EE 569 Discussion

Zohreh Azizi

03/05/2021

Announcements

- HW3

  Due: 03/11/21 11:59PM

Announcements

Preview of Discussion Sessions:

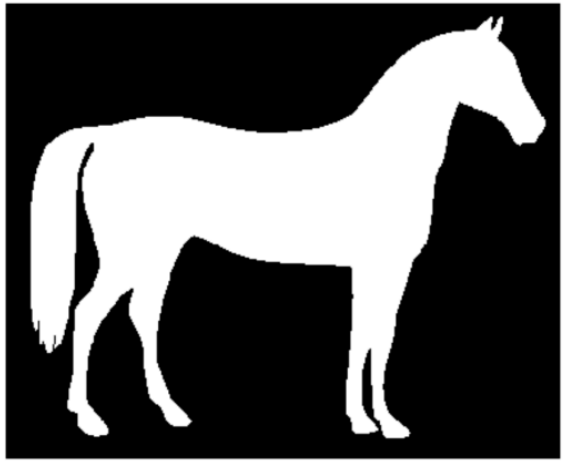
- …
- Week 8 (Mar. 5): Morphological Processing (HW3, Problem 3)
- Week 9 (Mar. 12): No Discussion - USC Spring 2021 Wellness Day
- Week 10 (Mar. 19): HW4 part 1
- Week 11 (Mar. 26): HW4 part 2
- …

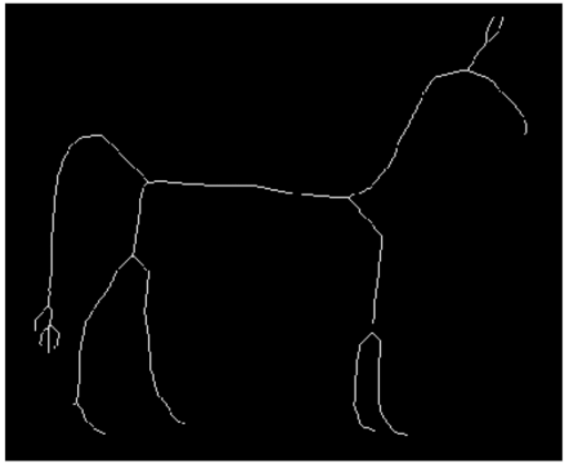
Overview

- Basic Morphological Processing
- Solution to the maze
- Defect detection and count

Basic Morphological Processing

- Shrinking: central pixel
- Thinning: central line
- Skeletonizing: skeleton containing edges





Basic Morphological Processing

- Shrinking:

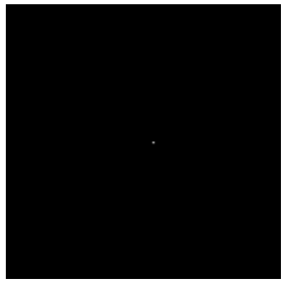
Objects without holes → a point

Objects with holes → a connected ring halfway between each hole and the outer boundary.

- Thinning:

Object without holes → a minimally connected stroke

Object with holes → a connected ring halfway between each hole and the outer boundary.

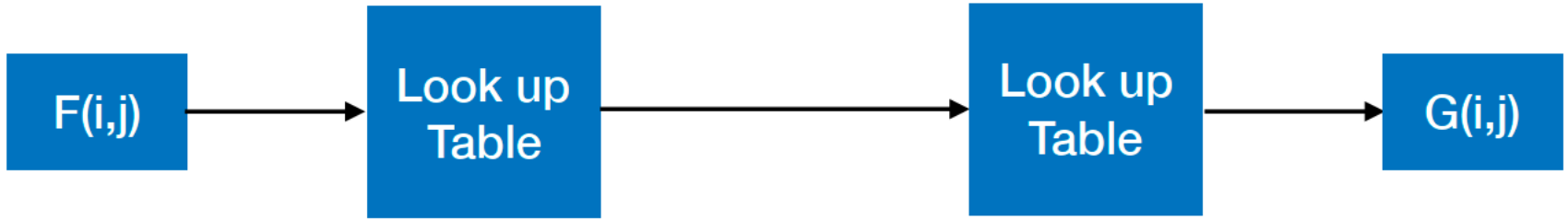


- Skeletonizing:

Removes pixels on the boundaries of objects but does not allow objects to break apart. The remaining pixels make up the image skeleton.

