Advanced Image Analysis

a.a. 2017-2018, 2nd semester

Project

Mass Segmentation in Digital Mammograms

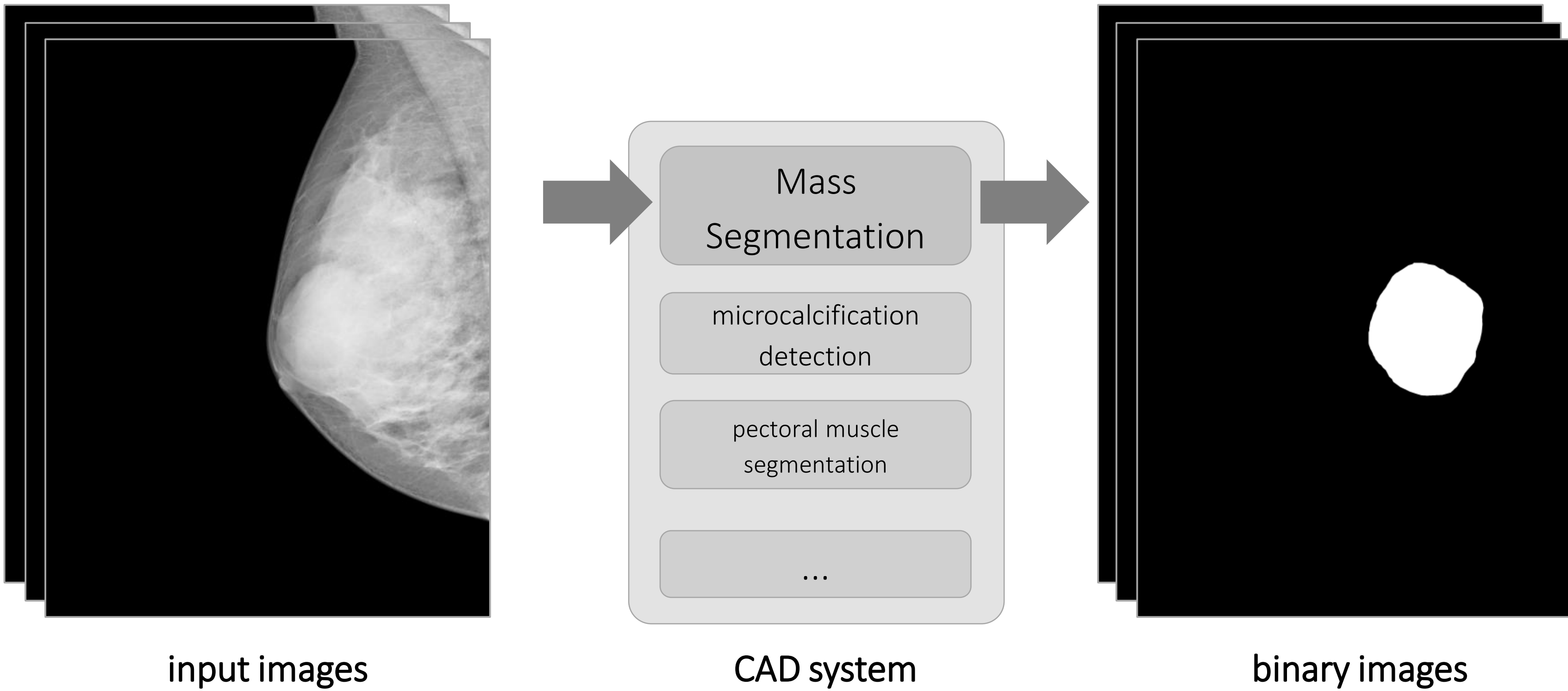
Alessandro Bria
Contract Professor
and Post-doc Researcher
email: a.bria@unicas.it

Motivations

- **X-ray mammography** is a widely used method to screen women for early detection of **breast cancer**
- **Computer Aided Diagnosis** (CAD) helps radiologists in interpreting screening mammograms
- the two most important lesions that may be present on a mammogram are **microcalcifications** and **masses**
  - CAD often consists first in **detecting** the lesions and then **classifying** them into benign / malignant

Goal

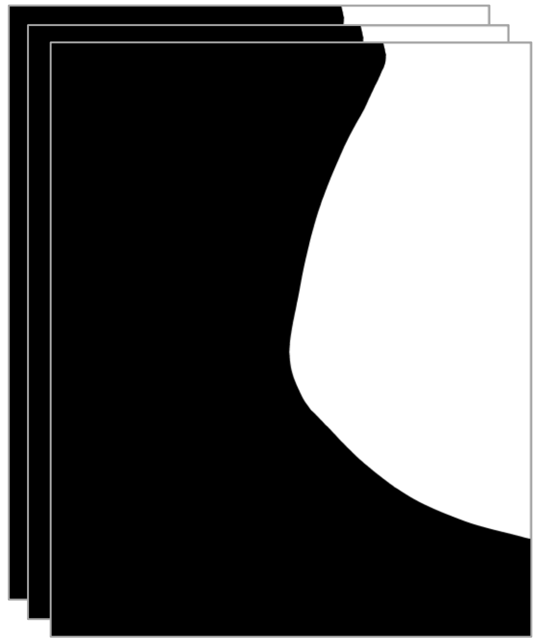
- implement a reusable module for *automated* **Mass Segmentation**
  - a must-have module in most **CAD** systems



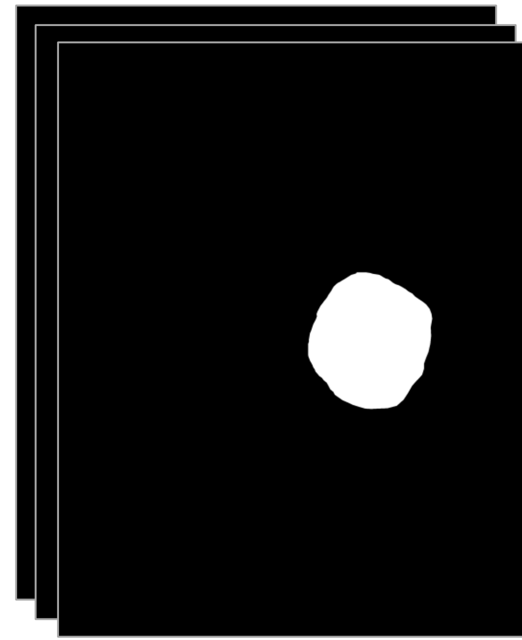
Materials

- **INBREAST** dataset (410 images) containing:

mammograms (16-bit)
`/dataset/images`



breast-air masks
`/dataset/masks`

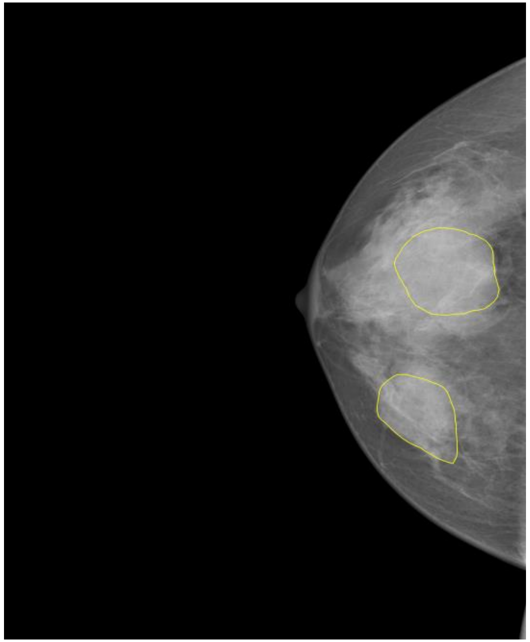


manual annotations
`/dataset/groundtruths`

Materials

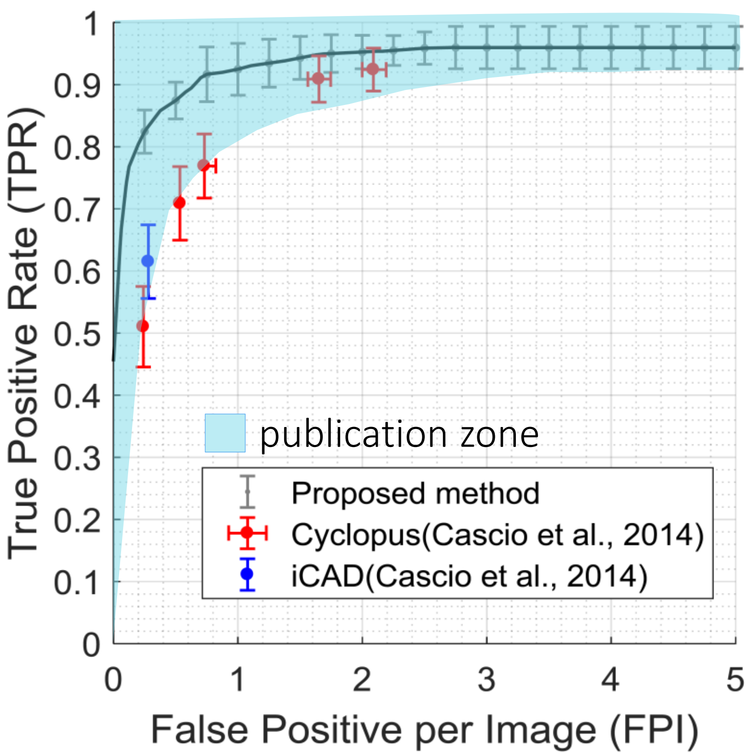
- warning: among the 410 images, only 107 contain masses (positive images)
  - there are only 107 manual annotations in the `/dataset/groundtruths` folder
  - see also the 107 overlayed annotations in the `/dataset/overlay` folder





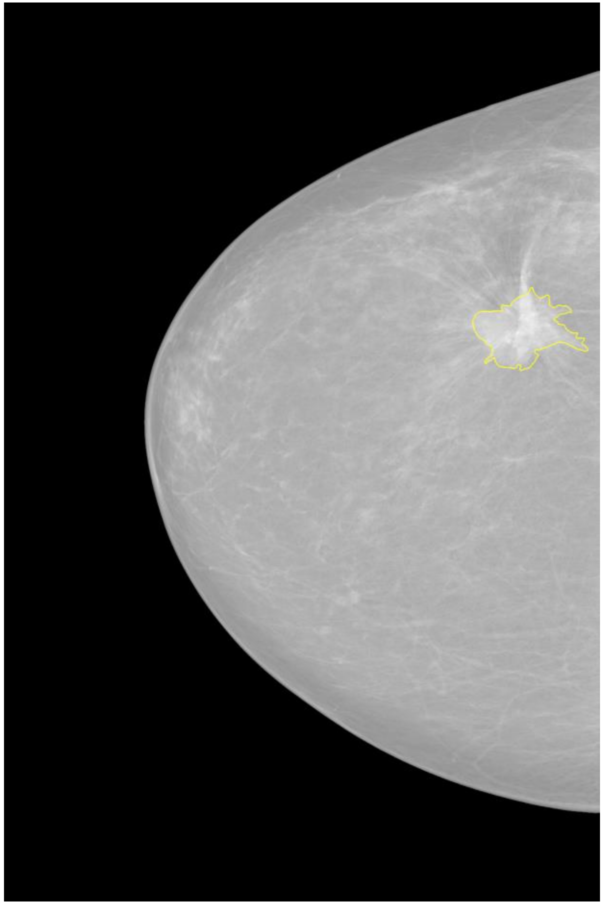
Performance evaluation

- **FROC** curve
  - True Positive Rate (**TPR**) vs. False Positive Rate per Image (**FPpI**)
    - *true positive* = a CAD finding that **matches** with a groundtruth finding
    - *false positive* = a CAD finding that *does not* **match** with any of the groundtruth findings
      - true positives should be evaluated on the 107 positive images, whereas false positives on the remaining 303 images
  - a match is found when the Dice Similarity Index between the two findings is above 0.2:
    - $match(A,B)$ if $DSI(A,B) = \frac{2|A \cap B|}{|A|+|B|} \geq 0.2$
  - the curve is obtained by varying the decision threshold of the CAD system



Coefficient of difficulty

- base coefficient of difficulty
  - $c = 0.8$
  - yes, this is a difficult project!
    - e.g. see the mass on the right
- no bonuses



Constraints

- implementation in **any language**/framework
  - C++/OpenCV suggested for extracting the mass candidates / features (but not mandatory)
- **reusable** module
  - the code *must be* commented
- **same parameters** for all the images
  - unless parameters are automatically-adjusted on each image
- **machine learning**
  - *highly recommended*
  - use the *first half* of the dataset for **testing**, and the *second half* for **training**
    - then switch testing with training, and merge the results (2-fold cross-validation)

Code snippets

- how to calculate the **Dice Similarity Index**?
  - adapt and modify `/code-snippets/compute-segmentation-accuracy.cpp`
- how to **load** all images within a **folder** w/o knowing their file names?
  - see `/code-snippets/load-all-images-within-folder.cpp`

Hints

- top-down approach
  1. **preprocessing**
     - e.g. contrast enhancement with one of the techniques learnt from *Image Processing*
  2. **mass candidate extraction**
     - e.g. oversegmentation with one of the techniques learnt from *AIA*
  3. **feature extraction**
     - extract a set of meaningful features from each mass candidate (features learnt from *AIA* + others)
  4. **classification**
     - use machine learning (SVM, Boosting, