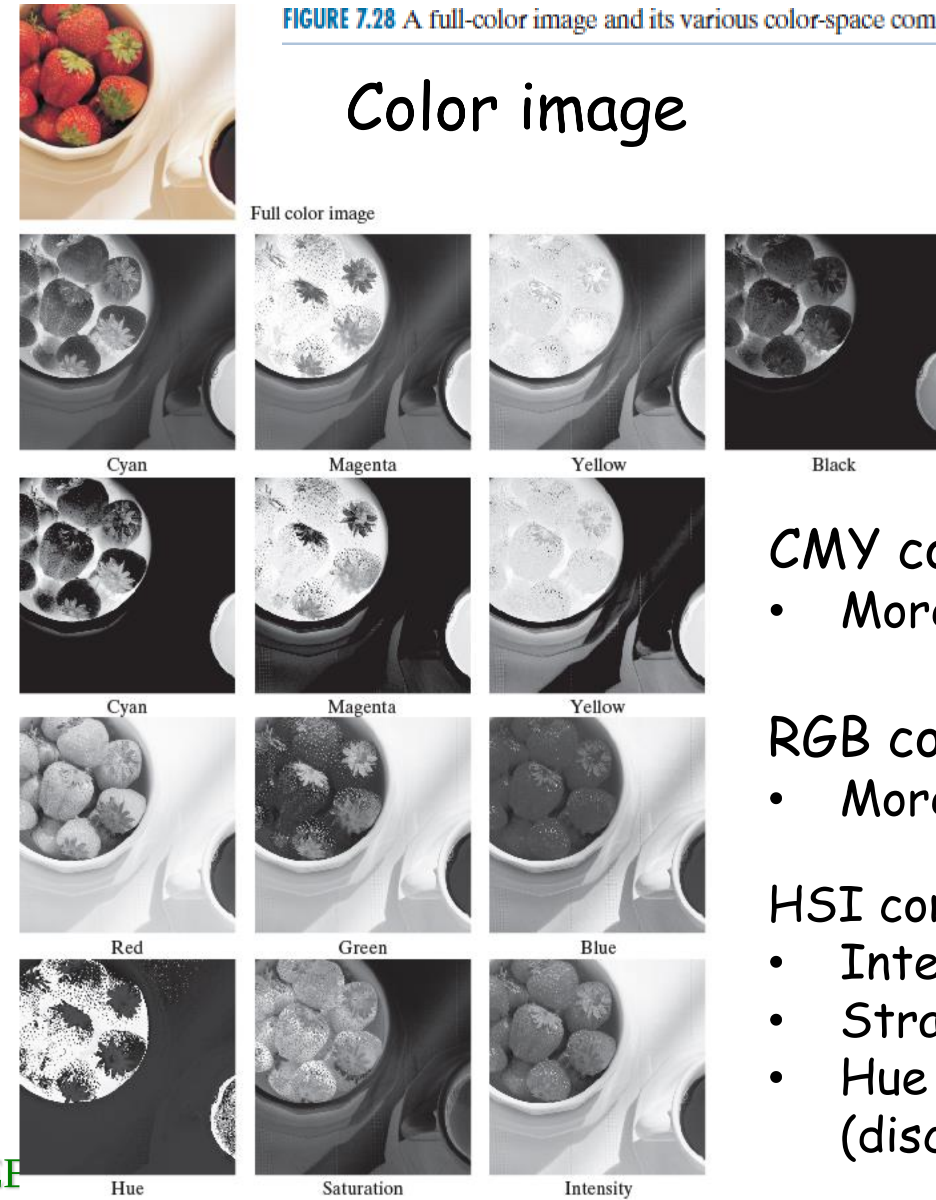
7.5 Color Transformations

- Process color images within a single color model
- No RGB-> HSI or HSI-> RGB conversion
- Gray scale image transformations may also be applied to each color separately.

FIGURE 7.28 A full-color image and its various color-space components. (Original image courtesy of MedData Interactive.)

Example: Full-Color Image and Interpretation of Color Space Components

FIGURE 7.28 A full-color image and its various color-space components. (Original image courtesy of MedData Interactive.)



Color image

- Black-> 0 & white-> 1

CMYK components

- More magenta & yellow mixing up to form red

CMY components

- More magenta & yellow mixing up to form red

RGB components

- More red, very little blue or green

HSI components

- Intensity: Monochrome of color image
- Strawberries are pure → i.e., saturated
- Hue is most difficult to interpret (discontinuity where 0 and 360 meet)

7.5 Color Transformations

Used to transform colors to colors.

Formulation: $g(x,y) = T\left[f(x,y)\right]$

$f(x,y)$ = Input color image
$g(x,y)$ = Output color image
T = Operation on f over a spatial neighborhood of (x,y)

When only data at one pixel is used in the transformation, we can express the transformation as:

$s_i = T_i(r_i)$ $\quad i = 1, 2, \ldots, n$ $\quad$ For RGB images, $n = 3$

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

Example: Color Transformation – Intensity Manipulation

Formula for RGB:

$$s_R(x,y) = kr_R(x,y)$$
$$s_G(x,y) = kr_G(x,y)$$
$$s_B(x,y) = kr_B(x,y)$$

Formula for HSI:

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

Formula for CMY:

$$s_C(x,y) = kr_C(x,y) + (1-k)$$
$$s_M(x,y) = kr_M(x,y) + (1-k)$$
$$s_Y(x,y) = kr_Y(x,y) + (1-k)$$

Formula for CMYK:

$$s_C(x,y) = r_C(x,y)$$
$$s_M(x,y) = r_M(x,y)$$
$$s_Y(x,y) = r_Y(x,y)$$
$$s_K(x,y) = kr_K(x,y) + (1-k)$$

