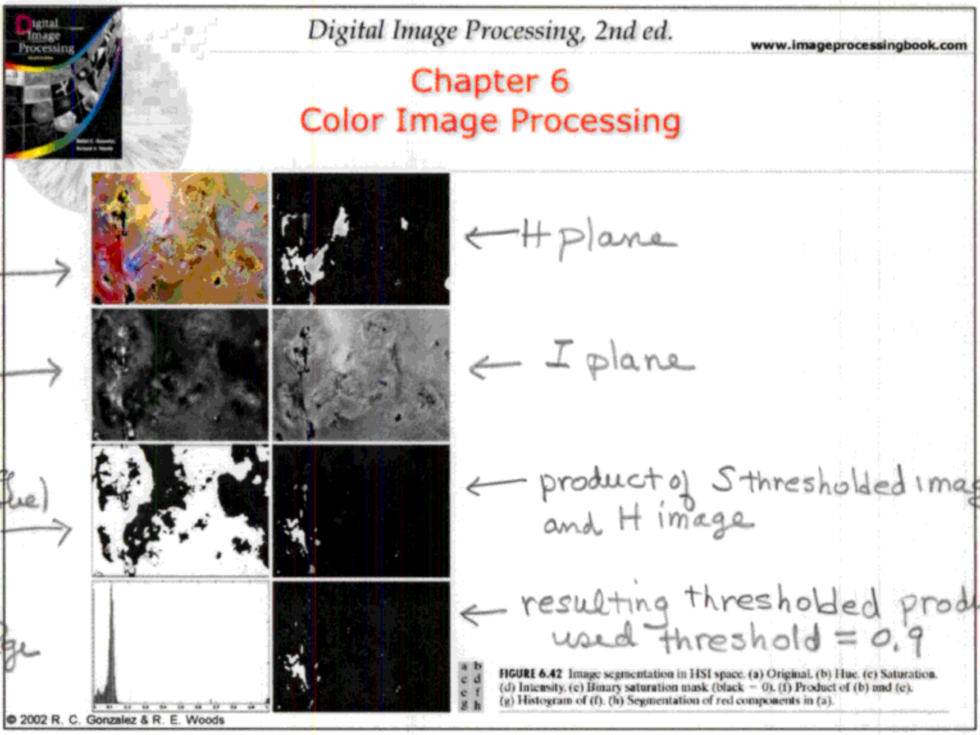


Digital Image Processing, 2nd ed.

www.imageprocessingbook.com

Chapter 6
Color Image Processing

want to segment this area

← H plane

S plane

← I plane

thresholding (0.1 max value) S plane

← product of S thresholded image and H image

histogram of product image

← resulting thresholded product image.
used threshold = 0.9

FIGURE 6.42 Image segmentation in HSI space. (a) Original. (b) Hue. (c) Saturation. (d) Intensity. (e) Binary saturation mask (black = 0). (f) Product of (b) and (e). (g) Histogram of (f). (h) Segmentation of red components in (a).

© 2002 R. C. Gonzalez & R. E. Woods

Segmentation (Chap. 10 topic)

If we want to segment an image based on color the hue (H) image is the most natural to use.

1st thresholding — identify saturated colors 1 - white saturated
0 - blk - unsaturated

products - regions of significant color

color distribution
histogram of regions of significant color

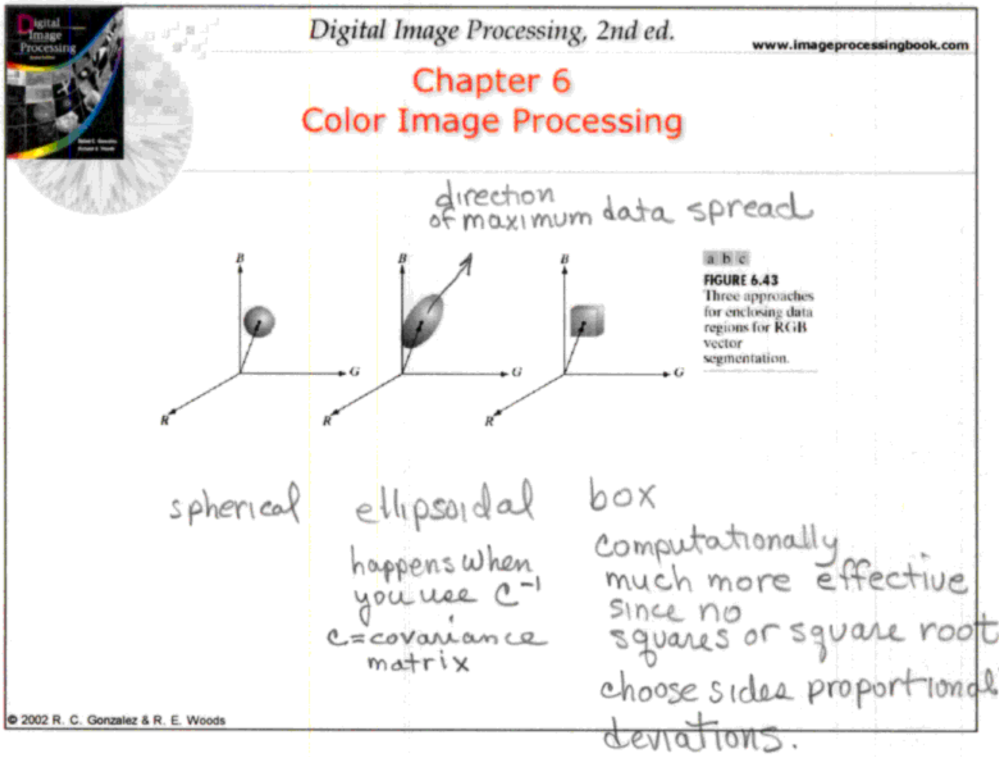
2nd thresholding will identify colors.
Colors > 0.9 are colors of interest.

colors near red

Digital Image Processing, 2nd ed.

www.imageprocessingbook.com

Chapter 6
Color Image Processing



a b c

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

© 2002 R. C. Gonzalez & R. E. Woods

direction of maximum data spread

spherical

ellipsoidal

happens when you use C^{-1}

C=covariance matrix

box

computationally much more effective since no squares or square roots

choose sides proportional to standard deviations.

Segmentation divides an image into constituent regions or objects.

Segmentation usually works better in RGB space than in HSI space.

