Lecture notes: Segmentation (continued)

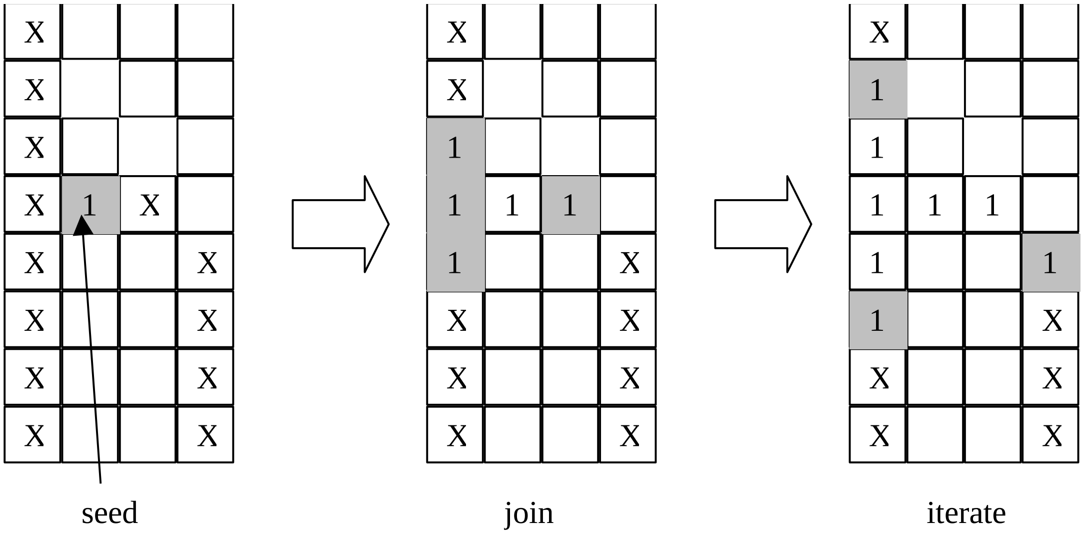
One of the most useful methods for segmentation is based on the concept of “paint-fill”, also called “flood-fill” and “region growing”. The region growing segmentation method works by starting at a seed point and adding neighboring pixels to it that share the same desired properties. The boundary of the region iteratively grows outward, until it reaches pixels that no longer share the desired properties.

The steps can be written as follows:

a) **Find seed.** Scan image for a seed pixel. Example: Looking for oranges? Find the "most orange" pixel that is not labeled yet. A generic approach would be to find the pixel "farthest away" from all pixels that have been labeled so far.

b) **Similarity test(s).** Test neighboring (4- or 8-) pixels for similarity (segmentation predicate) to seed pixel.

c) **Join and iterate.** For each pixel that passes similarity test(s), add to region (mark with same label as seed) and goto (b) (recursively).



X			
X			
X			
X	1	X	
X			X
X			X
X			X
X			X

X			
X			
1			
1	1	1	
1			X
X			X
X			X
X			X

X			
1			
1			
1	1	1	
1			1
1			X
X			X
X			X

In this example, the grey pixels show the "**wavefront**" (boundary of the region as it is growing).

Region growing is like the familiar "**paint-fill**" operation of drawing tools. It starts at a point and flows into neighboring pixels until it reaches a boundary where the similarity test fails.

Posted at the web site is some C-code that implements a recursive implementation of paint-fill, in which the similarity predicate is that the pixel must have the same exact value as the seed. This would be useful for example in creating a segmentation from a thresholded image (as in the above example). The image dishware.ppm is a good candidate.

The C-code also shows a queue-version of paint-fill, which will run much faster and use a lot less memory.

Finally, some C-code shows a "GrowRegion" function and a bit of calling code that uses paint-fill to segment regions of similar intensity (pixels within a given variance).

Note that the similarity tests can use the growing core as the test benchmark, instead of just the seed pixel.

All these version of C-code will be demonstrated and discussed during class.