3rd Edition

Intensity Reduction by 30% (k=0.7)

- The output is the same regardless of the color space

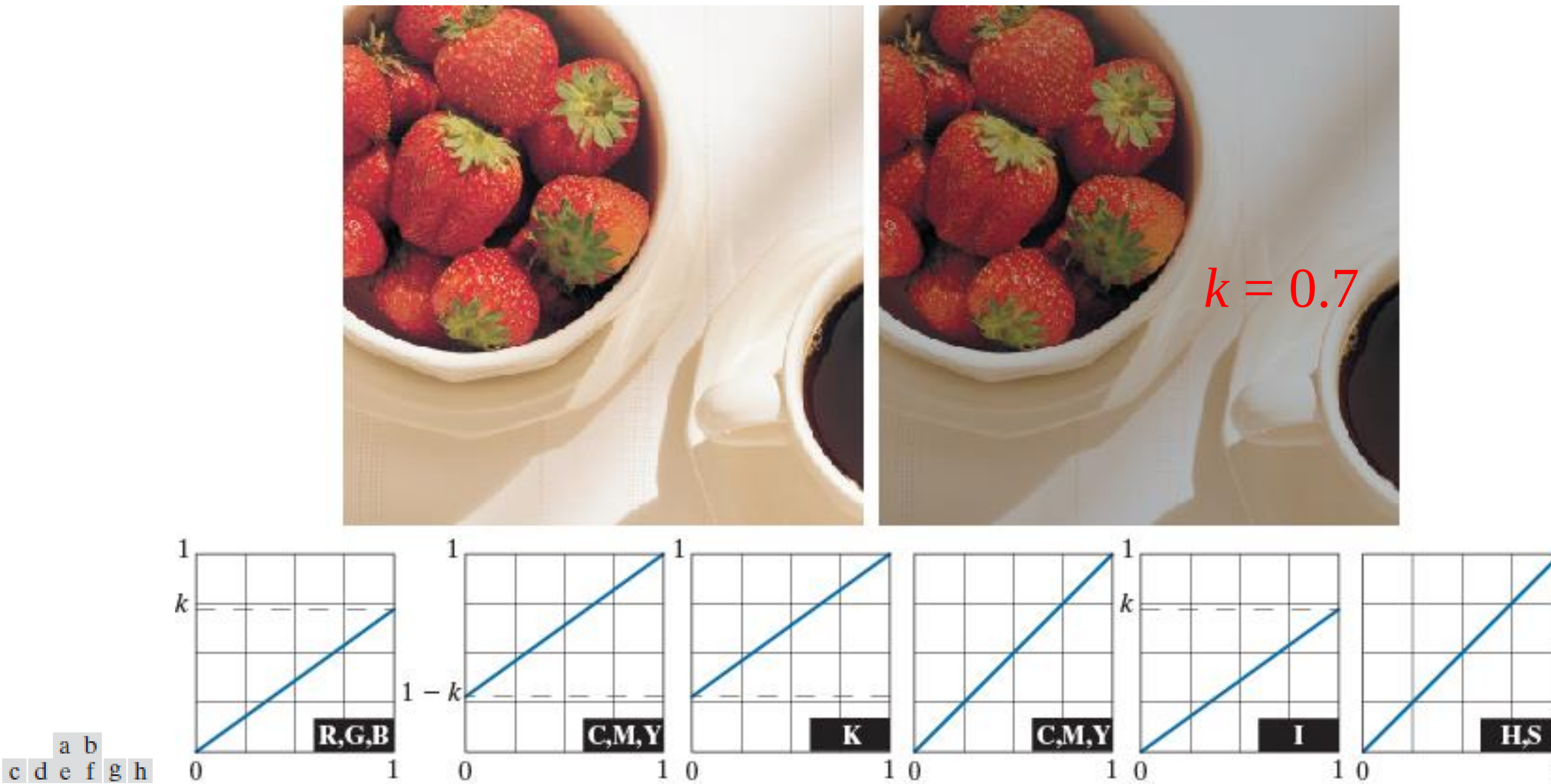


FIGURE 7.29 Adjusting the intensity of an image using color transformations. (a) Original image. (b) Result of decreasing its intensity by 30% (i.e., letting $k = 0.7$). (c) The required RGB mapping function. (d)–(e) The required CMYK mapping functions. (f) The required CMY mapping function. (g)–(h) The required HSI mapping functions. (Original image courtesy of MedData Interactive.)

Color Complements

- For enhancement of dark regions.
- Opposite hues in the color circle (e.g., reds replaced by cyan)
- Straightforward in RGB space
- No equivalent transformation in HSI space.
  - The saturation (S) component cannot be computed from the S component of the original image (because of the equations for S in slide-47)