Objective - classify each color pixel as having a color in the specified range or not

Use Euclidean distance

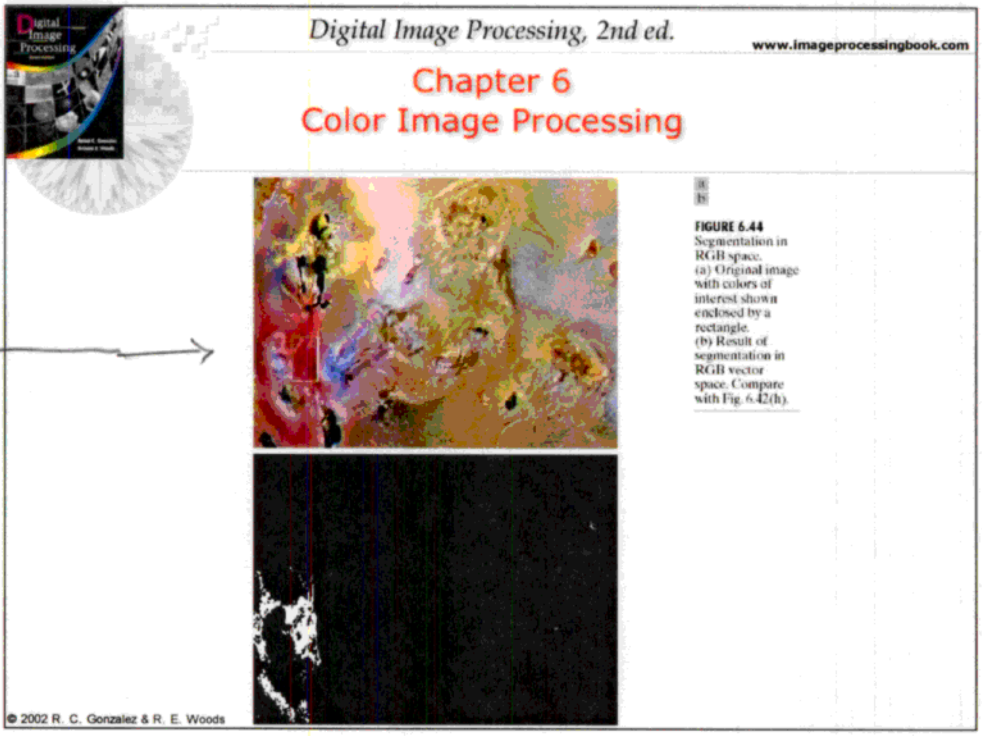
$$D(\underline{z}, \underline{a}) = \|\underline{z} - \underline{a}\| = \sqrt{(\underline{z} - \underline{a})^T (\underline{z} - \underline{a})}$$

$$D(\underline{z}, \underline{a}) = \sqrt{(z_R - a_R)^2 + (z_G - a_G)^2 + (z_B - a_B)^2}$$

A generalized distance measure is

$$D(\underline{z}, \underline{a}) = \sqrt{(\underline{z} - \underline{a})^T \underset{\uparrow \text{covariance matrix}}{\underline{C}^{-1}} (\underline{z} - \underline{a})}$$

$\underline{a}$ = average color vector for what
we are interested in
i.e. cluster cen



Digital Image Processing, 2nd ed.

www.imageprocessingbook.com

Chapter 6
Color Image Processing

FIGURE 6.44 Segmentation in RGB space. (a) Original image with colors of interest shown enclosed by a rectangle. (b) Result of segmentation in RGB vector space. Compare with Fig. 6.42(h).

rectangle shows color we want to segment →

Same image and goal as figure 6.42 but done in RGB space.

Procedure

1. Compute mean and standard deviations of the color contained in the sample rectangle.

   mean $\underline{a}$

   std deviations $\sigma_R, \sigma_G, \sigma_B$

2. using box in color space

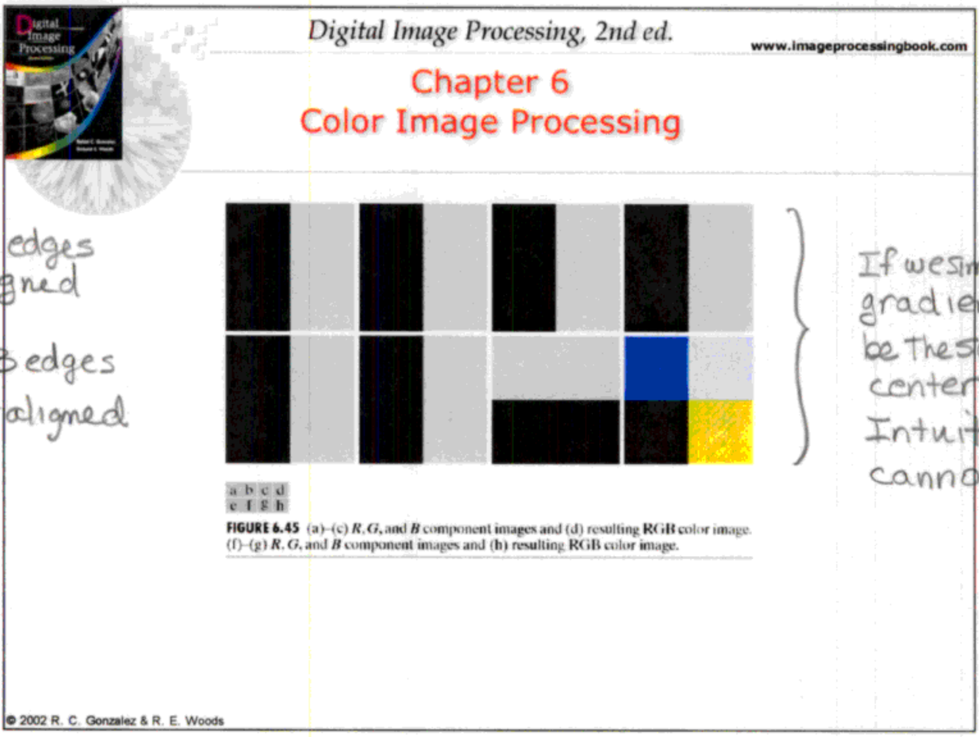
   $a_R \pm 1.25\sigma_R$

   $a_G \pm 1.25\sigma_G$

   $a_B \pm 1.25\sigma_B$

   classify pixel as white if inside this box, or white outside

Compare results with 6.42(h)



Digital Image Processing, 2nd ed.

www.imageprocessingbook.com

Chapter 6
Color Image Processing

FIGURE 6.45 (a)–(c) *R*, *G*, and *B* component images and (d) resulting RGB color image. (f)–(g) *R*, *G*, and *B* component images and (h) resulting RGB color image.

RGB edges aligned

RGB edges not aligned

If we simply add the gradients they would be the same at the center point. Intuitively they cannot be the same.

Computing edges in individual color planes vs. computing directly in color space.