Basic Morphological Processing (P. 3a)

- Implement: Look up Table
- 2-pass filter



- Note:
  - **Conditional MARK logic If match: mark the pixel**
  - **Unconditional MARK logic If match: keep the pixel**
  - Table 14.3-1 Shrink /Thin /Skeleton conditional LUT
  - Table 14.3-2 Shrink/ Thin unconditional LUT
  - Table 14.3-3 Skeleton unconditional LUT

Basic Morphological Processing (P. 3a)

Table 14.3-1 Shrink, Thin and Skeletonize Conditional Mark Patterns (M=1 if hit)

Type	Bond	Patterns
S	1	0 0 1 / 0 1 0 / 0 0 0 ; 1 0 0 / 0 1 0 / 0 0 0 ; 0 0 0 / 0 1 0 / 1 0 0 ; 0 0 0 / 0 1 0 / 0 0 1
S	2	0 0 0 / 0 1 1 / 0 0 0 ; 0 1 0 / 0 1 0 / 0 0 0 ; 0 0 0 / 1 1 0 / 0 0 0 ; 0 1 0 / 0 1 0 / 0 1 0
S	3	0 0 1 / 0 1 1 / 0 0 0 ; 0 1 1 / 0 1 0 / 0 0 0 ; 1 1 0 / 0 1 0 / 0 0 0 ; 1 0 0 / 1 1 0 / 0 0 0 ; 0 0 0 / 0 1 0 / 1 1 0 ; 0 0 0 / 1 1 0 / 0 1 0 ; 0 0 0 / 0 1 0 / 0 1 1 ; 0 0 1 / 0 1 1 / 0 0 0

Table 14.3-2 Shrink and Thin Unconditional Mark Patterns

Spur	0 0 M / 0 M 0 / 0 0 0 ; M 0 0 / 0 M 0 / 0 0 0
Single 4-connection	0 0 0 / 0 M 0 / 0 M 0 ; 0 0 0 / 0 M M / 0 0 0
L Cluster	0 0 M / 0 M M / 0 0 0 ; 0 M M / 0 M 0 / 0 0 0 ; M M 0 / 0 M 0 / 0 0 0 ; M 0 0 / M M 0 / 0 0 0

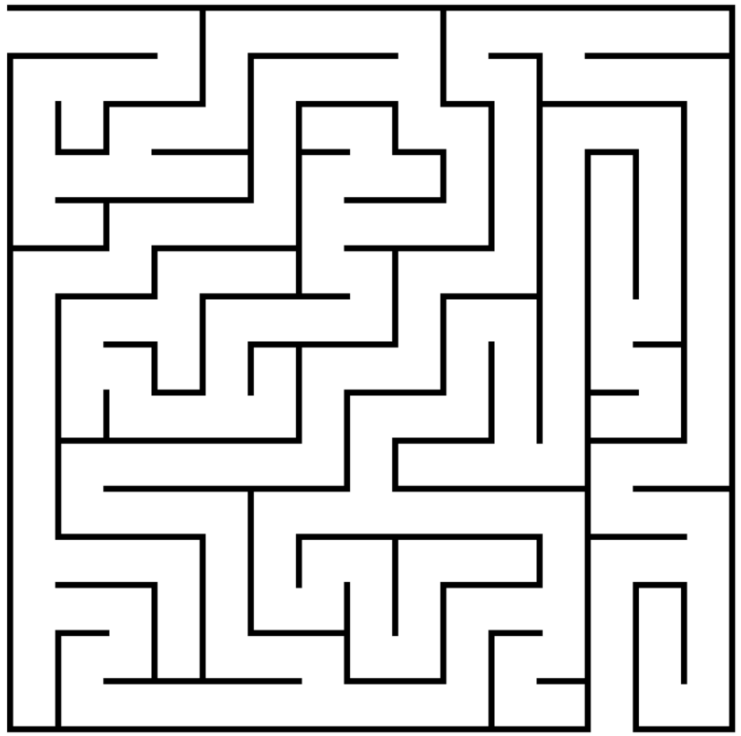
Table 14.3-3 Skeletonize Unconditional Mark Patterns