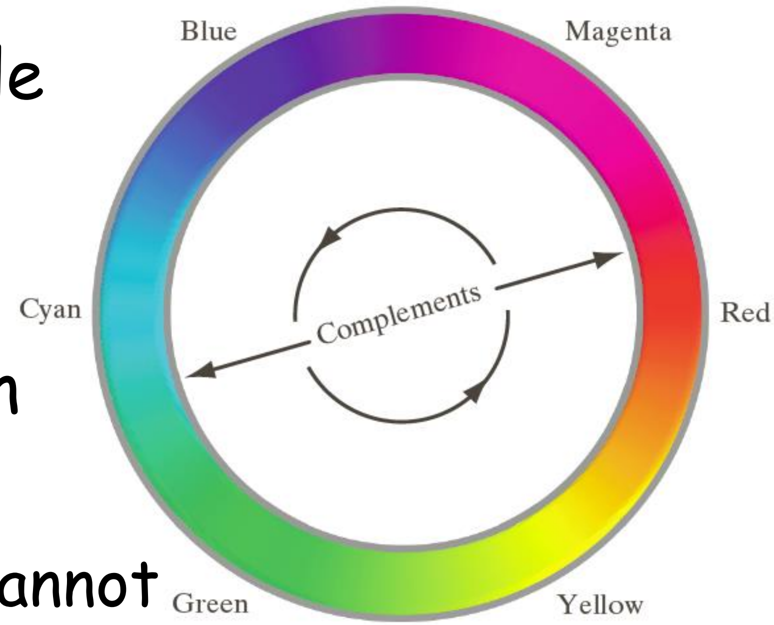
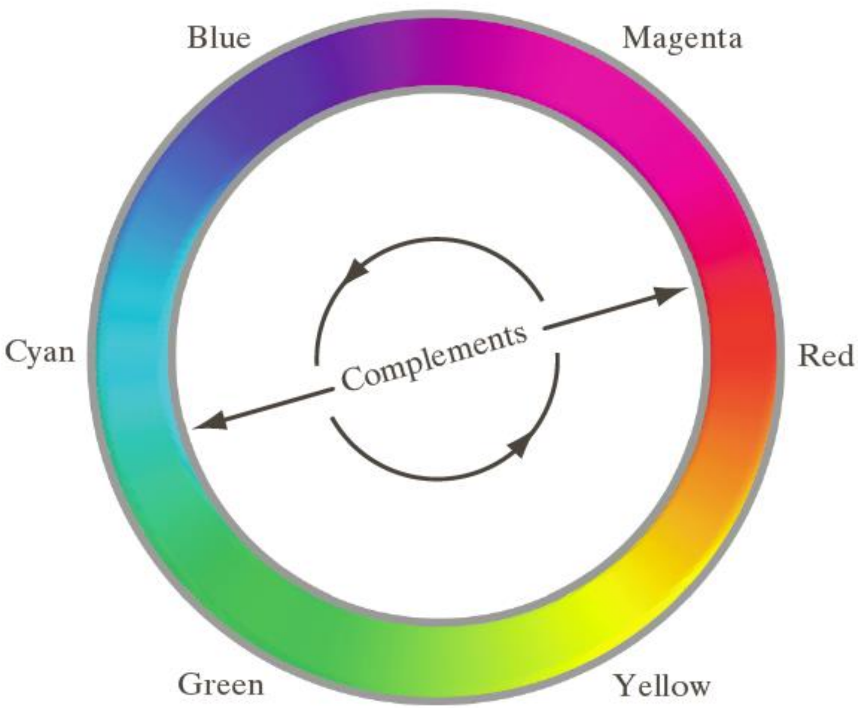


FIGURE 7.30 Color complements on the color circle.

Color Complements

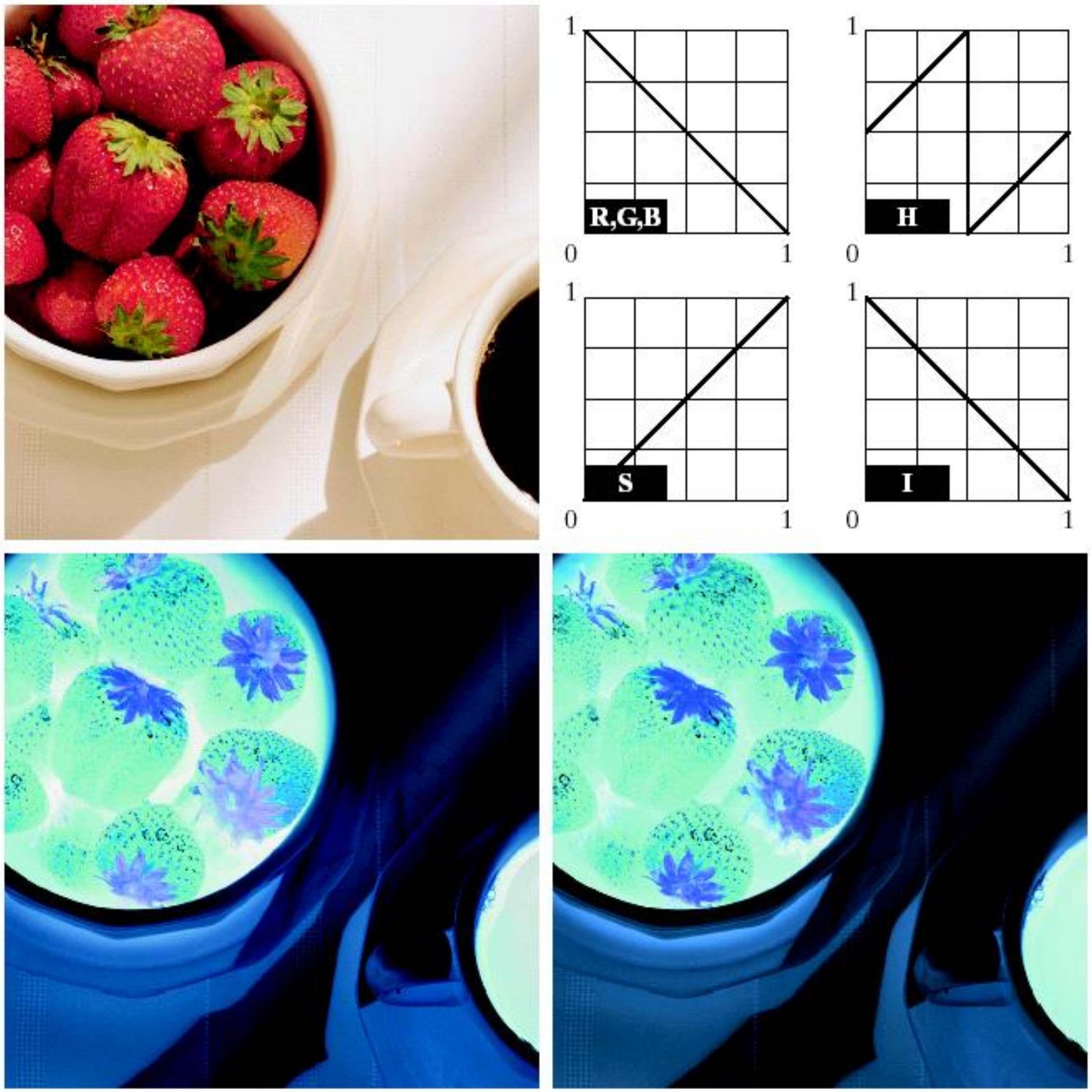
- Color complement replaces each color with its opposite color in the color circle of the Hue component
- This operation is analogous to image negative in a gray scale image.



