Lecture 26:

Logical indexing and for loops

Lecturer: Jian Wei Tay

Date: 22 October 2021

Learning objectives

- How logical indexing works
- How to use logical indexing to filter data in a matrix
- How to use logical indexing to update a mask with watershed ridge lines
- Writing a `for` loop
- How to identify objects in a `regionprops` struct

Logical indexing

- In logical indexing, a logical array is used to select elements

- The indexed elements correspond to the true elements of the logical array

Example of logical indexing

```
>> A = [16 5 9 4 2 11];

>> L = [true false false true true false]

>> A(L)

ans =

    16      4      2
```

Practice

- What is the value of L?

$$M = [16\ 5\ 9\ 4\ 2\ 11]$$

$$L = M > 9$$

A. L = 9
B. L = [1 0 0 0 0 1]
C. L = [1 0 1 0 0 1]
D. L = true

Practice

- What is value of `M(L)`?

```
M = [16 5 9 4 2 11]
L = M > 9
M(L)
```

A. `M(L) = [16 11]`

B. `M(L) = 5`

C. `M(L) = [16 9 11]`

D. `M(L) = [5 9 4 2]`

Applications of logical indexing

- Filtering data from a matrix

```
M = [16 5 9 4 2 11];
L = M > 9;
M(L) = [];
```

Note: Assigning a matrix element to an empty matrix [ ] deletes the elements

```
>> M
M =
        5       9       4       2
```

Practice

- Given the matrix B below, which command will remove any elements with a value of 12 or higher?

    B = [10 15 8 2 11 25 31]

    ```
    A. B(B > 12) = 0
    B. B(B > 12) = []
    C. B(B >= 12) = []
    D. B(12) = []
    ```
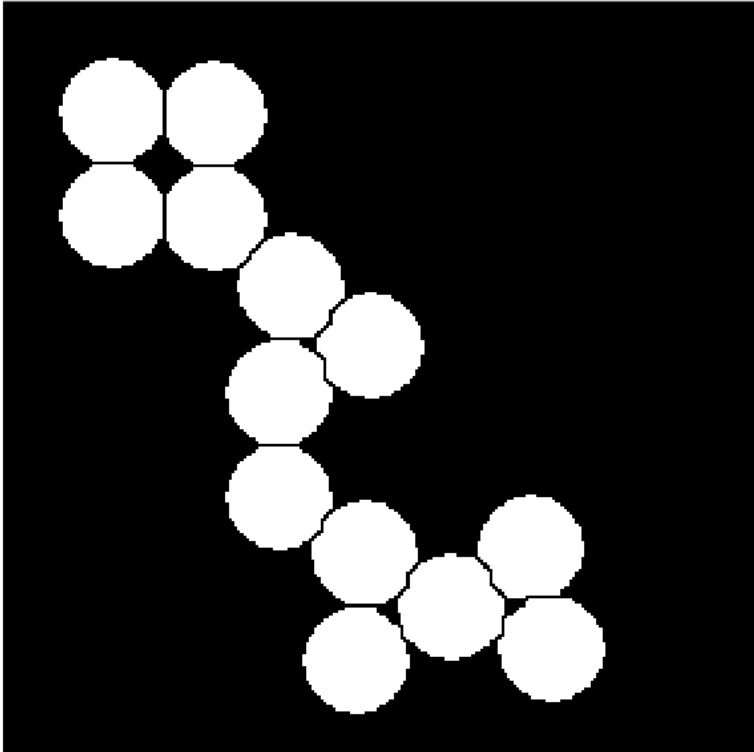
Applications of logical indexing

