IMAG/e

TU/e Technische Universiteit Eindhoven University of Technology

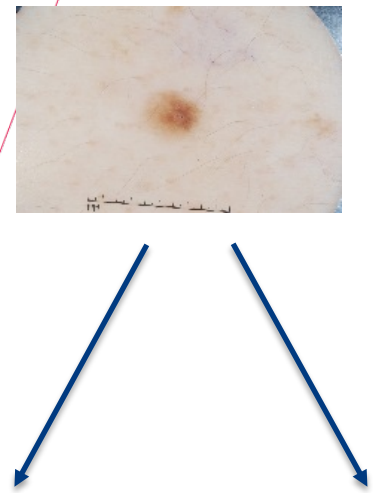
Image Analysis Project 8QA01

Part 2 – Measuring features

Dr. Veronika Cheplygina

Goal: measure features in the image

- Existing or newly designed features
- A single number that encodes some property
  - Category (e.g. color)
  - Continuous number (e.g. size of spot)



Q1: Age	…	Q7: Color	New feature
20		1	1.7
25		1	2.5
40		2	1.3
70		3	0.1

https://challenge.kitware.com/#challenge/n/ISIC_2017%3A_Skin_Lesion_Analysis_Towards_Melanoma_Detection

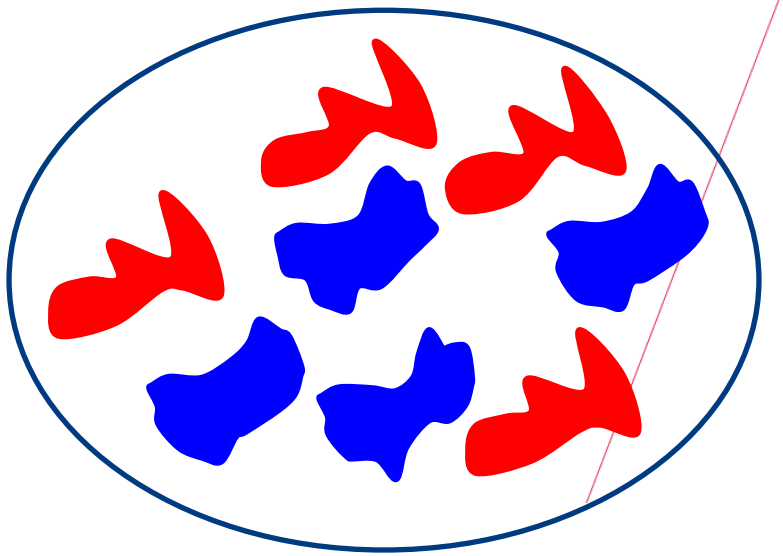
Features for skin cancer

- Experts use “ABCDE” features to recognize melanoma / skin cancer
- A – Asymmetry
- B - Border
- etc