Color circle

FIGURE 7.30 Color complements on the color circle.

Color Complement Transformation Example



a b
c d

FIGURE 7.31 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.

Color Slicing Transformation

- Useful for separating objects from surroundings
- If the color of a pixel is far from a desired color than threshold distance, then set that color to some specific color such as gray. Otherwise keep the original color unchanged. $(a_1, a_2, \cdots, a_n)$ → Average color components

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

→ Set to gray

→ Keep the original color

$$i = 1, 2, \ldots, n$$

Or, use a sphere

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

→ Set to gray

→ Keep the original color

$$i = 1, 2, \ldots, n$$

Color Slicing – Edible part

Red within RGB cube Red within RGB sphere

Original image

a b

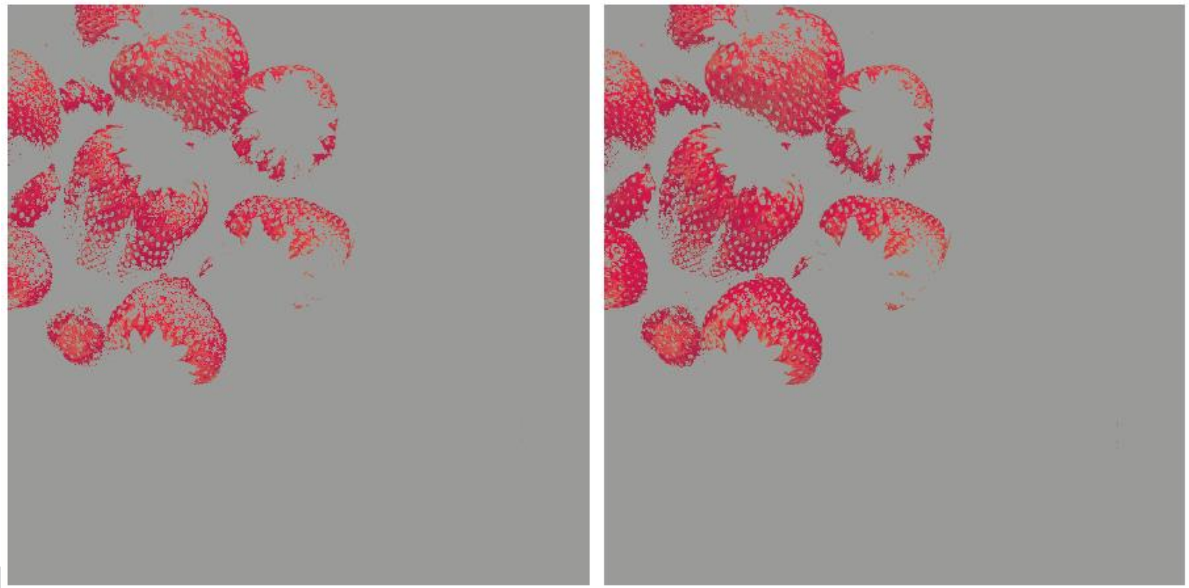


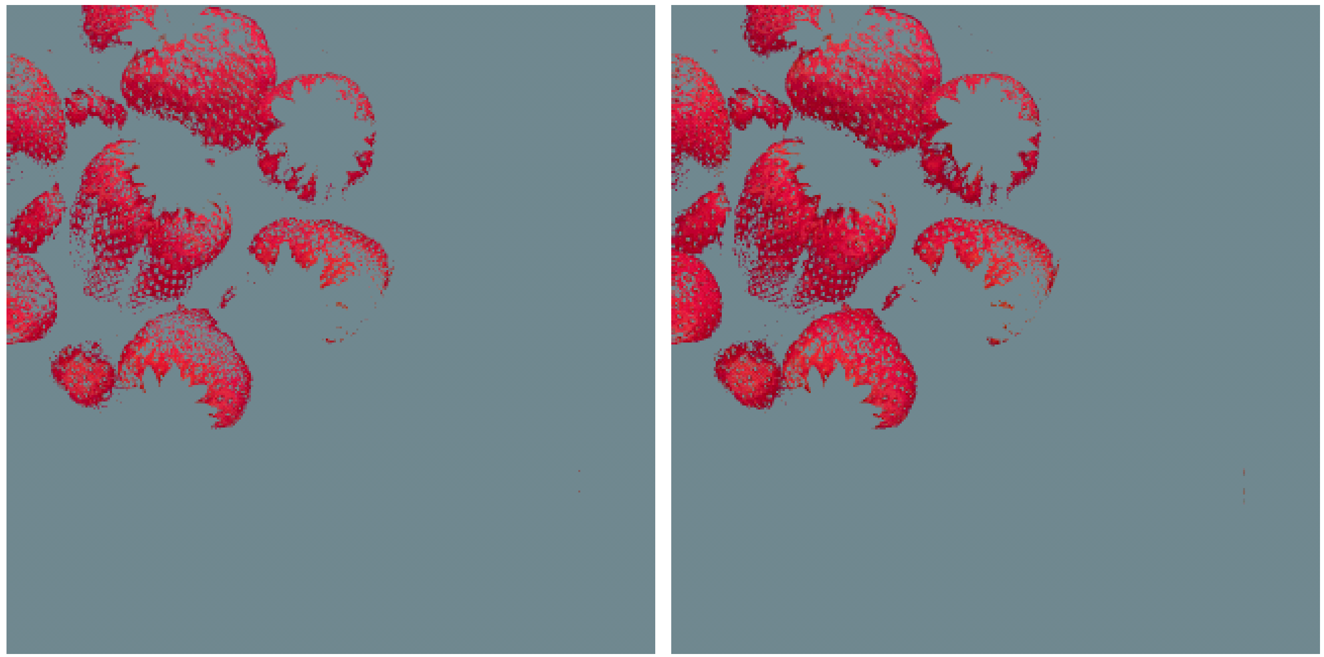