- Adding watershed ridge lines back to the mask

```
M = imread('circles.png');
dd = -bwdist(~M);
dd(~M) = -Inf;
dd2 = imhmin(dd, 2);
L = watershed(dd2);
M(L == 0) = false;
```
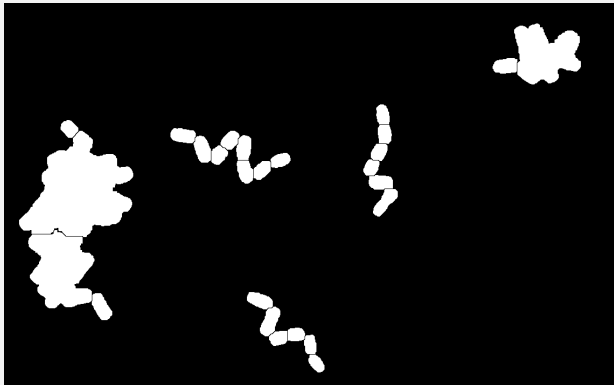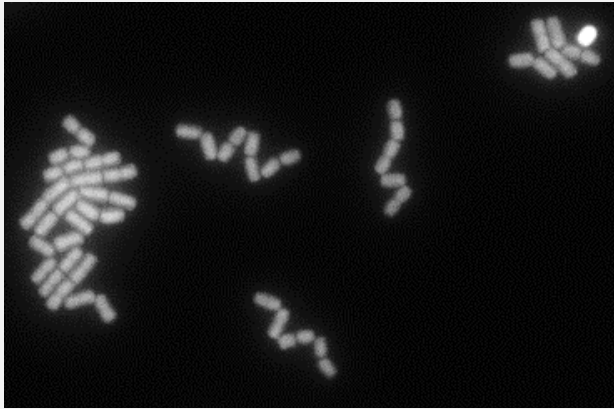
Applications of logical indexing

- Updating the mask allows you to make additional modifications

- E.g.: Using `bwareaopen` to remove small areas or additional morphological operations

Questions?

Coding practice

- Read in the image
  `L26_mask.tif`

- Measure the length of single cells only (exclude the objects which were undersegmented)
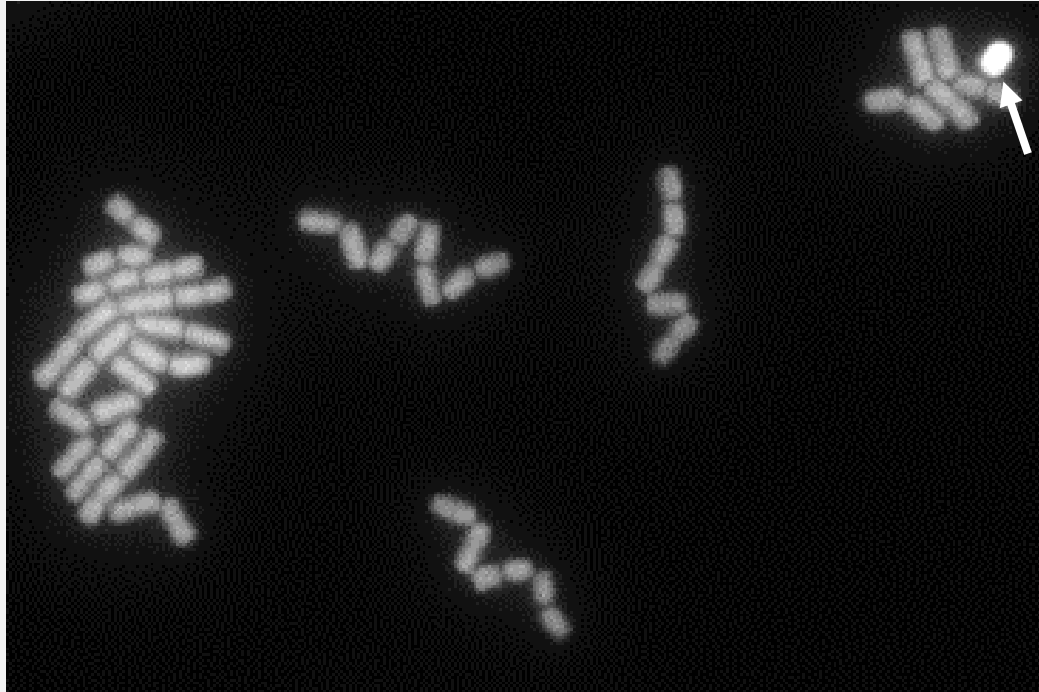
Tips