Alternative vector definition

S. Di Zenzo (1986)
A Note on the Gradient of a Multi-Image
Computer Vision, Graphics and Image Processing
Vol. 33, pp. 116-125

We want to define the gradient (magnitude and direction) of the vector $\underline{c}(x,y)$

For a scalar function $f(x,y)$ the gradient is a vector pointing in the direction of maximum rate of change of f at (x,y)

$\hat{r}, \hat{g}, \hat{b}$ be unit vectors along the R, G, B axis of a RGB color space

define

$$\left.\begin{aligned} \underline{u} &= \frac{\partial R}{\partial x}\hat{r} + \frac{\partial G}{\partial x}\hat{g} + \frac{\partial B}{\partial x}\hat{b} \\ \underline{v} &= \frac{\partial R}{\partial y}\hat{r} + \frac{\partial G}{\partial y}\hat{g} + \frac{\partial B}{\partial y}\hat{b} \end{aligned}\right\}$$

These can be computed using a sobel operator.

further define

Implemented by GWE as colorgradient.

$$g_{xx} = \underline{u}\cdot\underline{u} = \underline{u}^T\underline{u} = \left|\frac{\partial R}{\partial x}\right|^2 + \left|\frac{\partial G}{\partial x}\right|^2 + \left|\frac{\partial B}{\partial x}\right|^2$$

$$g_{yy} = \underline{v}\cdot\underline{v} = \underline{v}^T\underline{v} = \left|\frac{\partial R}{\partial y}\right|^2 + \left|\frac{\partial G}{\partial y}\right|^2 + \left|\frac{\partial B}{\partial y}\right|^2$$

$$g_{xy} = \underline{u}\cdot\underline{v} = \underline{u}^T\underline{v} = \frac{\partial R}{\partial x}\frac{\partial R}{\partial y} + \frac{\partial G}{\partial x}\frac{\partial G}{\partial y} + \frac{\partial B}{\partial x}\frac{\partial B}{\partial y}$$

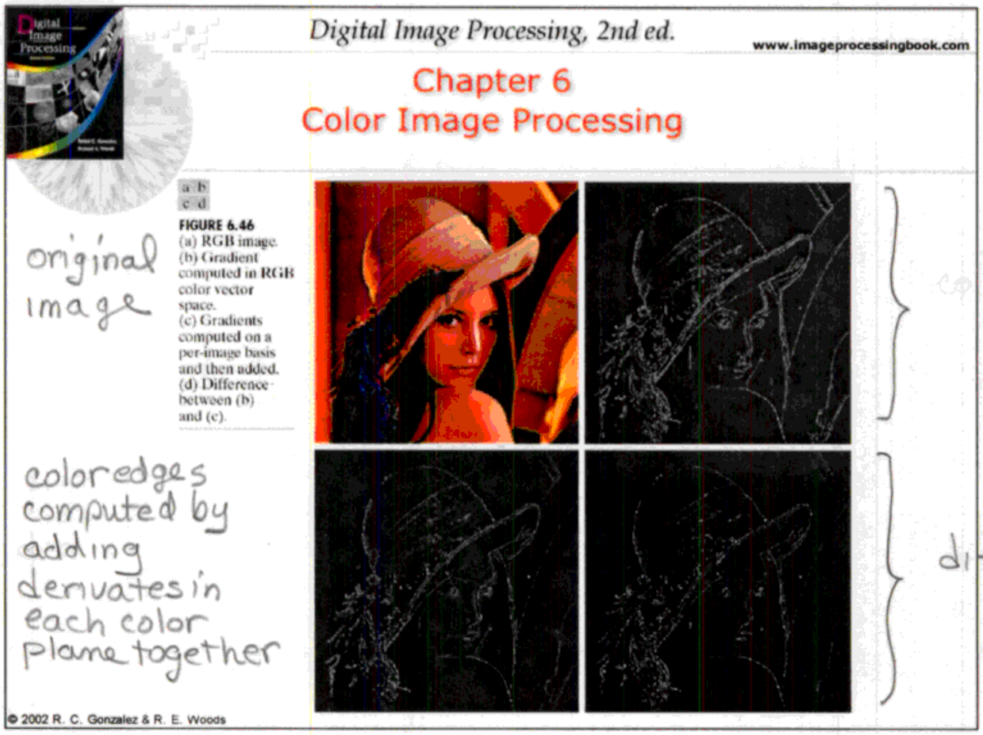
The maximum rate of change of $\underline{c}(x,y)$ at (x,y) is in the direction given by

$$\theta(x,y) = \frac{1}{2}\,\mathrm{Tan}^{-1}\left[\frac{2g_{xy}}{g_{xx}-g_{yy}}\right]$$

and the rate of change in the direction $\theta(x,y)$ is given by

$$F(\theta) = \sqrt{\frac{1}{2}(g_{xx}+g_{yy}) + (g_{xx}-g_{yy})\cos 2\theta + 2g_{xy}\sin 2\theta}$$

θ is given in two orthogonal directions. One is a maximum for F and the other is a minimum.



Digital Image Processing, 2nd ed.

www.imageprocessingbook.com

Chapter 6
Color Image Processing

a b
c d

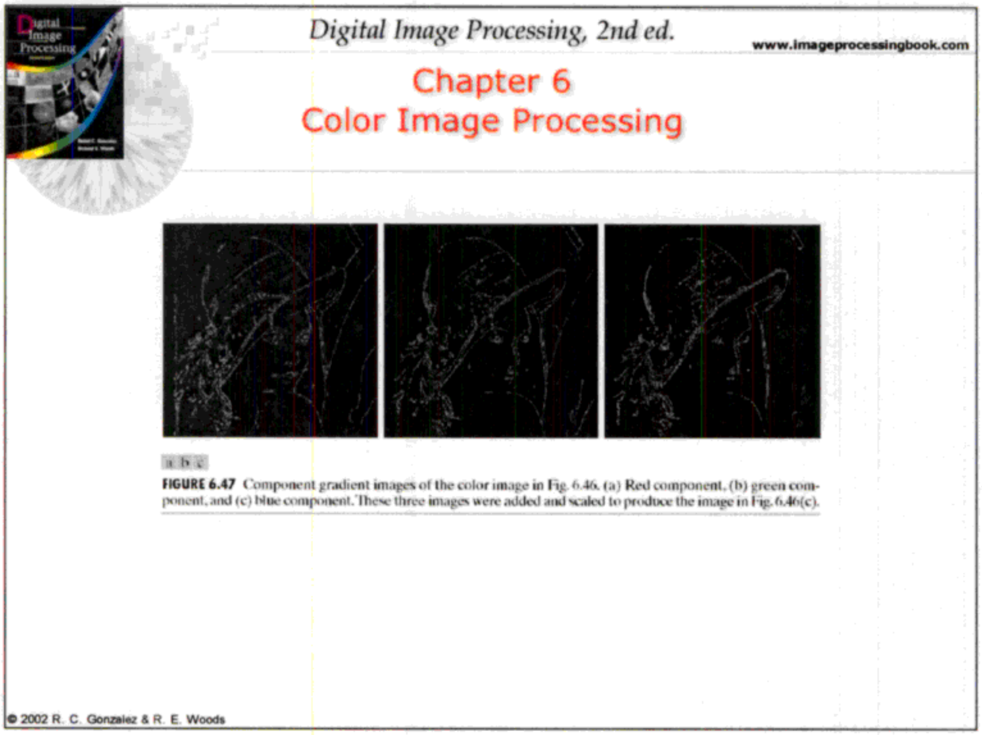
FIGURE 6.46 (a) RGB image. (b) Gradient computed in RGB color vector space. (c) Gradients computed on a per-image basis and then added. (d) Difference between (b) and (c).

original image

color edges computed used vector method in RGB color space

color edges computed by adding derivates in each color plane together

difference between (b) and (c)



Digital Image Processing, 2nd ed.

