Computer Vision

Spring 2006 15-385,-685

Instructor: S. Narasimhan

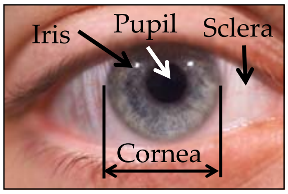
Wean 5403

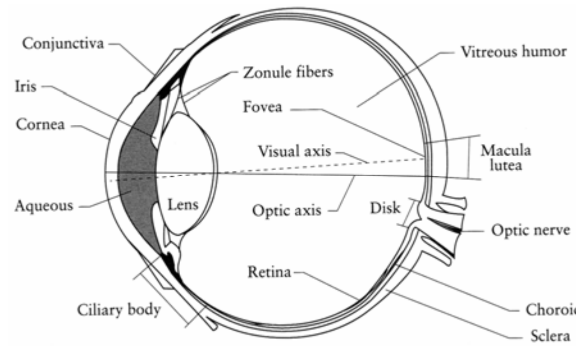
T-R 3:00pm – 4:20pm

Announcements

- Homework 1 is due on Thursday in class.
- Homework 2 will be out on Thursday.
- Start homeworks early.
- Post questions on bboard.

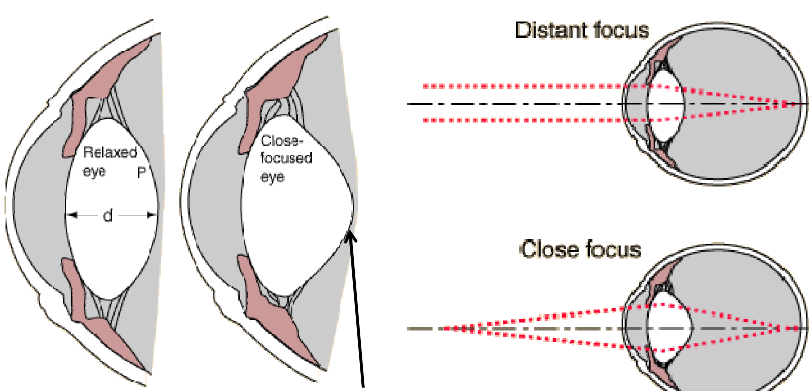
Our Eyes





- Index of refraction: cornea 1.376, aqueous 1.336, lens 1.406-1.386
- Iris is the diaphragm that changes the aperture (pupil)
- Retina is the sensor where the fovea has the highest resolution

