

CS 331

Computer Vision

09 / 21 / 2026

Instructor: Michael Eckmann

Today's Topics

- Questions? / Comments?
- Solutions to in-class exercises
- Binary Image Morphology operations and uses

Let me solve the In class exercises

Cross-correlation with horizontal Prewitt mask

Determining angle and magnitude of an edge

Cross-correlation with a smoothing mask

Uses of Binary Morphology

- Let me motivate at least one use of binary morphology with this image of just the green hue selected pixels



Uses of Binary Morphology

- Let's look at that a portion of that image zoomed in --- notice “holes” in the leaves



Uses of Binary Morphology

- Binary morphology can close those holes in an automated way



Morphology

- Morphology in general works on binary images (those that truly have pixels of only 1 of 2 possible colors (e.g. black or white))
- We can use a certain morphology operation on the image on the previous slide if we consider the black pixels as black (off/background) and the green pixels as white (on/foreground).
- Besides filling in holes, other morphology operations are also used to smooth jagged edges of regions, remove small regions entirely, and other purposes
- Think of the white pixels as foreground (objects) and the black pixels as background.
- 0=black=off=background (all used interchangeably)
- 1=white=on=foreground (all used interchangeably)
- If the black and white image is actually stored as grayscale (e.g. 0 – 255 values) then white would be 255.

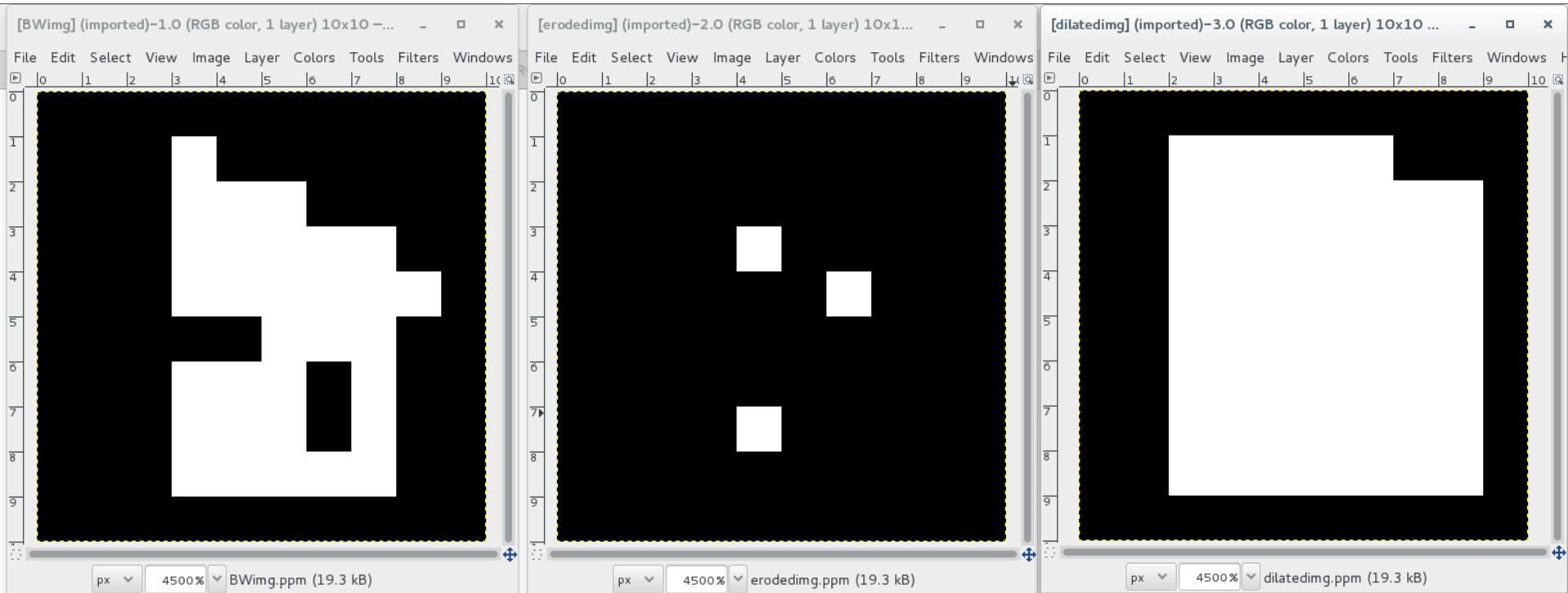
structuring element

- a structuring element is typically a small binary image representing some shape. A structuring element has an origin.
 - e.g. a 3x3 square of all 1-pixels, with center as origin
 - a 5x5 binary image with a diamond shape of 1-pixels, others 0's
- a structuring element is applied to a binary image by hovering the origin pixel over each pixel in the image one at a time
 - only the 1-pixels in the structuring element matter (the 0-pixels of the structuring element are not compared to anything)
 - the structuring element **fits** means that *all* the 1-pixels in the structuring element correspond to 1-pixels in the image
 - the structuring element **hits** means that *at least one* 1-pixel in the structuring element corresponds to a 1-pixel in the image