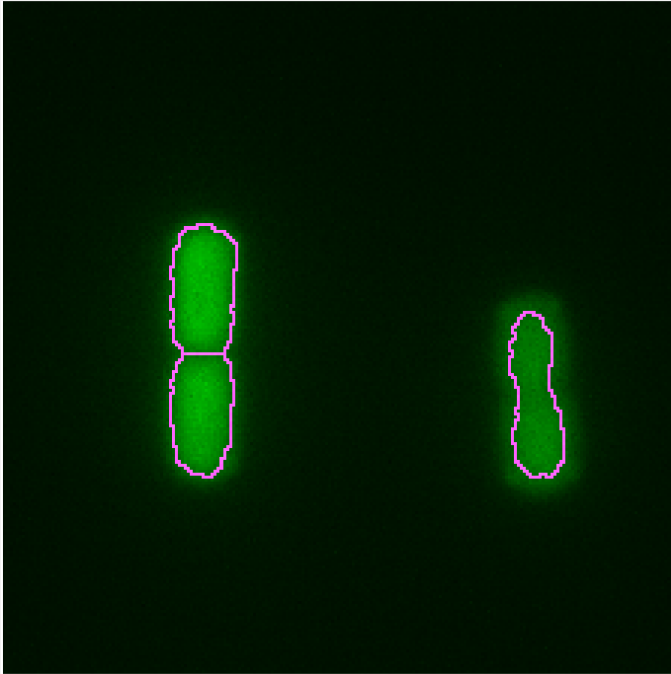- If you have a lot of segmented data, you can plot a histogram of the data to choose a good cutoff

```
celldata = regionprops(mask, 'Area')
areas = cat(1, celldata.Area)
histogram(areas)
```

- What are some pitfalls to be careful of?

Write code to exclude the bright cell

Discussion

- Object brightness can affect segmentation results

- In some approaches, intensity is used to reject these – e.g., by keeping only objects that are in the middle 75 percentile of brightness

- Is this good practice? Are there any problems with doing so?

Questions?

For loops

- A for loop repeats lines of code a set number of times

Basic syntax

```
for index = vector


end
```

Basic syntax

```
for index = vector

 keywords

end
```

Basic syntax

```
for index = vector
```
Name of index variable

```
end
```

Basic syntax

```
for index = vector
```

Values of index

```
end
```

The loop will repeat as many times as the number of elements in the index vector

Example for loop

```
for ii = [1 2 3 4 5]
    disp(ii)
end
```

Note: `disp` displays the value of the variable

- The index variable `ii` changes each time the loop runs (iteration)
  - First iteration: `ii = 1`
  - Second iteration: `ii = 2`
  - …
  - Fifth iteration: `ii = 5`

Note: This loop will run 5 times

Practice

- How many times will the following loop run?

```
for jj = 1:15
    x = jj * 2
end
```

A. 15 times

B. 16 times

C. Once

D. I don't know

Practice

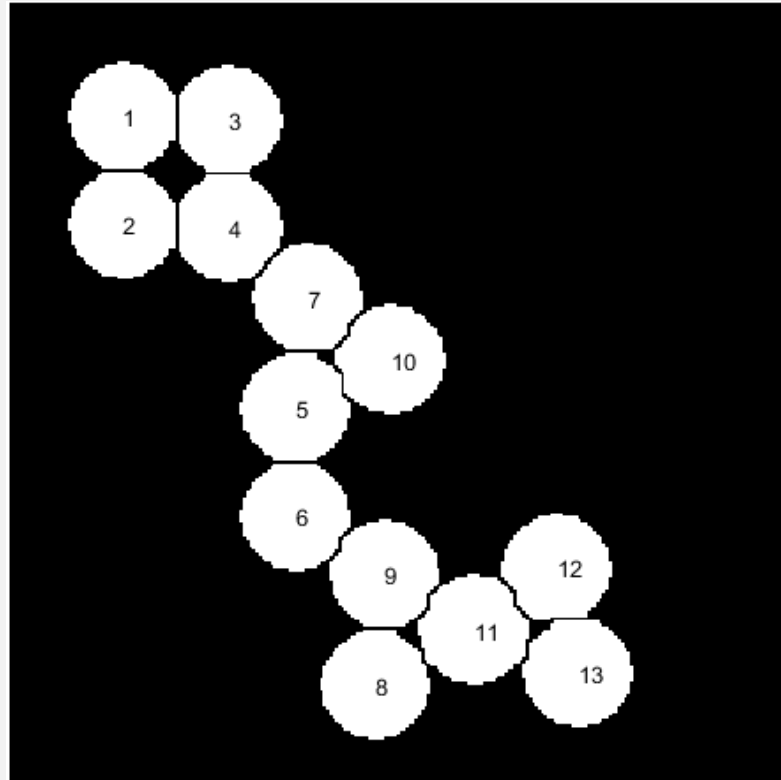- What is the value of `idx` in the fourth iteration in the following loop?

```
for idx = [10 1 42 2 5 20]
    disp(idx)
end
```