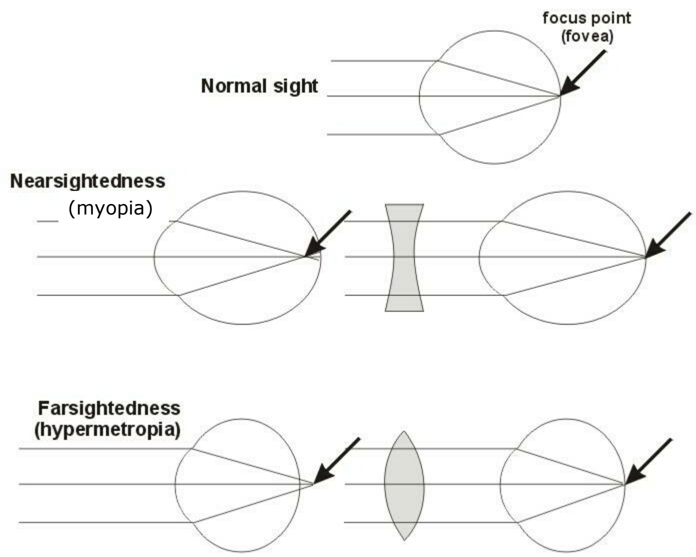
Accommodation



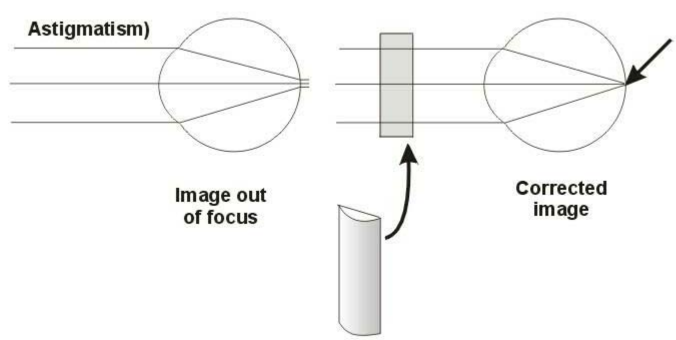
shorter focal length

Changes the focal length of the lens

Myopia and Hyperopia



Astigmatism

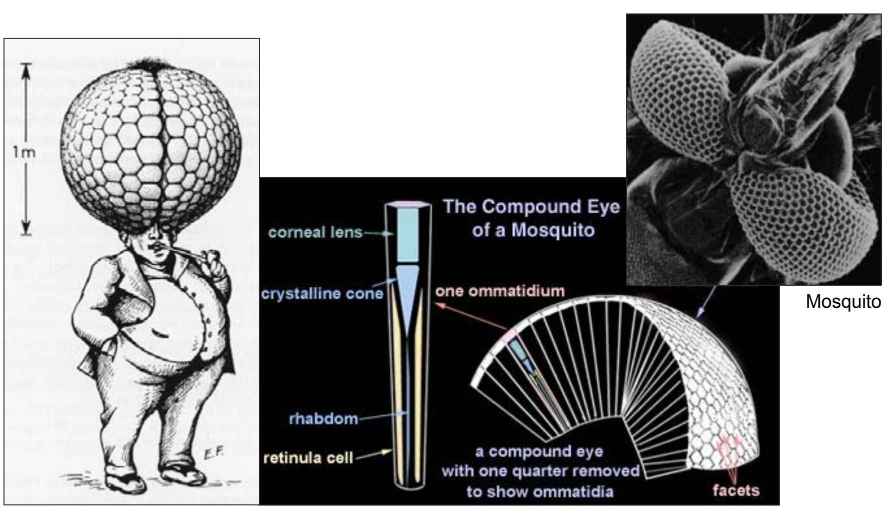


The cornea is distorted causing images to be un-focused on the retina.

Blind Spot in Eye

Close your right eye and look directly at the "+"

Eyes in Nature

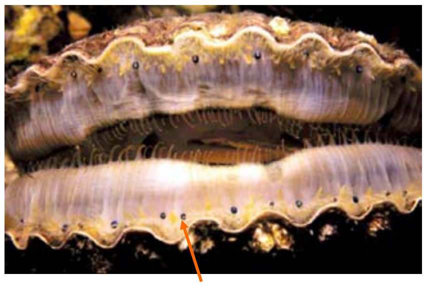


Mosquito

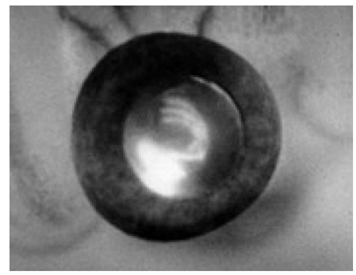
http://ebiomedia.com/gall/eyes/octopus-insect.html

Mosquitos have microscopic vision, but to focus at large distances their would need to be 1 m!