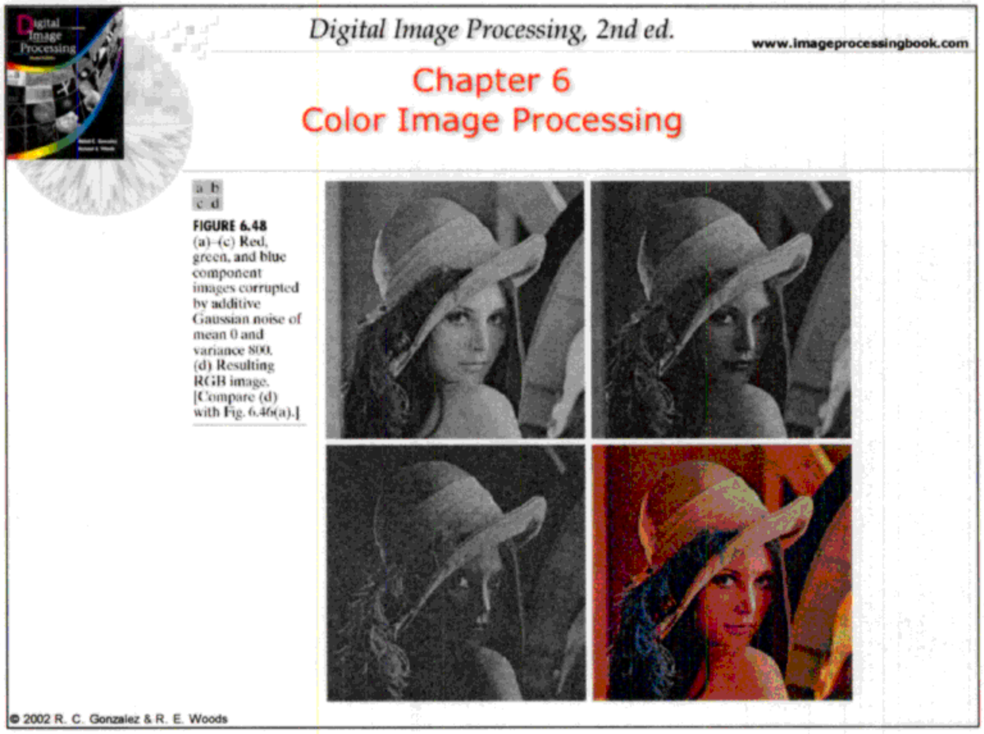
www.imageprocessingbook.com

Chapter 6
Color Image Processing

a b c

FIGURE 6.47 Component gradient images of the color image in Fig. 6.46. (a) Red component, (b) green component, and (c) blue component. These three images were added and scaled to produce the image in Fig. 6.46(c).

Individual RGB gradient images
Added and scaled to produce RGB gradient image.



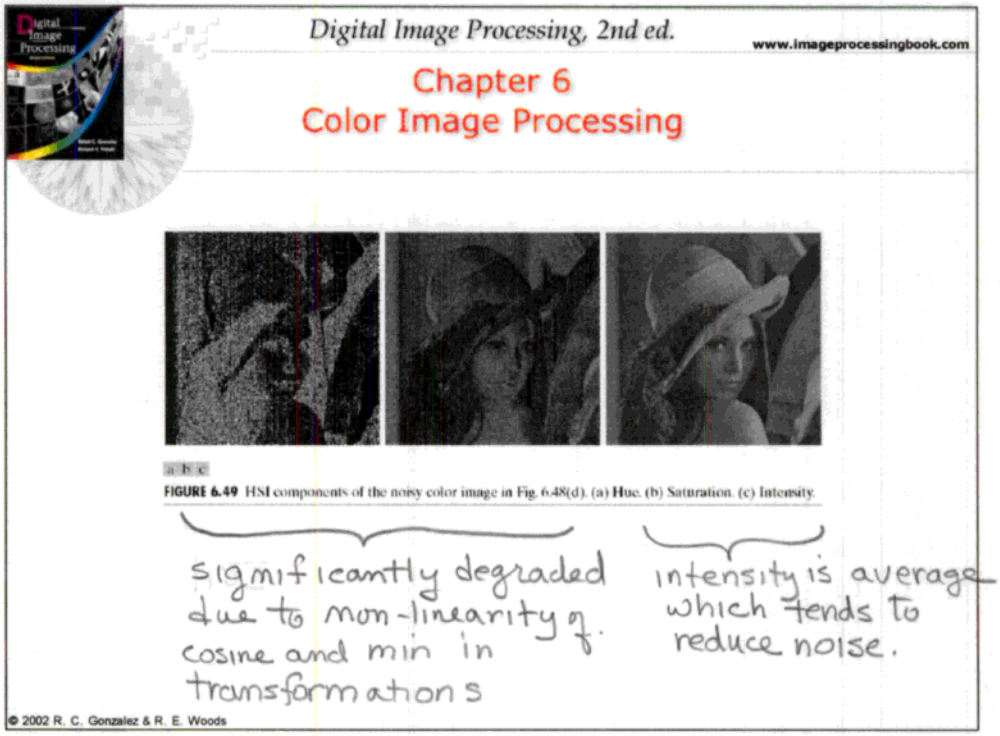
Digital Image Processing, 2nd ed.

www.imageprocessingbook.com

Chapter 6
Color Image Processing

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).]

Added Gaussian noise to the R, G, and B color planes.
RGB image with noise shown in (d).