FIGURE 7.32 Color-slicing transformations that detect (a) reds within an RGB cube of width $W = 0.2549$ centered at $(0.6863, 0.1608, 0.1922)$, and (b) reds within an RGB sphere of radius 0.1765 centered at the same point. Pixels outside the cube and sphere were replaced by color $(0.5, 0.5, 0.5)$.

- **Color slicing** for separating objects from their surroundings.
- A first step towards image segmentation.

Color Slicing Transformation Example

After color slicing

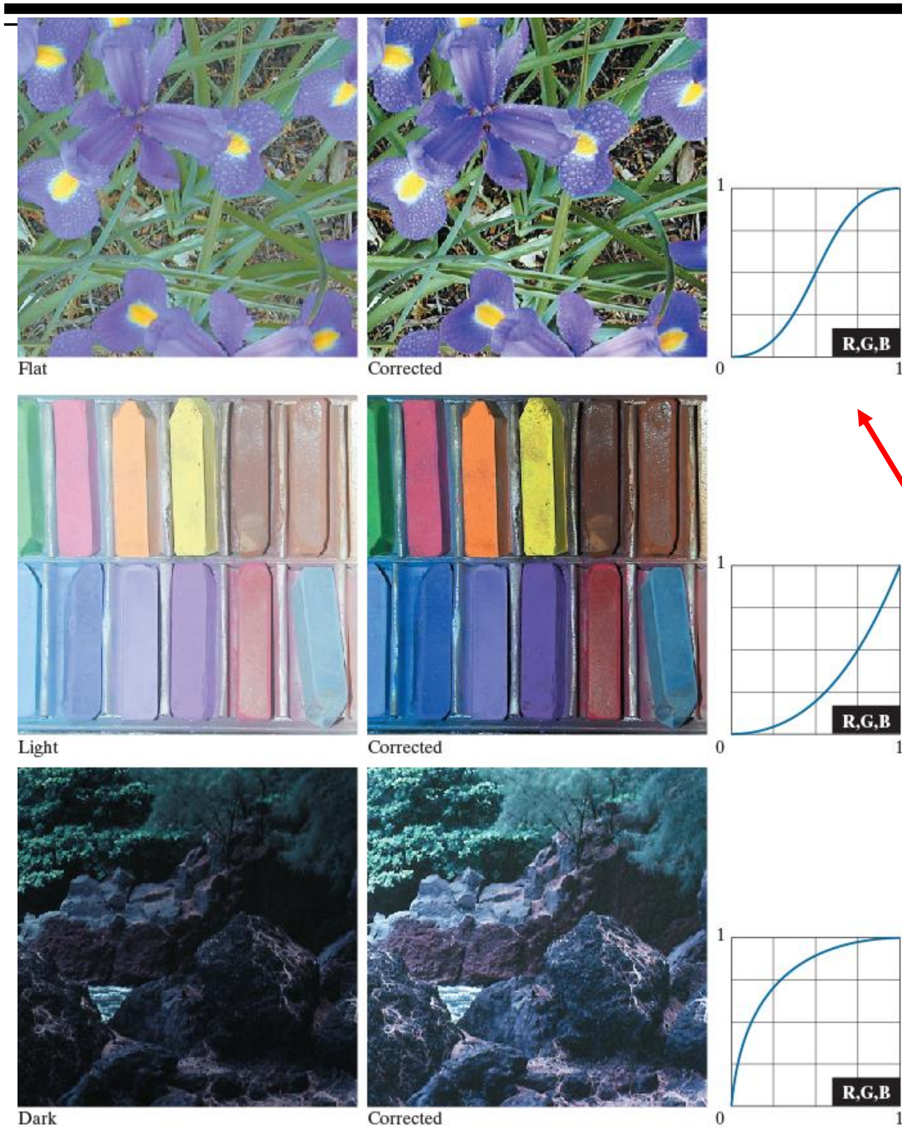
Original image



a b

FIGURE 7.32 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 $(0.5, 0.5, 0.5)$.

Tonal Correction Examples



- Only brightness and contrast are adjusted while keeping color unchanged.
- This can be done by using the same transformation for all RGB or CMYK components.
- For HSI: Only Intensity is changed

Contrast enhancement

Power law transformations

FIGURE 7.33 Tonal corrections for flat, light (high key), and dark (low key) color images. Adjusting the red, green, and blue components equally does not always alter the image hues significantly.