Curved Mirrors in Scallop Eyes



Telescopic Eye



(by Mike Land, Sussex)

… More in the last part of the course

Binary Images: Properties and Methods

Lecture #4

Binary Images

- Images with only two values (0 or 1)
- Simple to process and analyze
- Very useful for industrial applications

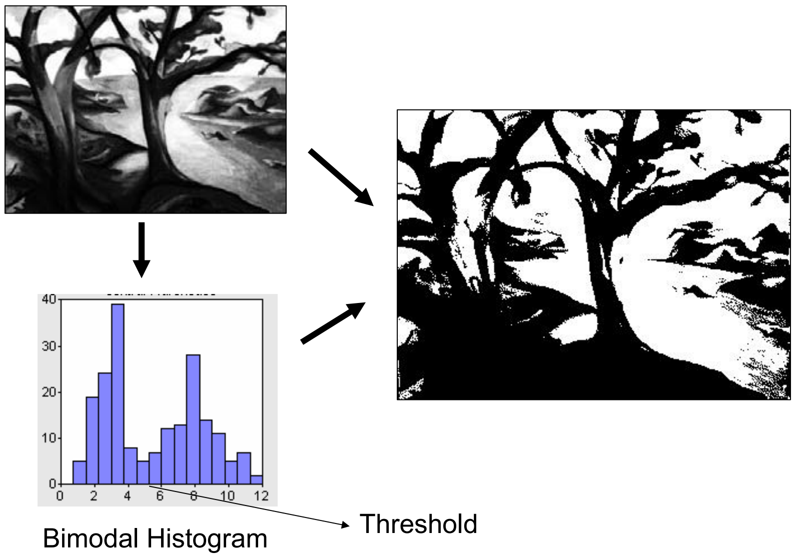
Binary Images

- Obtained from gray-level (or color) image $g(x, y)$ by Thresholding
- Characteristic Function

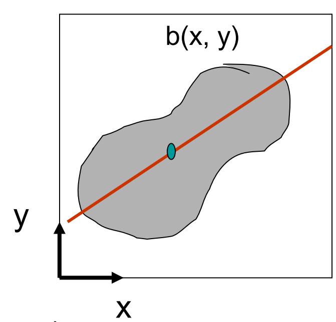
$$b(x, y) = \begin{cases} 1 & \text{if } g(x, y) < T \\ 0 & \text{if } g(x, y) >= T \end{cases}$$

- Topics Discussed:
  - Geometric Properties
  - Continuous and Discrete Binary Images
  - Multiple Objects (Connectivity)
  - Sequential (iterative) processing

Selecting a Threshold



Geometric Properties of Binary Images



- Assume:

  b(x, y) is continuous
  only one object

- Area: Zeroth Moment

$$A = \iint b(x, y)\, dx\, dy$$

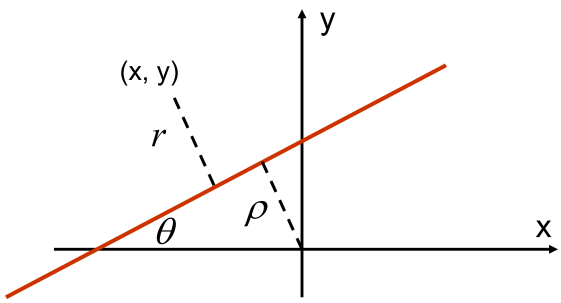
- Position: Center of Mass (First Moment)

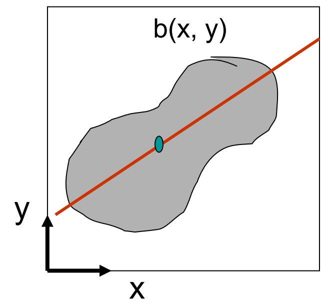
$$\bar{x} = \frac{1}{A} \iint x\, b(x, y)\, dx\, dy$$

$$\bar{y} = \frac{1}{A} \iint y\, b(x, y)\, dx\, dy$$

Geometric Properties of Binary Images

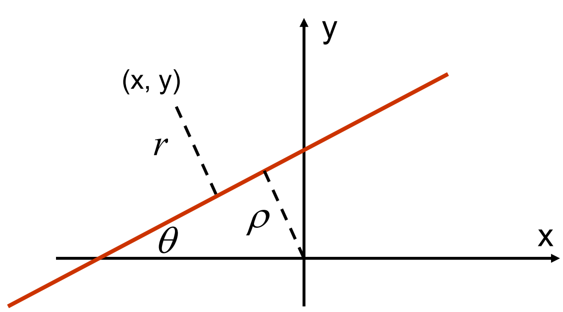
- Orientation: Difficult to define!
  - Axis of least second moment
  - For mass: Axis of minimum inertia