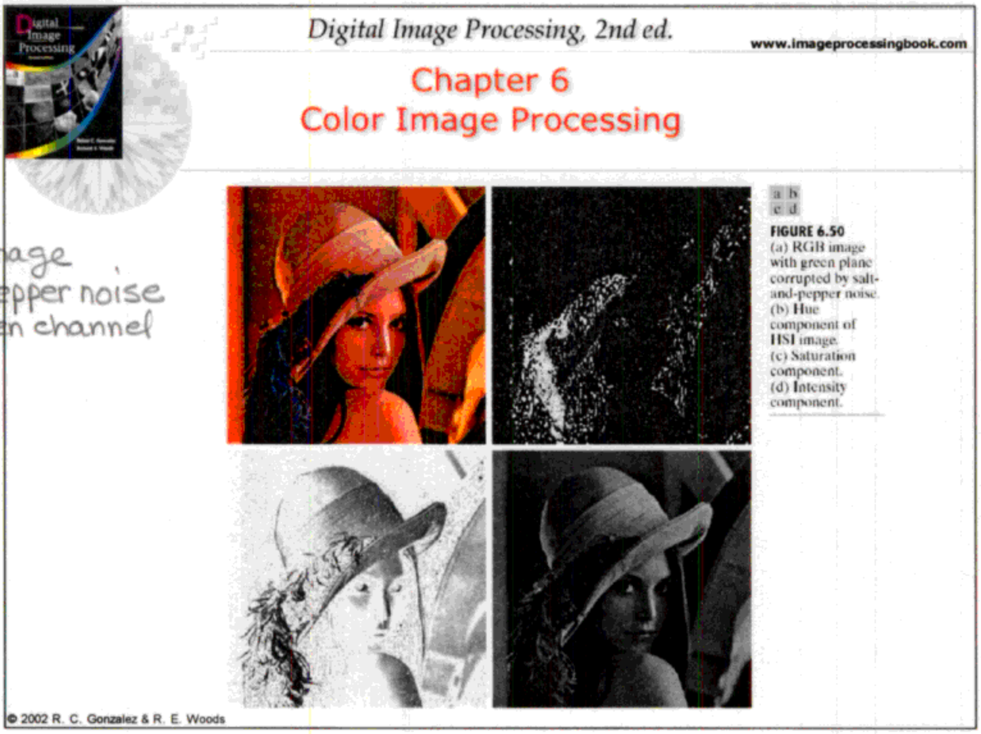
Chapter 6
Color Image Processing

FIGURE 6.49 HSI components of the noisy color image in Fig. 6.48(d). (a) Hue. (b) Saturation. (c) Intensity.

significantly degraded due to non-linearity of cosine and min in transformations

intensity is average which tends to reduce noise.

HSI components of the noisy RGB image.



Digital Image Processing, 2nd ed.

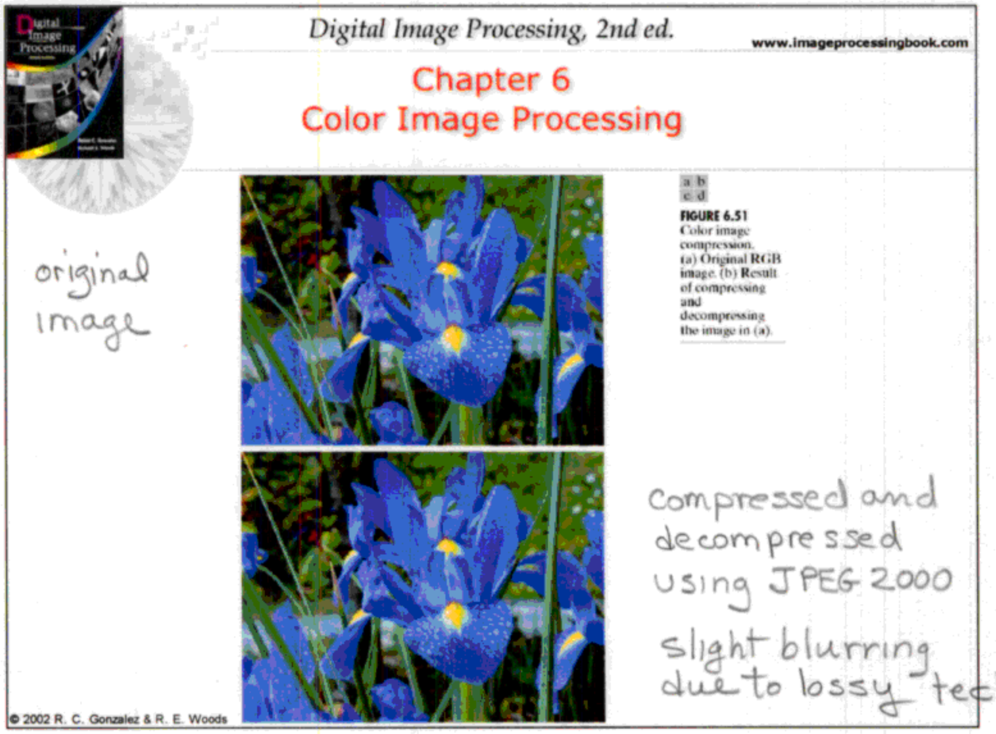
www.imageprocessingbook.com

Chapter 6
Color Image Processing

FIGURE 6.50 (a) RGB image with green plane corrupted by salt-and-pepper noise. (b) Hue component of HSI image. (c) Saturation component. (d) Intensity component.

RGB image
Salt & pepper noise
in Green channel

noise spreads to all HSI component images



Digital Image Processing, 2nd ed.

www.imageprocessingbook.com

Chapter 6
Color Image Processing

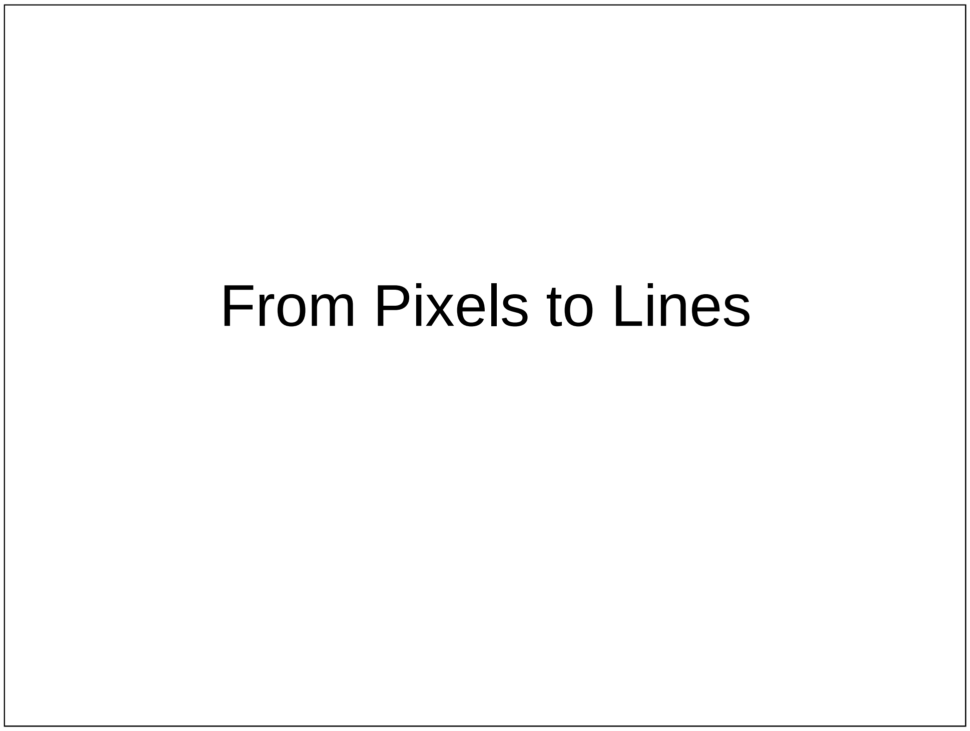
a b
c d

FIGURE 6.51 Color image compression. (a) Original RGB image. (b) Result of compressing and decompressing the image in (a).