A. 10

B. 42

C. 2

D. 20

Example using a for loop

- `regionprops` identifies objects based on the order that it encounters it in the image
- So to identify which object corresponds to which element of the `regionprops` output, we can plot numbers using the centroid

Example

Example

- Using the output of the watershed algorithm on `circles.png`

```
data = regionprops(mask, 'Centroid');
imshow(mask)
hold on
for ii = 1:numel(data)
    text(data(ii).Centroid(1), data(ii).Centroid(2),
int2str(ii))
end
hold off
```

> **Note:** `numel` returns the number of elements in a variable (see Lecture 16)

Some notes on the code

- The centroid data is returned as a 1x2 vector containing the coordinates [x, y]

- So `data(ii).Centroid(1)` returns the x-coordinate of object `ii`

- The function `int2str(M)` rounds the elements of `M` to integer, then converts the number into a string

Example
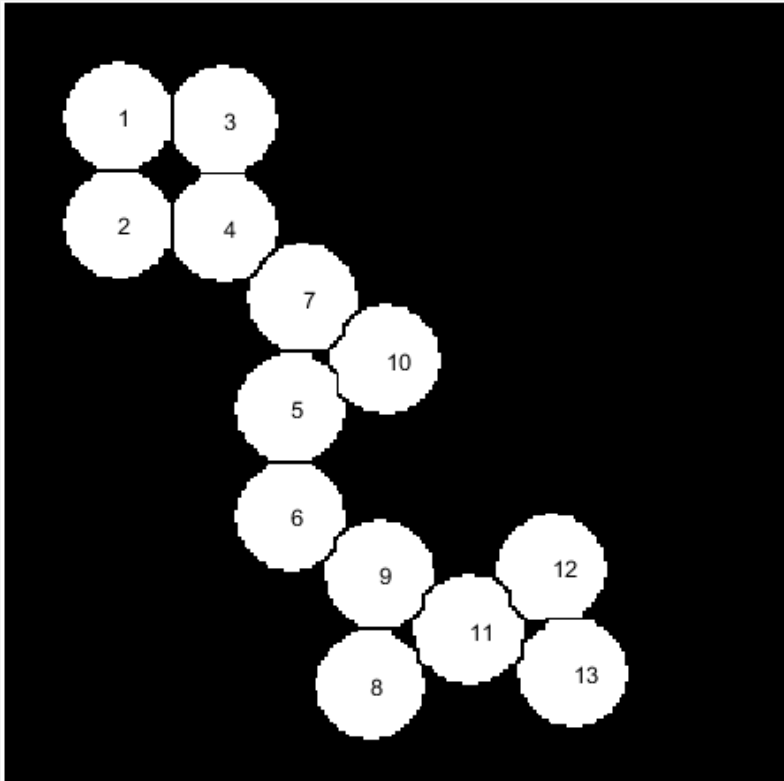
- Making a plot like this is useful when trying to identify a single object

- `data(11).Area` will return the area of the circle labeled 11 in the image on the left

Questions?