Minimize: $E = \iint r^2 \, b(x, y) \, dx \, dy$

Which equation of line to use?



$$y = mx + b \, ? \qquad 0 \leq m \leq \infty$$

We use:

$$x \sin\theta - y \cos\theta + \rho = 0$$

$\theta \quad \rho$
are finite

Minimizing Second Moment

Find θ and ρ that minimize E for a given b(x,y)

We can show that: $r = x\sin\theta - y\cos\theta + \rho$

So, $E = \iint (x\sin\theta - y\cos\theta + \rho)^2\, b(x,y)\, dx\, dy$

Using $\frac{dE}{d\rho} = 0$ we get: $A(\bar{x}\sin\theta - \bar{y}\cos\theta + \rho) = 0$

Note: Axis passes through the center $(\bar{x}, \bar{y})$

So, change co-ordinates: $x' = x - \bar{x}, \quad y' = y - \bar{y}$

Minimizing Second Moment

We get: $E = a\sin^2\theta - b\sin\theta\cos\theta + c\cos^2\theta$

where,

$$a = \iint (x')^2\, b(x,y)\, dx'\, dy'$$

$$b = 2\iint (x'y')\, b(x,y)\, dx'\, dy'$$

$$c = \iint (y')^2\, b(x,y)\, dx'\, dy'$$

- second moments w.r.t $(\bar{x}, \bar{y})$

We are not done yet!!

Minimizing Second Moment

$$E = a\sin^2\theta - b\sin\theta\cos\theta + c\cos^2\theta$$

Using $\frac{dE}{d\theta} = 0$ we get: $\tan 2\theta = \frac{b}{a-c}$

$$\sin 2\theta = \pm\frac{b}{\sqrt{b^2+(a-c)^2}} \qquad \cos 2\theta = \pm\frac{a-c}{\sqrt{b^2+(a-c)^2}}$$

Solutions with +ve sign must be used to minimize E. (Why?)

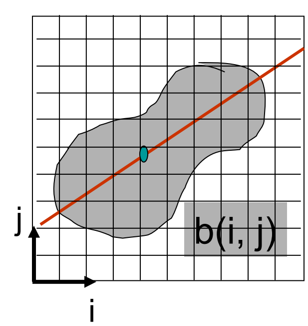
$$\frac{E_{\min}}{E_{\max}} \longrightarrow roundedness$$

Discrete Binary Images

- Assume:

  b(x, y) is discrete
  only one object

- Area: Zeroth Moment $A = \sum\sum b_{ij}$



- Position: Center of Mass (First Moment)

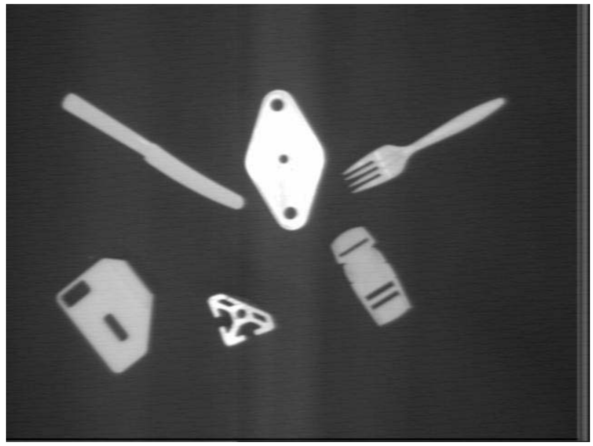
$$\bar{x} = \frac{1}{A}\sum\sum i\, b_{ij} \qquad \bar{y} = \frac{1}{A}\sum\sum j\, b_{ij}$$

- Second Moments:

$$a' = \sum\sum i^2\, b_{ij} \qquad b' = 2\sum\sum ij\, b_{ij} \qquad c' = \sum\sum j^2\, b_{ij}$$

Note: a',b',c' are defined w.r.t origin

Multiple Objects



Need to SEGMENT image into separate COMPONENTS (regions)

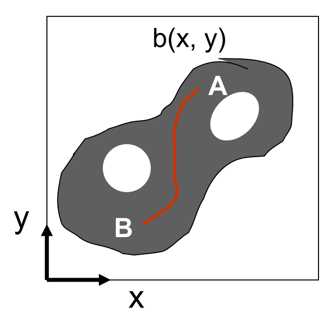
- (Non-trivial !)

Connected Components

Maximal Set of Connected points

Remember Graph Theory?

A & B are connected: Path exists between A & B along which b(x,y) is constant.



Connected Component Labeling

Region Growing Algorithm:

(a) Start with "SEED" point where b(x,y) = 1

(b) Assign LABEL to seed point

(c) Assign SAME LABEL to its Neighbors with b(x,y) = 1

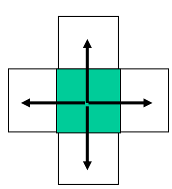
(d) Assign SAME LABEL to Neighbors of Neighbors

Terminates when a component is completely labeled.

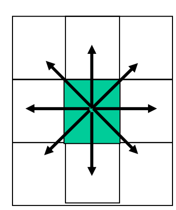
Then, pick another UNLABELED seed point.

What do we mean by Neighbors?

Connectedness:



4-connectedness



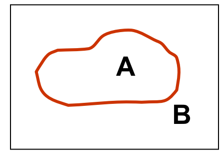
8-connectedness

Neither is perfect!

What do we mean by Neighbors?

- Jordan's Curve Theorem:

  Closed Curve → 2 connected regions



- Consider:

0	1	0
1	0	1
0	1	0

→

(4-C)

B1	O1	B1
O2	B2	O3
B1	O4	B1

Hole without Closed curve!

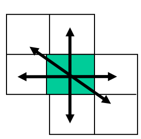
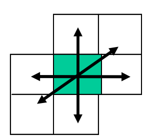
(8-C)

B	O	B
O	B	O
B	O	B

Connected Backgrounds with closed Ring!

Solution to Neighborhood Problem

- Introduce Asymmetry

(a)  OR (b) 

- Using (b)

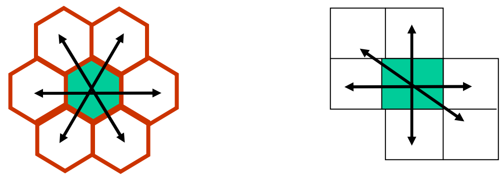
0	1	0
1	0	1
0	1	0

→

B	O1	B
O1	B	O2
B	O2	B

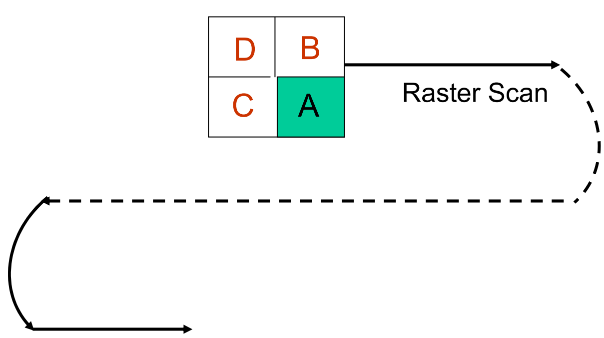
Two separate Line Segments

Hexagonal Tessellation



Asymmetry makes a SQUARE grid like HEXAGONAL grid

Sequential Labeling Algorithm



Note: We want to label A.

B, C, D are already labeled.