Histogram Processing of Color Images

- The approaches used so far are interactive.
- Histogram equalization of a color image can be performed automatically by adjusting color intensity uniformly while leaving the colors (i.e., hues) unchanged
- 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 images

Histogram Equalization

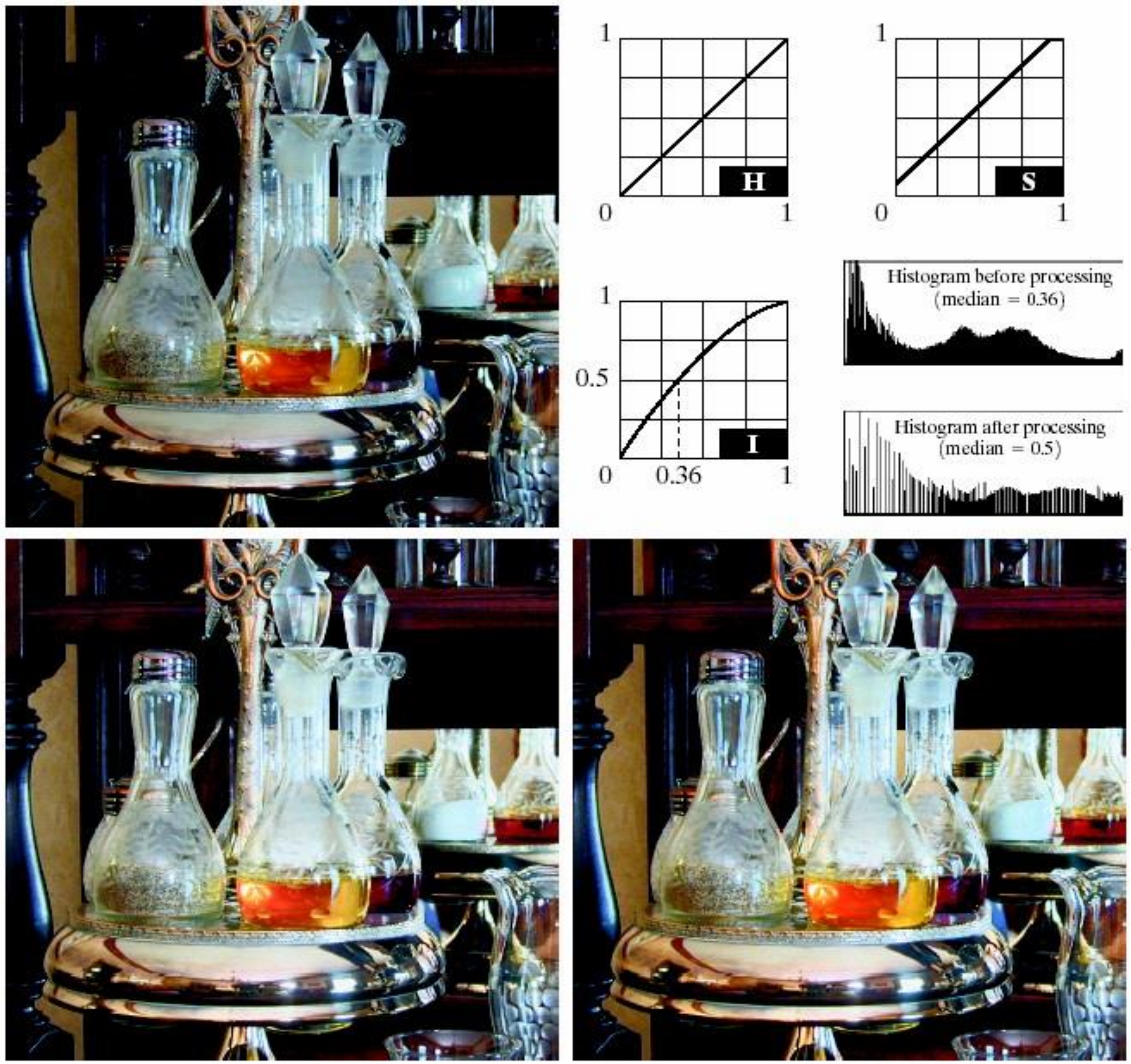


FIGURE 7.35 Histogram equalization (followed by saturation adjustment) in the HSI color space.

Histogram Equalization of Intensity without altering hue/saturation

Increased Saturation after Histogram Equalization → More vibrant oil and vinegar colors

7.6 Color Image Smoothing

Two Methods:

1. **Per-color-plane method:** For RGB and CMY color models
   Smooth each color plane using moving average and then combine back to RGB

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

2. **Smooth only Intensity (I) component (faster)** of an HSI image while leaving H and S unmodified

Note: These two methods are not equivalent

Color Image Smoothing Example (cont.)

Color image

Red

Green

Blue

a b
c d

FIGURE 7.36
(a) RGB image.
(b) Red component image.
(c)Green component.
(d) Blue component.

Color Image Smoothing Example (cont.)

Color image

a b c

FIGURE 7.37 HSI components of the RGB color image in Fig. 7.36(a). (a) Hue. (b) Saturation. (c) Intensity.

HSI Components

Color Image Smoothing Example (cont.)

Smooth all RGB components

Smooth only I component of HSI (faster)

a b c

FIGURE 7.38 Image smoothing with a 5×5 averaging kernel. (a) Result of processing each RGB component image. (b) Result of processing the intensity component of the HSI image and converting to RGB. (c) Difference between the two results.

Color Image Smoothing Example (cont.)