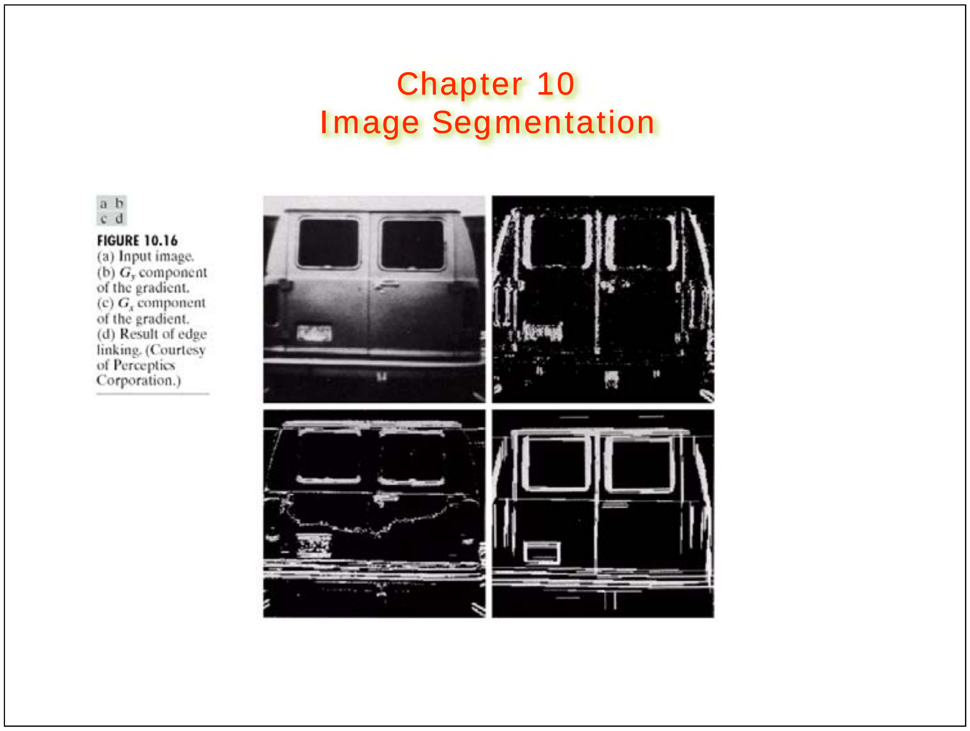
original image

compressed and decompressed using JPEG 2000

slight blurring due to lossy technique



From Pixels to Lines



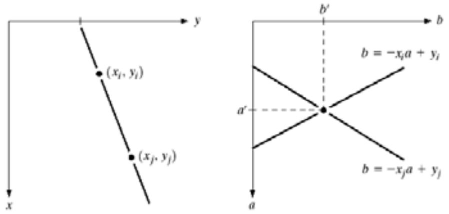
Chapter 10
Image Segmentation

FIGURE 10.16 (a) Input image. (b) G_y component of the gradient. (c) G_x component of the gradient. (d) Result of edge linking. (Courtesy of Perceptics Corporation.)

The final image is the result of linking all points that had a gradiant value >25 and whose direction did not differ by more than 15˚

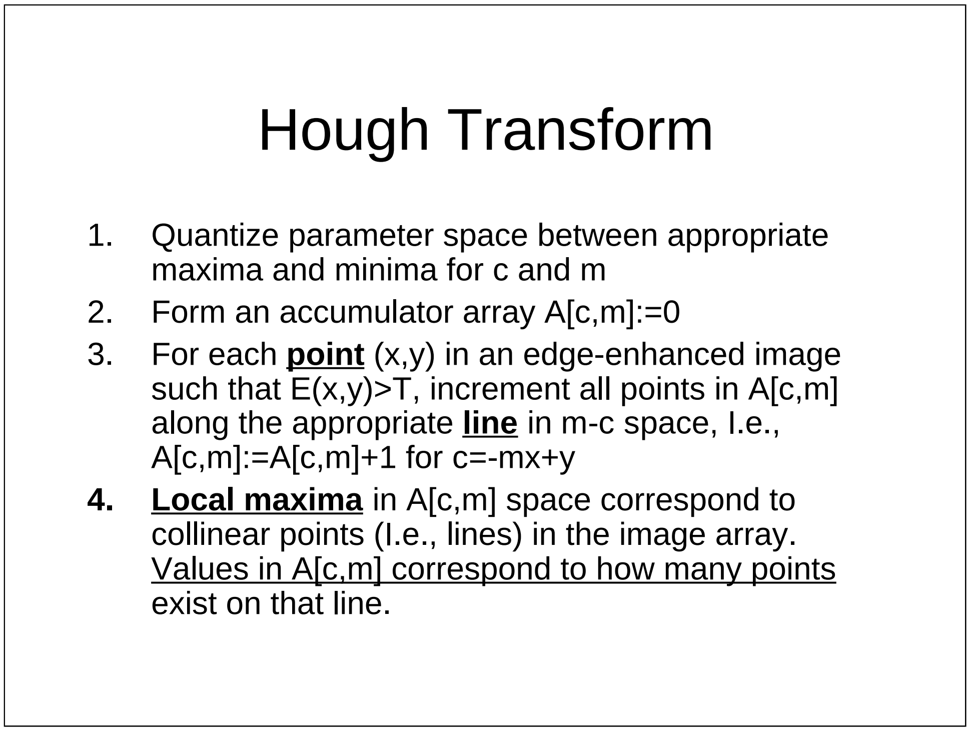
Additional processing to link line segments separated by short breaks and deleting short isolated segments.

Chapter 10
Image Segmentation



a b

FIGURE 10.17
(a) *xy*-plane.
(b) Parameter space.