Spur	0 0 0 / 0 M 0 / 0 0 M ; 0 0 0 / 0 M 0 / M 0 0 ; 0 0 M / 0 M 0 / 0 0 0 ; M 0 0 / 0 M 0 / 0 0 0
Single 4-connection	0 0 0 / 0 M 0 / 0 M 0 ; 0 0 0 / 0 M M / 0 0 0 ; 0 0 0 / 0 M 0 / 0 M 0 ; 0 0 0 / M M 0 / 0 0 0
L Corner	0 M 0 / 0 M M / 0 0 0 ; 0 M 0 / M M 0 / 0 0 0 ; 0 0 0 / M M 0 / 0 M 0 ; 0 0 0 / 0 M M / 0 M 0

1. Image: Do zero padding
2. M = zeros()
3. Table 14.3-1 on image: if match, mark on M → 1
4. → M = zeros and ones
5. Tables 14.3-2 & 14.3-3 on M:
   - If match, keep the pixel in image
   - If not match, change pixel to 0 in image
6. Repeat steps 2-5, until no more pixel is changed to 0, in step 5.

Solution to the maze

- Do shrinking so that the solution to the maze appears to you.
- Track and report the intermediate steps.

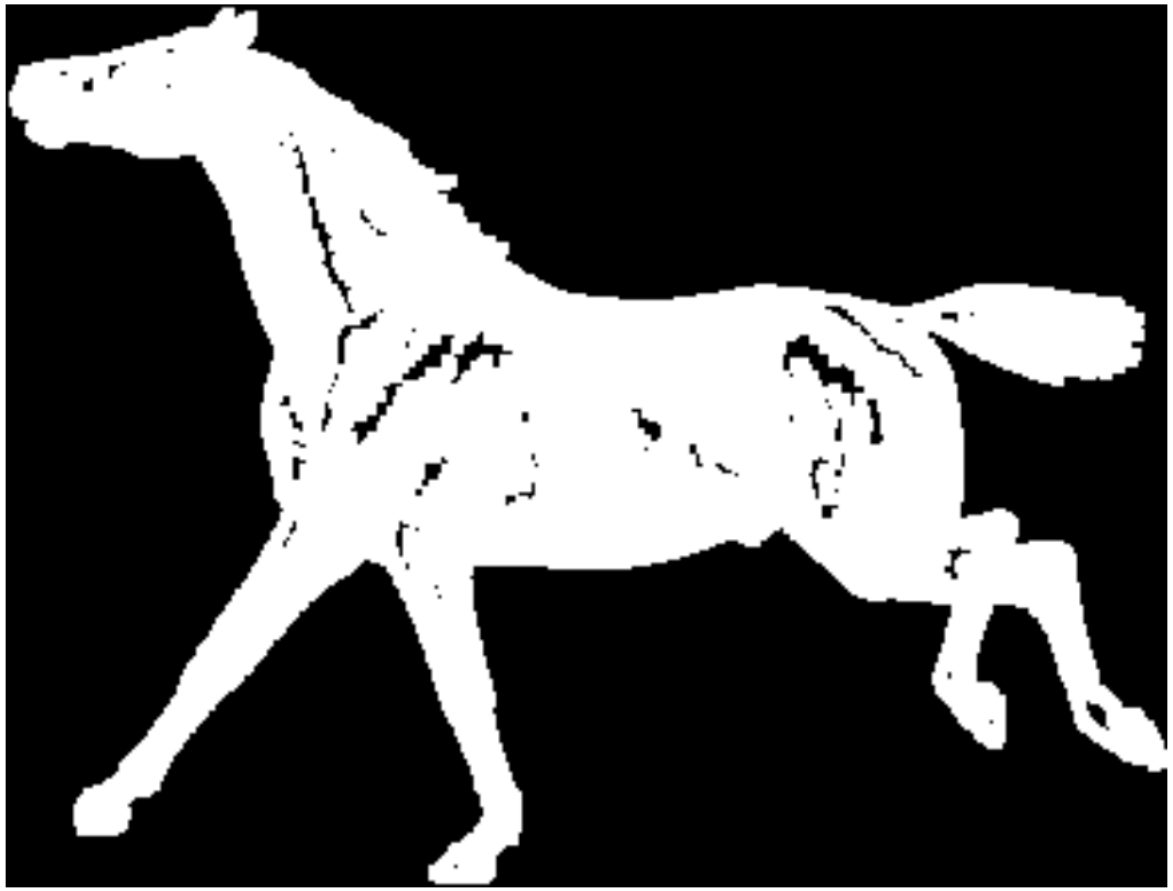


Defect Detection and Count

(1) Count the number of disconnected defects:

- Method 1:
  1. Do shrinking
  2. Count the white loops

- Method 2:
  1. Inverse the image
  2. Do shrinking
  3. Count the white dots

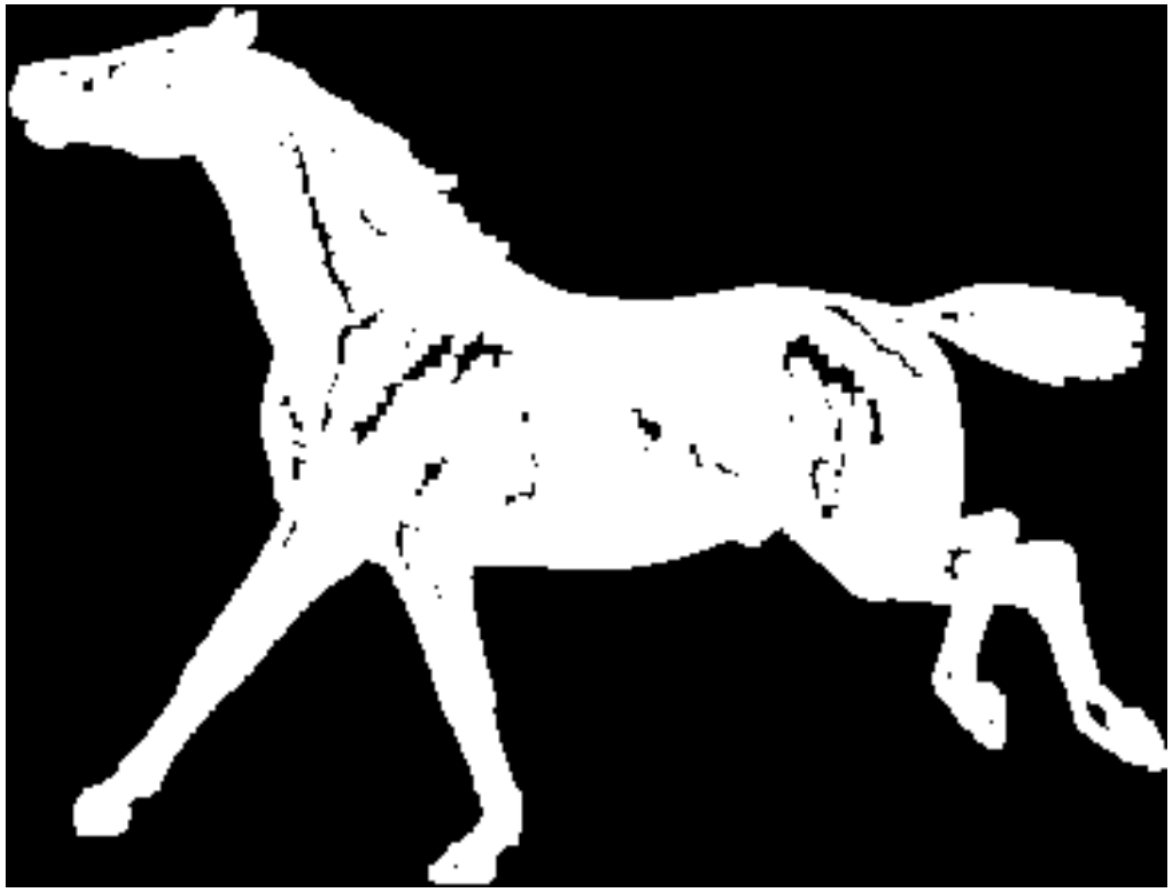


Defect Detection and Count

(2) Different sizes and their frequency:

- Size:
  Hint: how many shrinking operation you applied?
  Or other ideas…

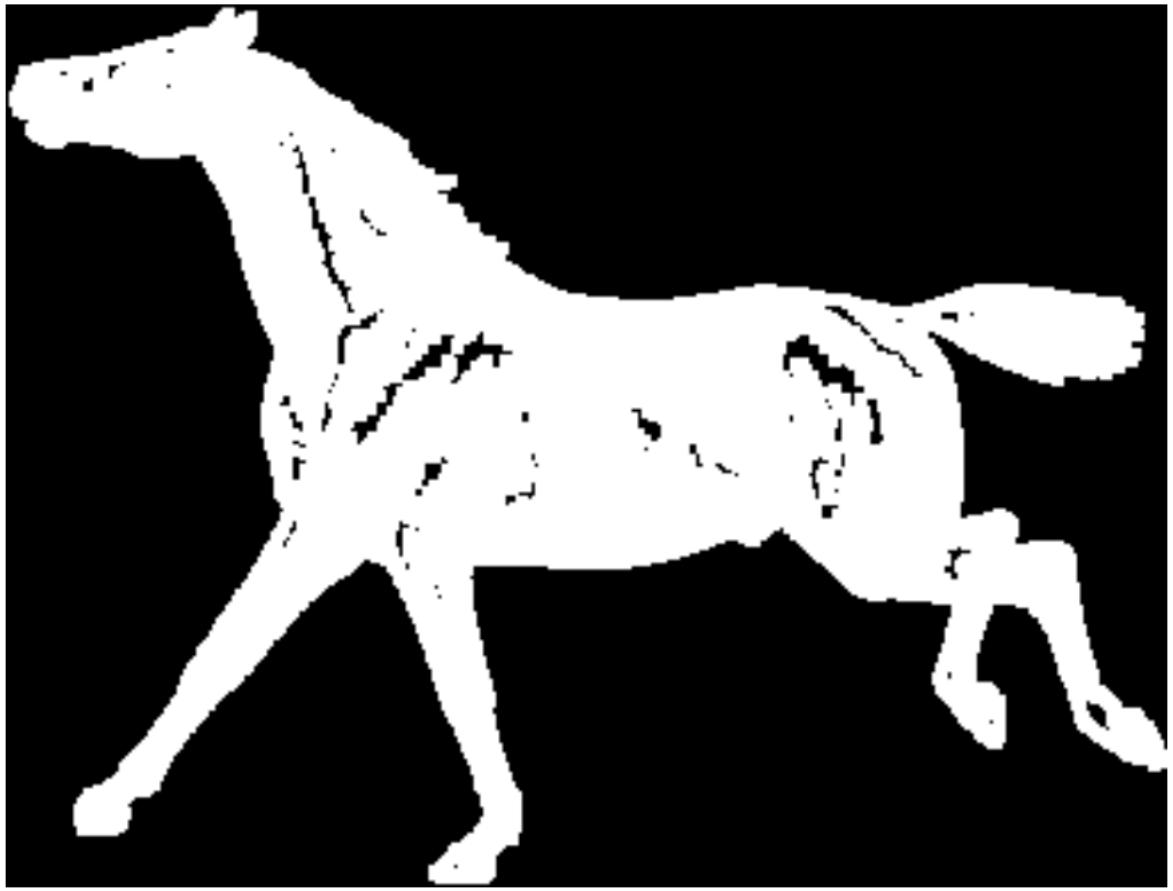
- Frequency:
  Histogram of the number of each size



Defect Detection and Count

(2) Defect clearing:

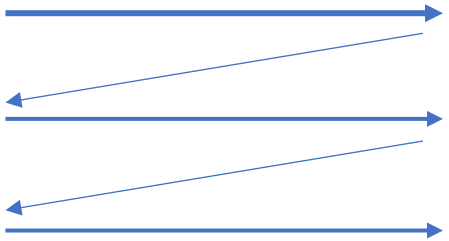
- Hint: thinning on the inverse image gives the approximate location of the defects.
  Or other ideas…