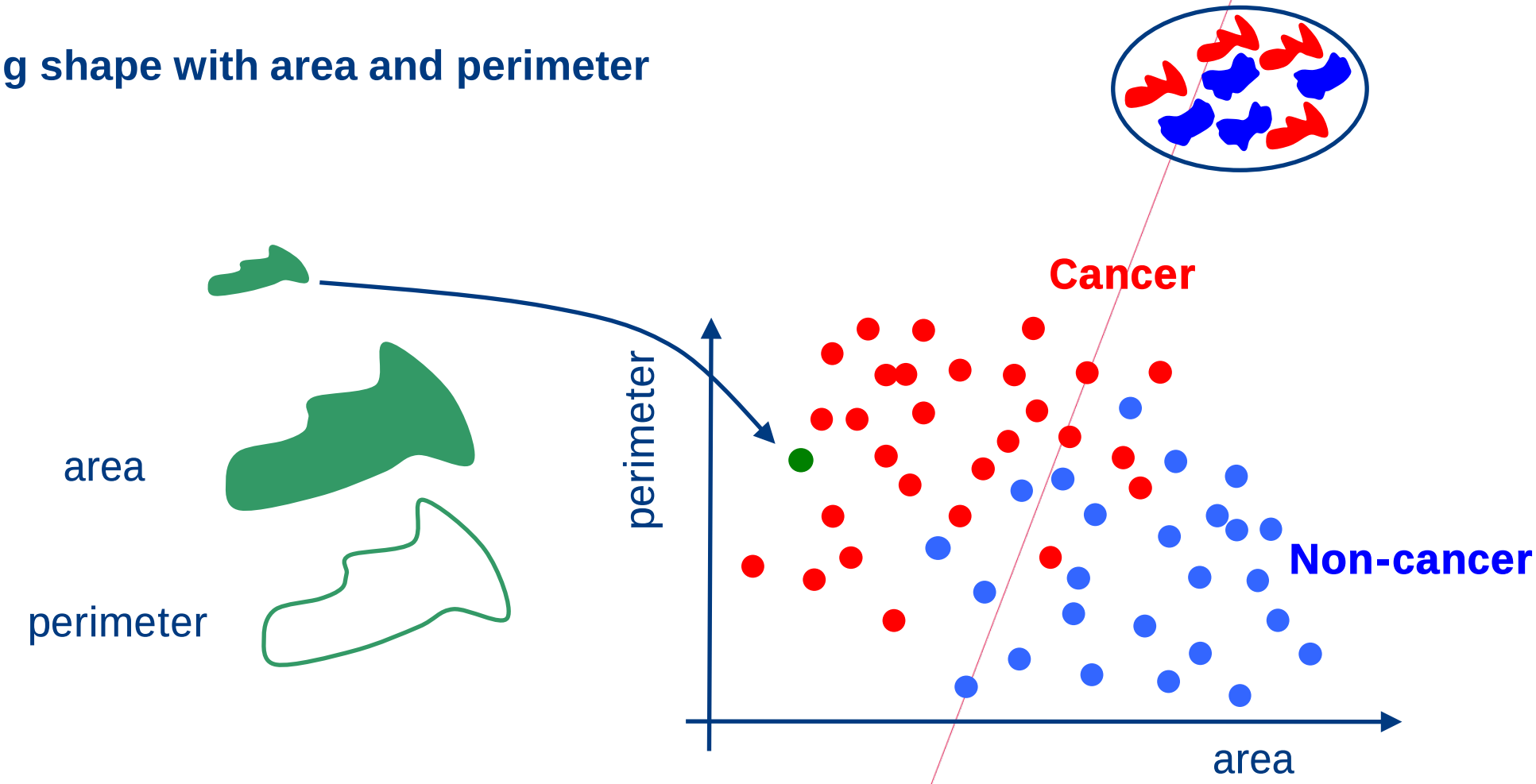
http://www.skincancer.org/skin-cancer-information/melanoma/melanoma-warning-signs-and-images/do-you-know-your-abcdes#panel1-2

Example: red spots = cancer and blue spots = non-cancer

Blue spots have more smooth shapes than the red → how to measure this?



Measuring shape with area and perimeter

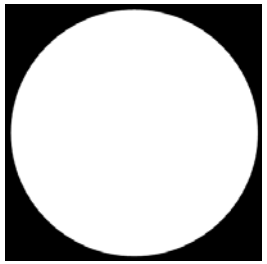


Measuring shape with area and perimeter

- The relationship of area and perimeter tells us about the shape
- We can combine two features into a single number
- Compactness $c = \frac{l^2}{4\pi A}$ (l = length i.e. perimeter, A = area)

Measuring shape with area and perimeter

- Compactness $c = \frac{l^2}{4\pi A}$

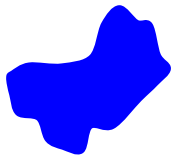


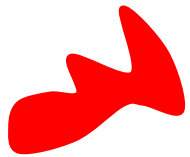
1.3

7.2

79.1

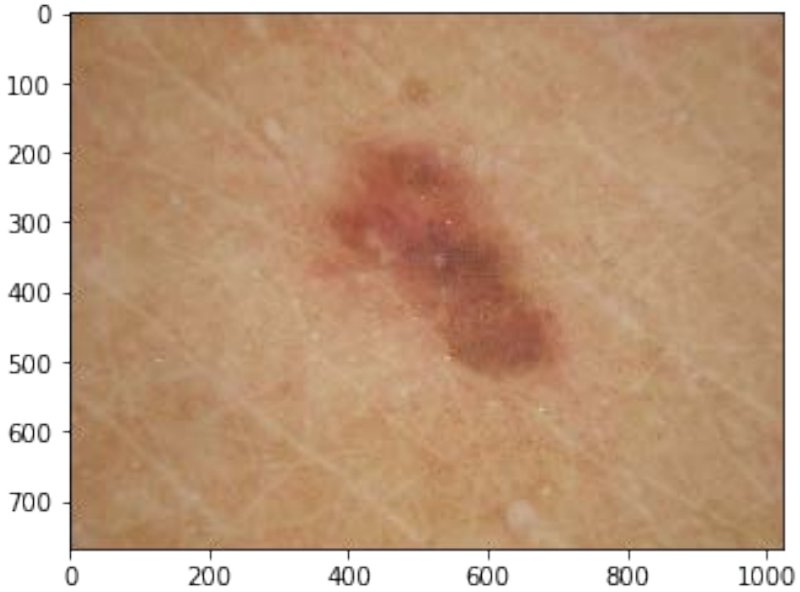
743.2

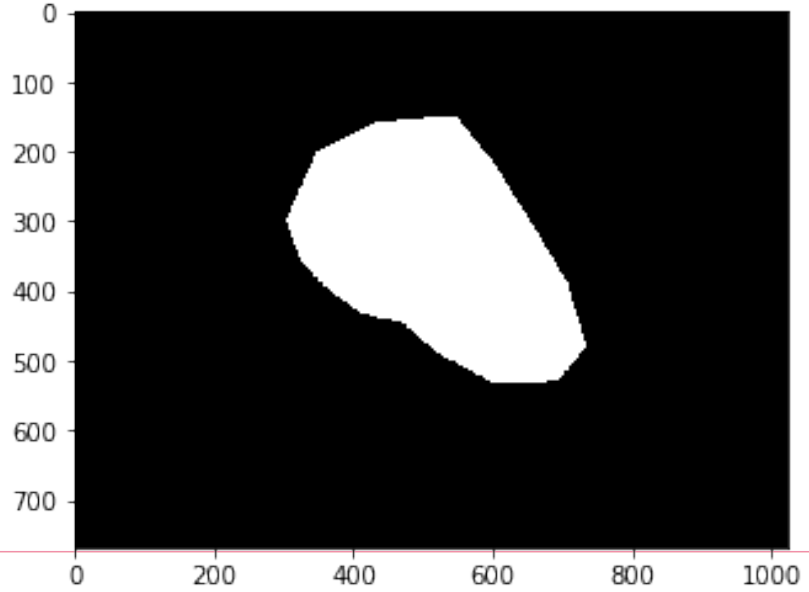




Measuring shape with area and perimeter - Steps

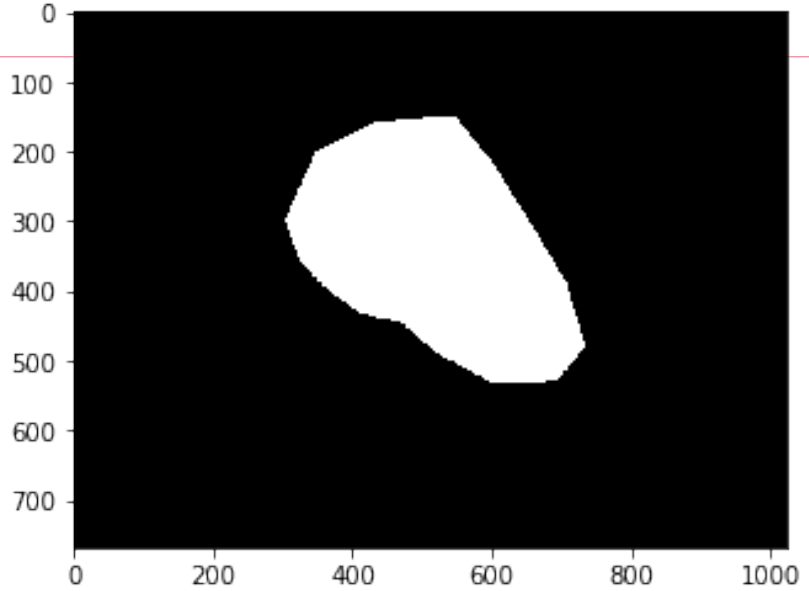
- Need a binary image or **mask** which indicates which pixel is inside (value = 1) or outside (value = 0) the shape
- Area = sum of all pixel values in the mask