Sequential Labeling Algorithm

0	0
0	1

→ label(A) = new label

X	X
X	0

→ label(A) = “background”

D	X
X	1

→ label(A) = label(D)

0	0
C	1

→ label(A) = label(C)

0	B
0	1

→ label(A) = label(B)

0	B
C	1

→ If
label(B) = label(C)
then,
label(A) = label(B)

Sequential Labeling Algorithm

0	B
C	1

→ What if label(B) not equal to label(C)?

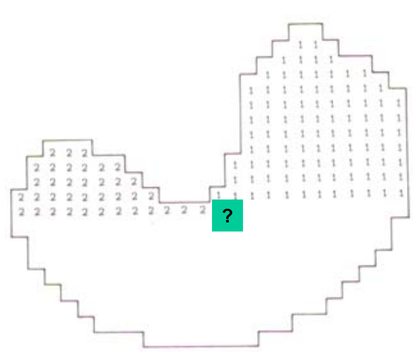


Figure 4-4. In the sequential labeling process we might discover that two regions previously thought to be separate are in fact connected. We must make a note of the equivalence of the two labels. (Reproduced by permission from *Lisp* by P.H. Winston & B.K.P. Horn, Addison-Wesley, Reading, Massachusetts, 1984.)

Sequential Labeling Algorithm

0	B
C	1

→ What if label(B) not equal to label(C)?

Solution:

Let: label(A) = label(B) = 2

Create EQUIVALENCE TABLE

$2 \equiv 1$
$7 \equiv 3,6,4$
⋮

Resolve Equivalence in Second Pass

Morphological Operations

- Euler Number

Number of Bodies (B) – Number of Holes (H)

B : E = -1

i : E = 2

A : E = 0

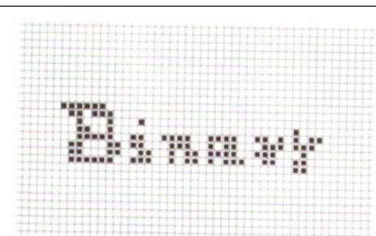


Figure 4-7. The Euler number is the difference between the number of objects and the number of holes. The Euler number of this binary image is 4 since there are 7 objects and 3 holes.

$$E_{image} = \sum E_{non-overlapping\ regions}$$

Iterative Modification

- Allows us to incrementally change image
- Example:

  Etch (thin) object to produce SKELETON

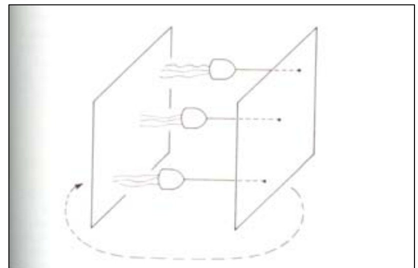
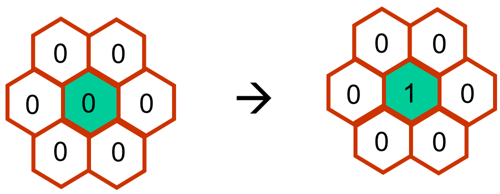


Figure 4-13. A new binary image is developed by means of many computing cells, each connected to a small number of neighboring picture cells.

- Conservative Operations:
  - Do not change the Euler number of image.



E=0

E=1

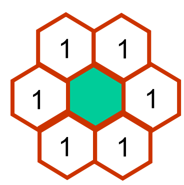
New Body!

