Enabling Technologies for Sports (5XSF0)

Image Segmentation

Chrysi Papalazarou, Sveta Zinger

This chapter: Segmentation

* Subdivide images into regions or objects
* What a "region" or object is, is usually application-dependent
* The problem is often the absence of a ground truth
* Also: when to stop segmenting?

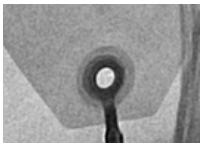
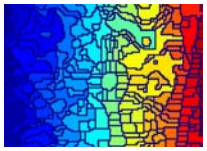


Image of ECG pad



Oversegmentation

Undersegmentation

"Good" segmentation

Segmentation: Basic approaches

- **"Top-down": based on *discontinuity***
  - Pixels belonging to different objects will have different gray values
  - Partition an image based e.g. on edges
- **"Bottom-up": based on *similarity***
  - Pixels belonging to the same object will have similar gray values
  - Partition an image based on similarity criterion

What we will see today

Implementations, using MATLAB, of the following techniques:

- **Point and line detection for segmentation**
- **Thresholding**
- **Region-based segmentation**
- **Watershed transform**

Point detection

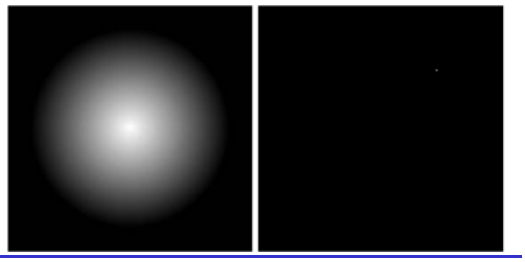
* Look for a point that is different from its neighborhood
* Apply an isolating mask to calculate:

$$R = w_1 z_1 + w_2 z_2 + \ldots + w_9 z_9$$

* A point is detected at the center of the mask if

$|R| \geq T$ where T is a threshold

−1	−1	−1
−1	8	−1
−1	−1	−1



Line detection

* Masks responding to lines of different orientations:

−1	−1	−1
2	2	2
−1	−1	−1

Horizontal

−1	−1	2
−1	2	−1
2	−1	−1

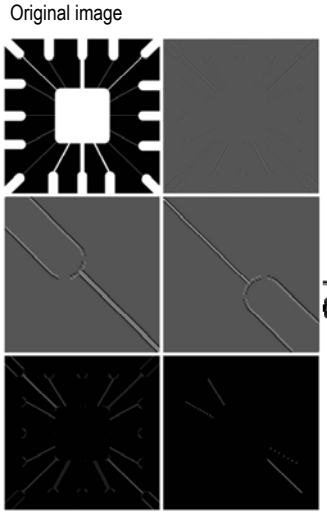
+45°

−1	2	−1
−1	2	−1
−1	2	−1

Vertical

2	−1	−1
−1	2	−1
−1	−1	2

−45°



Point and line detection in MATLAB

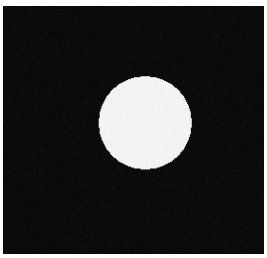
```
g = abs( imfilter (double(f), w ) ) >=T;
```

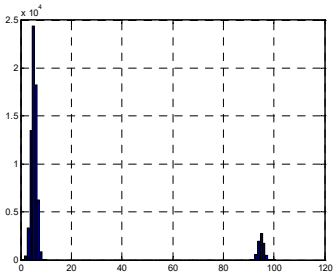
- Binary image giving point locations
- 2D filtering
- Convert to double!
- Input image
- Mask
- Threshold

* How to set T?
* In a first instance, by examining the values of the feature image
* More on thresholding later

Thresholding

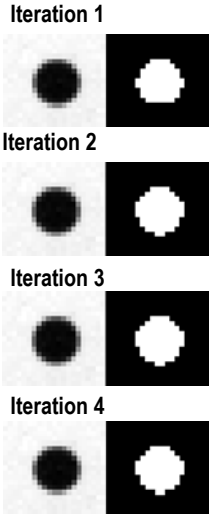
* Simple and computationally efficient
* Threshold selection uses intensity information → histogram
* Example: bimodal histogram





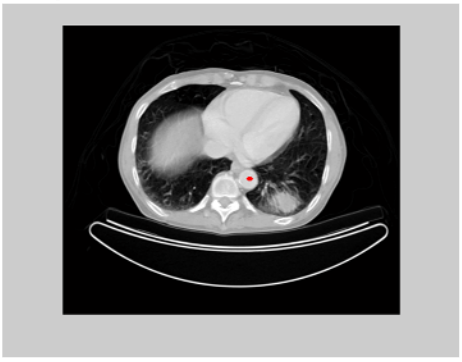
Optimal Thresholding

* What value of T will give us the best segmentation?
* Gonzalez and Woods:
  - Initialize T (e.g. halfway between min and max)
  - Iteratively set T=0.5*(μ1+μ2), where μ1, μ2 the mean values of pixels with value larger or smaller than T, respectively
  - Until T converges



Region Growing

* Start from a seed point, selected manually or automatically
  - Several seed points are also possible
* Include new neighboring points in the region based on some similarity criterion



* Image from http://www.mathworks.com/matlabcentral/fileexchange/19084

Region Growing in MATLAB

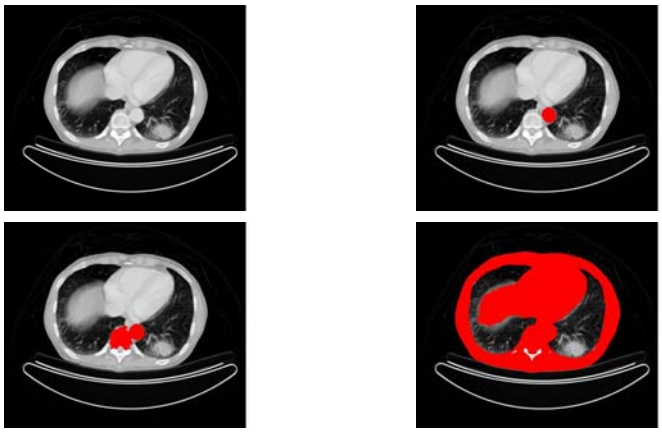
* **Function (from textbook, *not* Image Processing Toolbox):**

```
[g, NR, SI, TI] = regiongrow(f, S, T);
```

* **Input:**
  - **`f :`** image
  - **`S :`** seed points (scalar→all points with this value, array → 1's for seed points and 0's for non-seed points)
  - **`T :`** threshold (scalar → global threshold, array → local threshold at each point)
* **Output:**
  - **`g:`** segmentation result (different labels per region)
  - **`NR:`** # of regions
  - **`SI:`** final seed image
  - **`TI:`** image of pixels that passed threshold test

Region Growing in MATLAB

* **Play with seed points & thresholds**



References

- Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, "Digital Image Processing Using Matlab", Pearson Education, 2004
  - Chapter 10

Gonzalez' method for thresholding

```
function [bw, t]=threshold_gw(b);
%  b bimodal image (dark on bright or bright on dark)
b=double(b);
t=0.5*(min(b(:))+max(b(:)));
told=0;
c=1;
 bw(:,:,c)=(b<t);
while abs((told-t)/told)>0.01
    c=c+1;
    m1 = mean(b(b>t));
    m2 = mean(b(b<t));
    told=t;
    t=0.5*(m1+m2)
    bw(:,:,c)=(b<t);
end
```