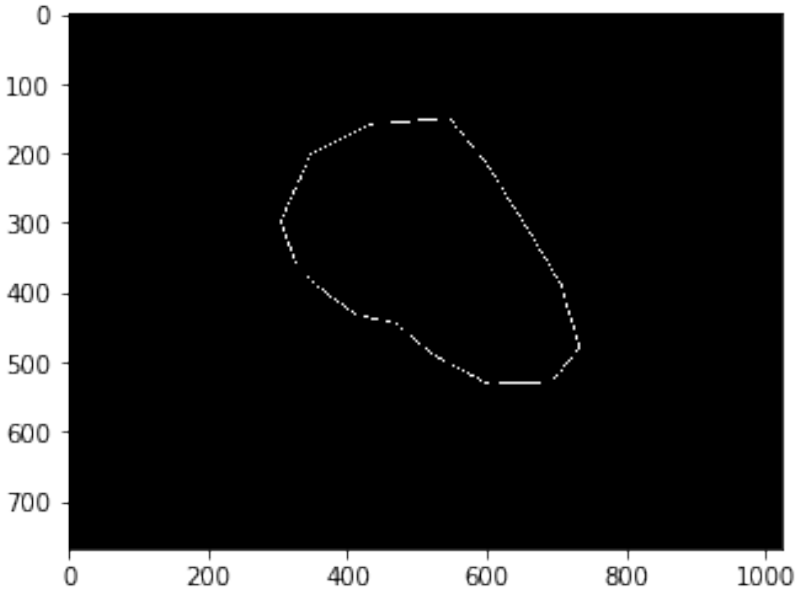




Measuring shape with area and perimeter - Steps

- Perimeter = sum of pixels on the border
  - Resize the image by 1-2 pixels
  - Subtract the smaller image from the larger image
  - Only border pixels will have value 1
  - Sum the pixel values



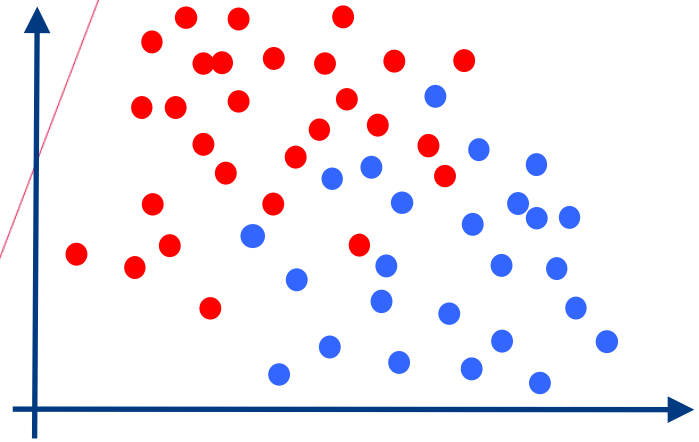


Measuring shape with area and perimeter - Steps

```
def measureAreaPerimeter(maskImage):

    #Measure area: the sum of all white pixels in the mask image
    area = np.sum(maskImage)

    #Measure perimeter: first find which pixels belong to the perimeter.
    struct_el = morphology.disk(1)
    maskEroded = morphology.binary_erosion(maskImage, struct_el)
    perimeterImage = maskImage - maskEroded

    #Now we have the perimeter image, the sum of all white pixels in it
    perimeter = np.sum(perimeterImage)

    return area, perimeter
```

What is a good feature?

- Variation of values for different images
  - “Value at location (0,0) in mask image” will probably always be zero
- Not too correlated with other features
  - “Value at (x,y)” and “value at (x+1,y)” will be similar



What is a good feature?

- The feature provides (some) information about the category of the image
- Combining features can help to separate the categories better
- Use scatterplots or “rain cloud plots” to see if combinations of your features create meaningful patterns