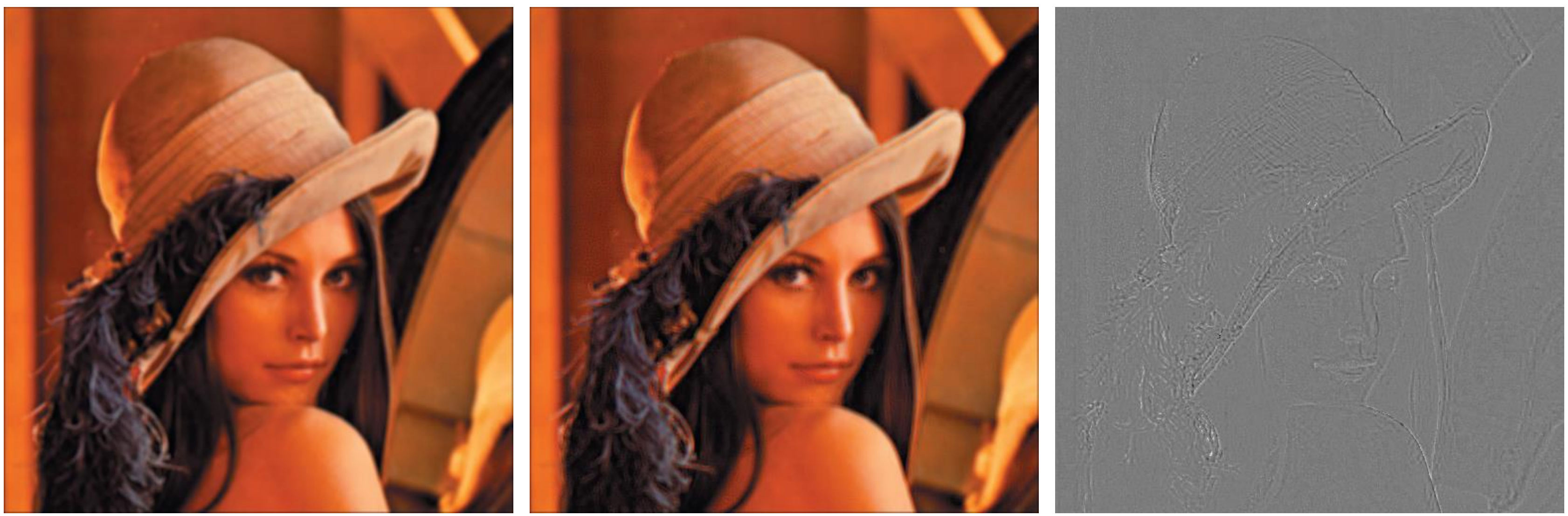
Difference between smoothed results from two methods in the previous slide.

a b c

FIGURE 7.38 Image smoothing with a 5×5 averaging kernel. (a) Result of processing each RGB component image. (b) Result of processing the intensity component of the HSI image and converting to RGB. (c) Difference between the two results.

Color Image Smoothing (cont…)

- HSI decouples intensity from color. Suitable for processing only the intensity component of an image.



a b c

FIGURE 7.38 Image smoothing with a 5×5 averaging kernel. (a) Result of processing each RGB component image. (b) Result of processing the intensity component of the HSI image and converting to RGB. (c) Difference between the two results.

Color Image Sharpening

Can be done in the same manner as color image smoothing:

1. Per-color-plane method for RGB & CMY images
2. Sharpening only I component of a HSI image

Sharpening all RGB components   Sharpening only I component of HSI

a b c

FIGURE 7.39 Image sharpening using the Laplacian. (a) Result of processing each RGB channel. (b) Result of processing the HSI intensity component and converting to RGB. (c) Difference between the two results.

Color Image Sharpening Example (cont.)

a b c

FIGURE 7.39 Image sharpening using the Laplacian. (a) Result of processing each RGB channel. (b) Result of processing the HSI intensity component and converting to RGB. (c) Difference between the two results.

Difference between sharpened results from two methods in the previous slide.

Color Image Sharpening

- The difference becomes more pronounced by increasing the filter size.

a b c

FIGURE 7.39 Image sharpening using the Laplacian. (a) Result of processing each RGB channel. (b) Result of processing the HSI intensity component and converting to RGB. (c) Difference between the two results.

7.8 Noise in Color Images

- The noise models discussed for grayscale images are also applicable to color images.
- However, in many applications, a color channel may be more (or less) affected than the other channels.
- For instance, using a red color filter in a CCD camera may affect the red component of the image (CCD sensors are noisier at low levels of illumination).
- We will take a brief look of how noise carries over when converting from one color model to another.

Noise in Color Images

Noise can corrupt each color component independently.

a b
c d

FIGURE 7.46 (a)–(c) Red, green, and blue 8-bit component images corrupted by additive Gaussian noise of mean 0 and standard deviation of 28 intensity levels. (d) Resulting RGB image. [Compare (d) with Fig. 7.44(a).]

FIGURE 7.44 (a) RGB image.