binary morphology

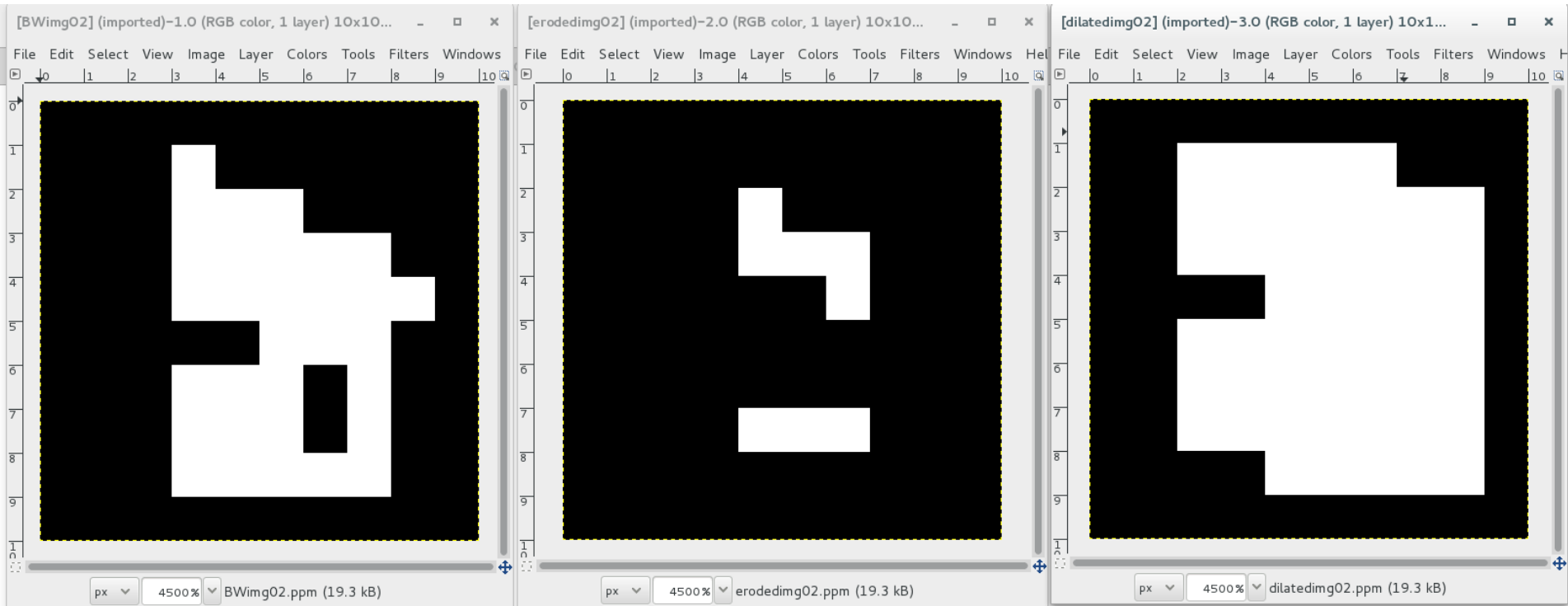
- binary image morphological operations
 - image is **f**, structuring element is **s**, resulting image is **g**
 - **dilation** – increases the size of regions
 - **f** dilated by **s** results in **g** where **g** contains a 1 where **s** hits **f**, 0 otherwise
 - **erosion** – decreases the size of regions
 - **f** eroded by **s** results in **g** where **g** contains a 1 where **s** fits **f**, 0 otherwise

examples



- Left image – original 10x10 w/ white starting in 2nd row, 4th column,
- Middle image – eroded w/ 3x3 Box,
- Right image – dilated w/ 3x3 Box.

examples



- Left image – original 10x10 w/ white starting in 2nd row, 4th column,
- Middle image – eroded w/ 3x3 L shape,
- Right image – dilated w/ 3x3 L shape.

binary morphology

- binary image morphological operations
 - Notice that dilation filled in holes, but the region size grew
 - And erosion got rid of jutting out pixels, but also the region shrunk

binary morphology

- binary image morphological operations
 - image is f , structuring element is s , resulting image is g
 - **closing** – closes up holes within regions
 - f closed by s results in g , which is the result of f dilated by s and then eroded by s
 - Dilate then Erode
 - **opening** – get rid of jutting out portions of regions
 - f opened by s results in g , which is the result of f eroded by s and then dilated by s
 - Erode then Dilate
- Both opening and closing are idempotent
 - this means after one closing of f by s , further closings by s do not change the result
 - same for opening