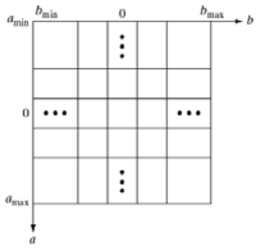


Hough Transform

1. Quantize parameter space between appropriate maxima and minima for c and m
2. Form an accumulator array A[c,m]:=0
3. For each **point** (x,y) in an edge-enhanced image such that E(x,y)>T, increment all points in A[c,m] along the appropriate **line** in m-c space, I.e., A[c,m]:=A[c,m]+1 for c=-mx+y
4. **Local maxima** in A[c,m] space correspond to collinear points (I.e., lines) in the image array. Values in A[c,m] correspond to how many points exist on that line.

Chapter 10
Image Segmentation

FIGURE 10.18 Subdivision of the parameter plane for use in the Hough transform.



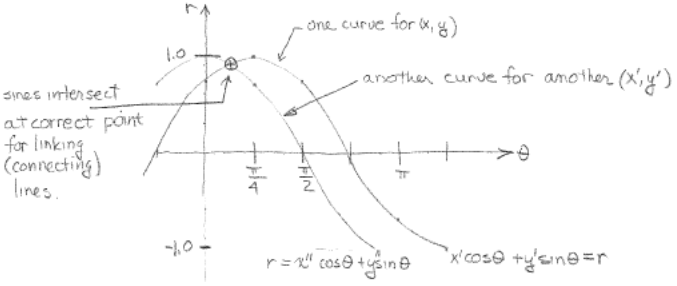
Hough Transforms

Problem : m is unbounded $-\infty < m < +\infty$

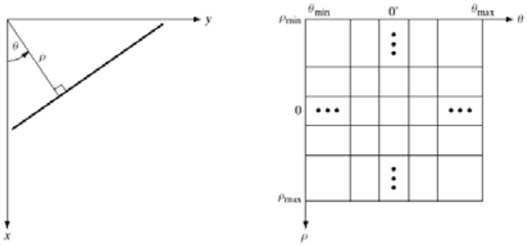
Solution : convert to polar coordinates <u>BEFORE</u> transforming.

i.e. $r = x\cos\theta + y\sin\theta$

This gives sine waves instead of straight lines in Hough space.



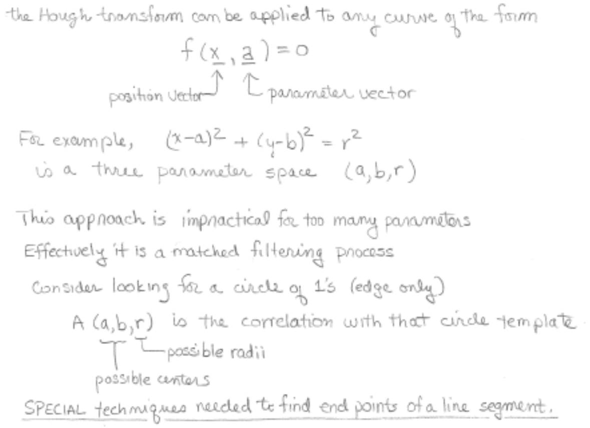
Chapter 10
Image Segmentation



a b

FIGURE 10.19 (a) Normal representation of a line. (b) Subdivision of the $\rho\theta$-plane into cells.

Hough Transforms



the Hough transform can be applied to any curve of the form

$$f(\underbrace{\underline{x}}_{\text{position vector}}, \underbrace{\underline{a}}_{\text{parameter vector}}) = 0$$

For example, $(x-a)^2 + (y-b)^2 = r^2$
is a three parameter space (a,b,r)

This approach is impractical for too many parameters

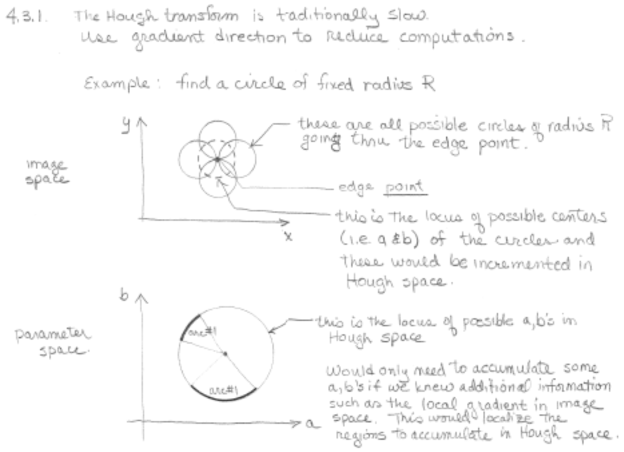
Effectively it is a matched filtering process

Consider looking for a circle of 1's (edge only)

$A(\underbrace{a,b}_{\text{possible centers}},\underbrace{r}_{\text{possible radii}})$ is the correlation with that circle template

<u>SPECIAL techniques needed to find end points of a line segment.</u>

Hough Transform



4.3.1. The Hough transform is taditionally slow.
Use gradient direction to reduce computations.

Example: find a circle of fixed radius R

these are all possible circles of radius R going thru the edge point.

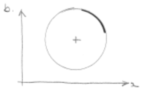
edge point

this is the locus of possible centers (i.e. a & b) of the circles and these would be incremented in Hough space.

this is the locus of possible a,b's in Hough space

Would only need to accumulate some a,b's if we knew additional information such as the local gradient in image space. This would localize the regions to accumulate in Hough space.

Hough Transform



Consider equation of circle

$$(x-a)^2 + (y-b)^2 = R^2$$

differentiating $2(x-a)\,dx + 2(y-b)\,dy = 0.$

$$\frac{dy}{dx} = \frac{x-a}{y-b} \qquad (1)$$

define $\frac{dy}{dx} = \tan\phi$ (the gradient of the edge in image space).

Solve for a & b in terms of x, y, R <u>and</u> ϕ

$$(x-a)^2 + (y-b)^2 = r^2$$

substituting(1) $\left[\frac{dy}{dx}(y-b)\right]^2 + (y-b)^2 = r^2$

$$(y-b)^2\left[\left(\frac{dy}{dx}\right)^2 + 1\right] = r^2$$

$$(y-b)^2\left[\tan^2\phi + 1\right] = r^2 = (y-b)^2\frac{1}{\cos^2\phi} = r^2$$

$$\therefore\ (y-b)^2 = r^2\cos^2\phi$$

$$y-b = \pm r\cos\phi$$

$$b = y \mp r\cos\phi$$

similarily for a using the equation of the circle and (1)

$$(x-a)^2 + (y-b)^2 = r^2$$

$$(x-a)^2 + \left[\frac{x-a}{\frac{dy}{dx}}\right]^2 = r^2$$

$$(x-a)^2\left[1 + \frac{1}{\frac{dy}{dx}}\right] = r^2$$

$$(x-a)^2\left[1 + \cot^2\phi\right] = r^2 = (x-a)^2\frac{1}{\sin^2\phi}$$

$$(x-a)^2 = r^2\sin^2\phi$$

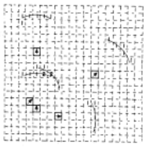
$$a = x \mp r\sin\phi.$$

$\Rightarrow$ Use $a = x \mp R\sin\phi$ R is known in this example

$b = y \mp R\cos\phi$ ϕ is known from gradient

(x,y) is location of edge candidate.