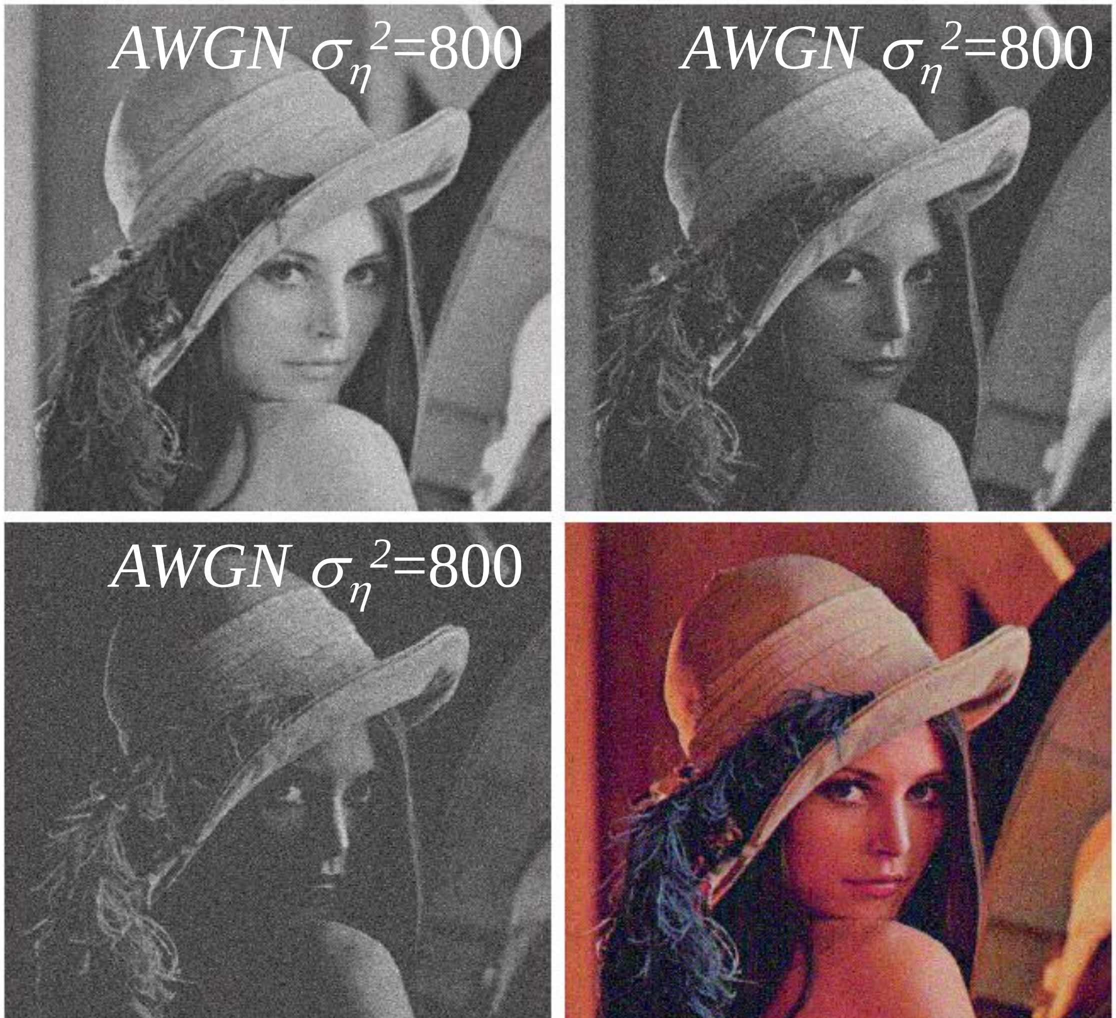
Noise is less noticeable in a color image

Noise in Color Images

Noise can corrupt each color component independently.

a b
c d

FIGURE 6.48 (a)–(c) Red, green, and blue component images corrupted by additive Gaussian noise of mean 0 and variance 800. (d) Resulting RGB image. [Compare (d) with Fig. 6.46(a).]



Noise is less noticeable in a color image

3rd Edition

Noise in Color Images

FIGURE 7.47 HSI components of the noisy color image in Fig. 7.46(d). (a) Hue. (b) Saturation. (c) Intensity.

- The hue and saturation components are significantly degraded. This is due to the nonlinearity of the cos and min operations used in the transformation.
- The intensity component is smoother due to averaging of the three noisy RGB components.

WRIGHT STATE
UNIVERSITY

Noise in Color Images

a b
c d

FIGURE 7.48
(a) RGB image with green plane corrupted by salt-and-pepper noise. (b) Hue component of HSI image. (c) Saturation component. (d) Intensity component.

Noise in Color Images (cont…)

- When only one channel is affected by noise, conversion to HSI spreads the noise to all HSI components images
- This is due to the transformation that makes use of all RGB components to compute each HSI components

a b
c d

FIGURE 7.48 (a) RGB image with green plane corrupted by salt-and-pepper noise. (b) Hue component of HSI image. (c) Saturation component. (d) Intensity component.

Greyscale from RGB

- Sometimes a single value at each pixel makes processing easier
- This value is usually the intensity or 'grey value'
- We can convert an RGB image to greyscale using
  - Simple average of red, green and blue, i.e.

    $i = (r+g+b)/3$ or
  - Weighted average, i.e.

    $i = a_1r+a_2g+a_3b$ where $a_1+a_2+a_3=1$

  where, i is the grey value and r, g, and b are the red green and blue values

Average Original Weighted

- Since our eyes are more sensitive to green light one commonly used formula:

i = 0.30r + 0.59g + 0.11b

Acknowlegements

The slides are primarily based on the figures and images in the Digital Image Processing textbook by Gonzalez and Woods:

- http://www.imageprocessingplace.com/DIP-3E/dip3e_book_images_downloads.htm

In addition, slides have been adopted and modified from the following sources:

  - http://www.cs.uoi.gr/~cnikou/Courses/Digital_Image_Processing
  - http://www.comp.dit.ie/bmacnamee/gaip.htm
  - http://baggins.nottingham.edu.my/~hsooihock/G52IIP/
  - http://gear.kku.ac.th/~nawapak/178353.html
  - https://cs.nmt.edu/~ip/index.html