Euler Differential = +1

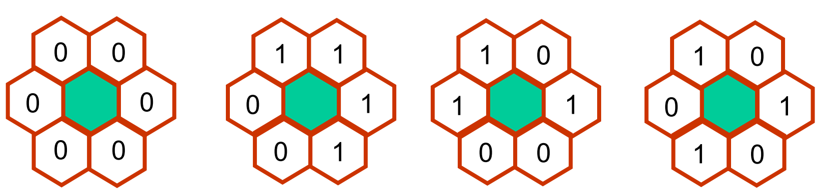
Euler Differential E*

- Note: If E* = +1, for center pixel, 0 → 1

  If E* = -1, for center pixel, 1 → 0
- Each pixel has 64 neighborhoods.
- We want to find Neighborhoods with E* = 0



E* = +1



E* = +1

E* = 0

E* = -1

E* = -2

- Only 5 different sets of neighborhoods possible

$$N_{+1}, N_0, N_{-1}, N_{-2}$$

Problem with Parallel Implementation

0	0	0	0
0	1	1	0
0	0	0	0

→

0	0	0	0
0	0	0	0
0	0	0	0

$E^* \neq 0\,!$

Solution: Use Interlaced Subfields

- No two neighbors have the same label
- Apply parallel operations to ONE subfield at a time

1	2	3	1	2	3
2	3	1	2	3	1
3	1	2	3	1	2
1	2	3	1	2	3
2	3	1	2	3	1
3	1	2	3	1	2

Notation for Iterative Modification

- Specify Neighborhood Set, *S,* for which the same action must be taken

  For example: $S = N_0$

- Consider pixel (i, j):
  - $a_{ij} = 1 \quad if\ \ Neighborhood\ of\ \ (i, j) \in S$
  - $b_{ij} = value\ of\ \ pixel\ (i, j)$
  - $c_{ij} = new\ value\ of\ \ pixel\ (i, j)$

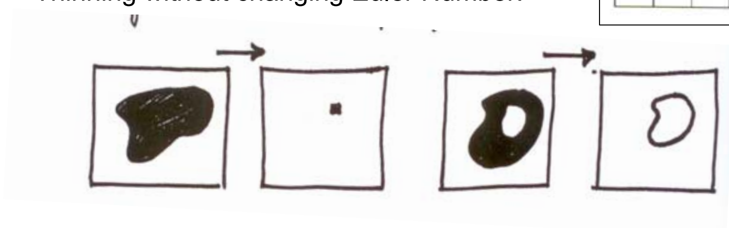
- 16 Boolean Functions

ab	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
00	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
01	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
10	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
11	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Finding Skeletons

- Use $S = N_0$ (Zero Euler Differential Set)
- Use 5th Column in the Boolean Functions Table
- Thinning without changing Euler Number!

ab	0	1	2	3	4	5
00	0	0	0	0	0	0
01	0	0	0	0	1	1
10	0	0	1	1	0	0
11	0	1	0	1	0	1



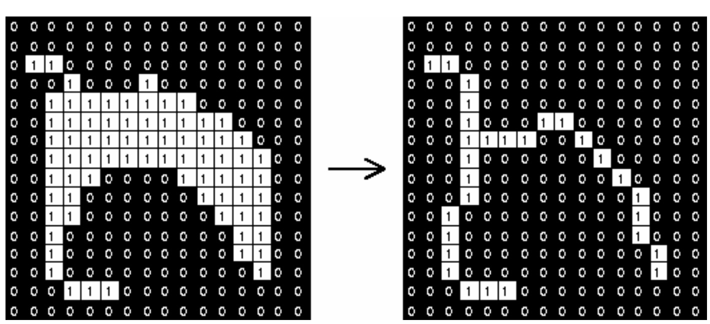
- Other Tricks:

  * Expanding: Thinning the Background
  * Noise Removal: Thin and Expand

Finding Skeletons

- Use $S = N_0$ (Zero Euler Differential Set)
- Use 5th Column in the Boolean Functions Table
- Thinning without changing Euler Number!

ab	0	1	2	3	4	5
00	0	0	0	0	0	0
01	0	0	0	0	1	1
10	0	0	1	1	0	0
11	0	1	0	1	0	1



Announcements

- Homework 1 is due on Thursday in class.
- Homework 2 will be out on Thursday.
- Start homeworks early.
- Post questions on bboard.

Next Two Classes

- 1D Signal and 2D Image Processing
- Horn, Chapter 6