Although this gives a single point in (a,b) space. Do an arc thru (a,b) to cover errors in ϕ due to quantization, noise, et



Contents of accumulator array

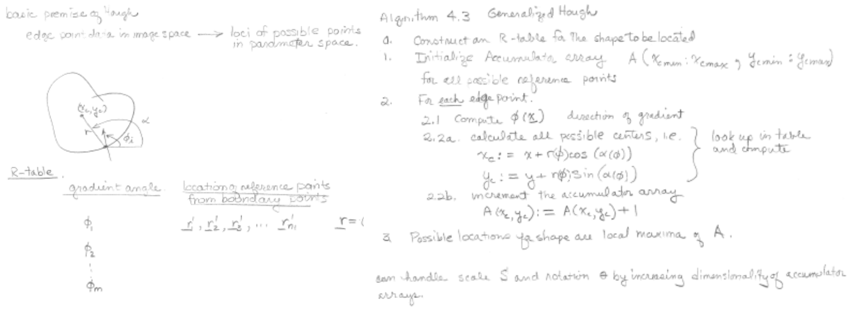
Gradient direction information for artifact $\Delta\phi = 45$

□ Denotes a pixel in $f(\underline{x})$ superimposed on accumulator array

➚ Denotes the gradient direction

Fig 4.6 Reduction in computation with gradient information

Generalized Hough Transform



basic premise of Hough

edge point data in image space → loci of possible points in parameter space.

(x_c, y_c)
r
α
ϕ_i

R-table.

gradient angle	location of reference points from boundary points
ϕ_1	$\underline{r}'_1, \underline{r}'_2, \underline{r}'_3, \dots \underline{r}'_{n_1}$
ϕ_2	
$\vdots$	
ϕ_m	

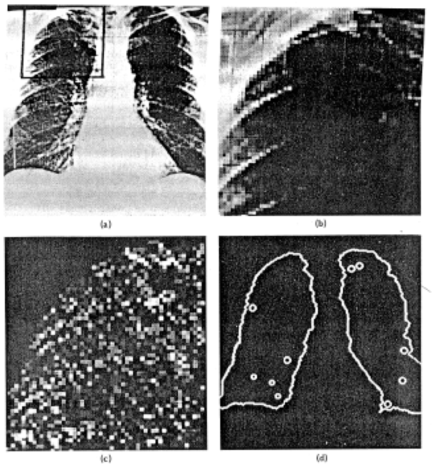
$\underline{r} = ($

Algorithm 4.3 Generalized Hough

```
0.  Construct an R-table for the shape to be located
1.  Initialize Accumulator array  $A(x_{cmin} : x_{cmax}, y_{cmin} : y_{cmax})$
    for all possible reference points
2.  For each edge point.
    2.1  Compute $\phi(\underline{x})$  direction of gradient
    2.2a. calculate all possible centers, i.e.      } look up in table
              $x_c := x + r(\phi)\cos(\alpha(\phi))$  } and compute
              $y_c := y + r(\phi)\sin(\alpha(\phi))$  }
    2.2b. increment the accumulator array
              $A(x_c, y_c) := A(x_c, y_c) + 1$
3.  Possible locations for shape are local maxima of A.
```

can handle scale S and rotation θ by increasing dimensionality of accumulator arrays.

Hough Transform



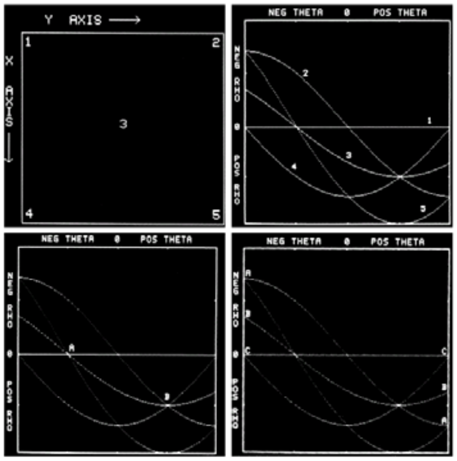
(a) (b) (c) (d)

Fig. 4.7 Using the Hough technique for circular shapes. (a) Radiograph. (b) Window. (c) Accumulator array for $r = 3$. (d) Results of maxima detection.

Chapter 10
Image Segmentation

a b
c d

FIGURE 10.20 Illustration of the Hough transform. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)



Chapter 10
Image Segmentation

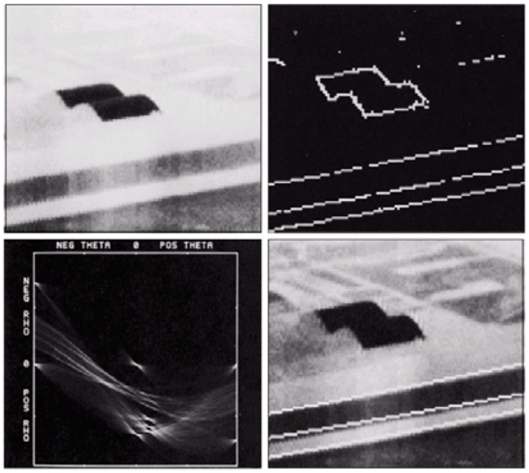


FIGURE 10.21 (a) Infrared image. (b) Thresholded gradient image. (c) Hough transform. (d) Linked pixels. (Courtesy of Mr. D. R. Cate, Texas Instruments, Inc.)

Chapter 10
Image Segmentation

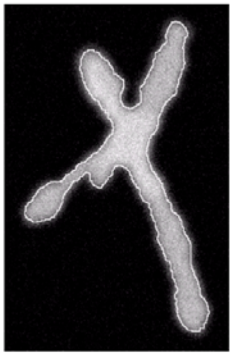
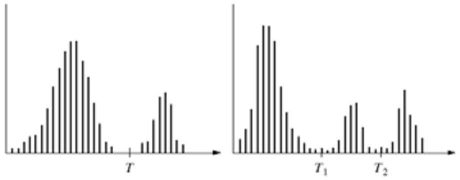


FIGURE 10.25 Image of noisy chromosome silhouette and edge boundary (in white) determined by graph search.

Chapter 10
Image Segmentation



a b

FIGURE 10.26 (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.