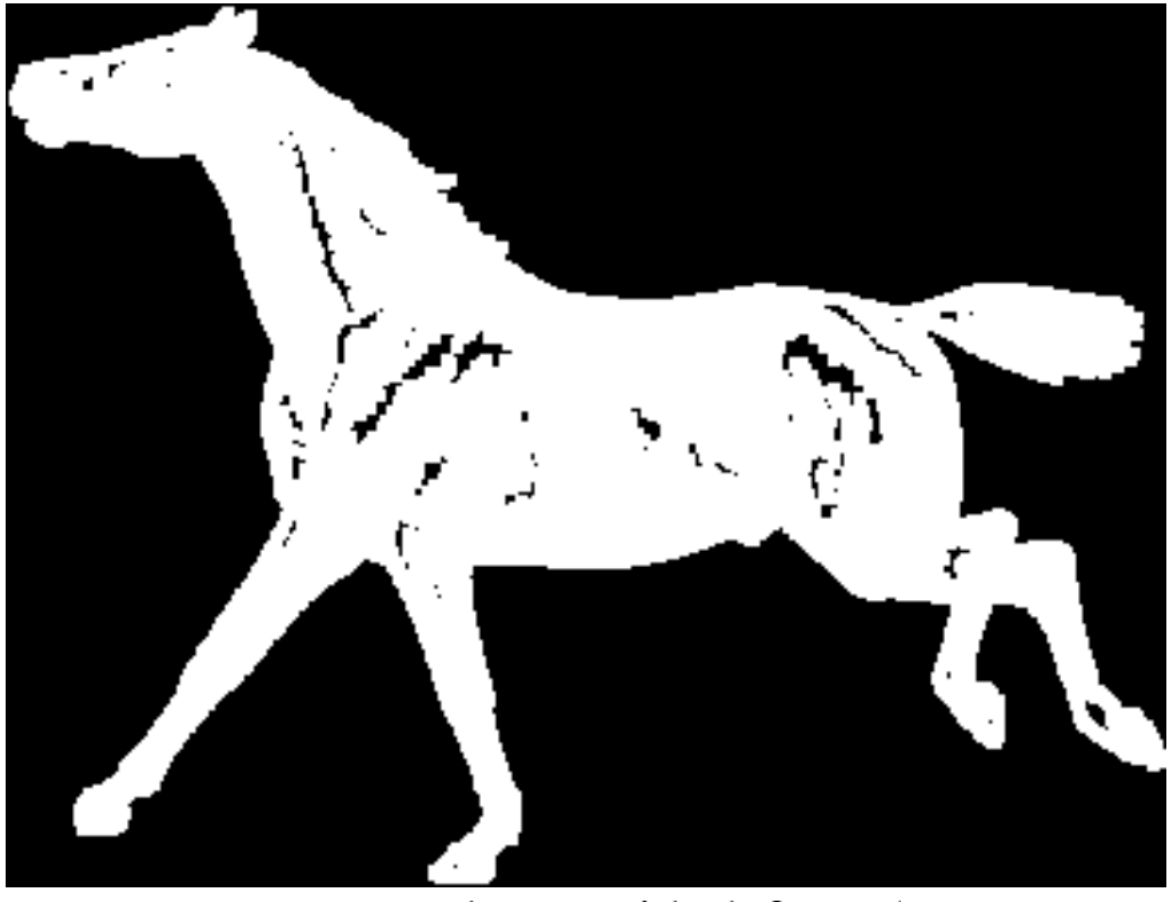
Defect Detection and Count

(4) Connected Component Labeling (CCL):

- Binary:
  - Object: 1
  - Background: 0
- Raster scan:



- Zero pad



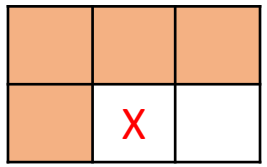
Defect Detection and Count

(4) ***1st pass*** of CCL:

1. Start scanning through the whole image (raster scan).

2. If the element is not a background;

   i. See the neighboring element's labels:



   ii. If there is no neighboring labels, assign a new label to current element.

   iii. If there are neighbors with labels, assign current element with minimum neighboring label.

2	0	0
0	X	

X ← 2

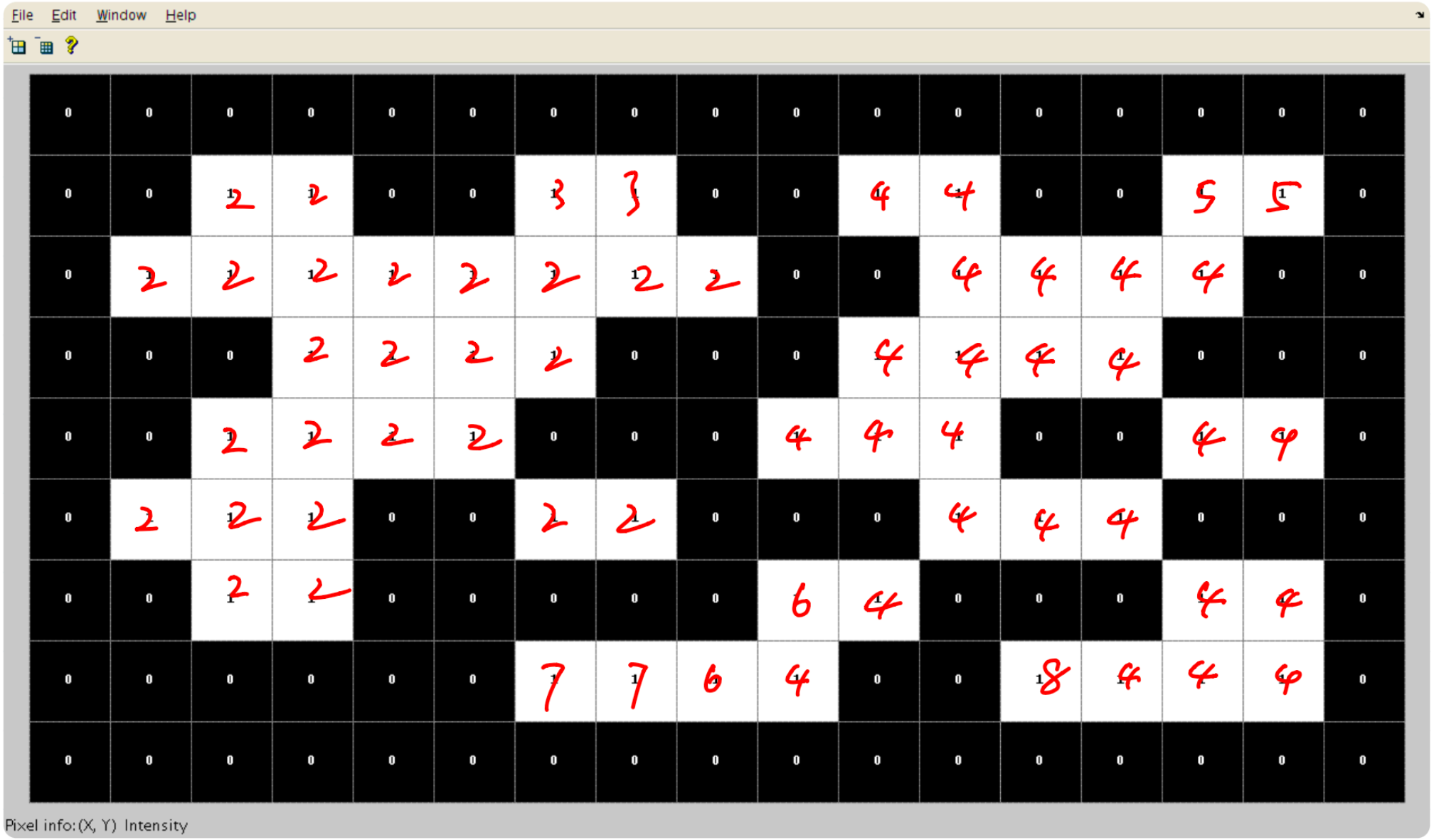
3	3	3
5	X	

X ← 3

   iv. Update equivalence table: 3, 5

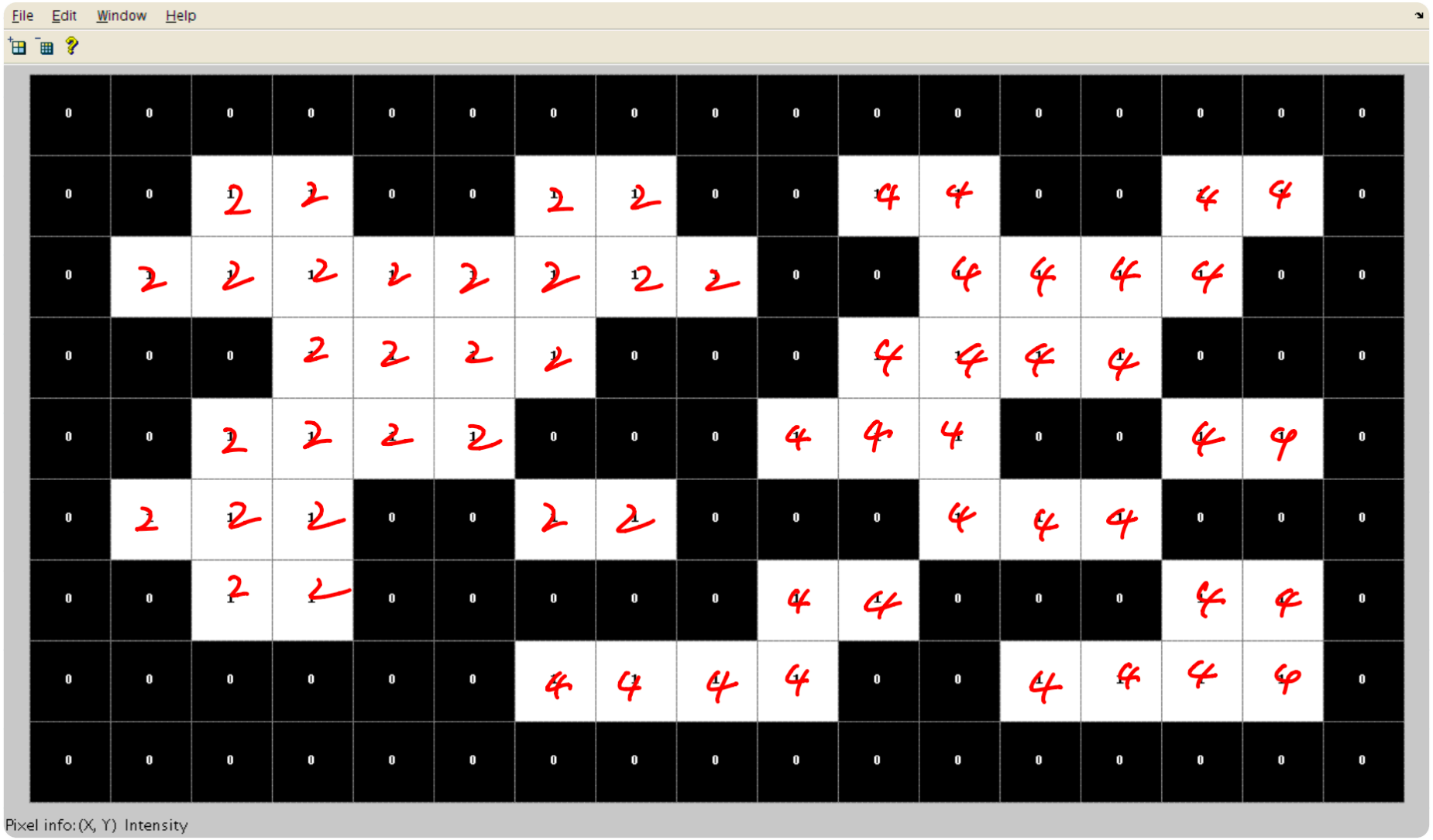
Defect Detection and Count

1st pass of CCL:



Defect Detection and Count

(4) ***2nd pass*** of CCL: Update labels according to equivalence table.



References

- Yao Zhu, EE569 Spring 2020 discussion 6 & 7


Q
A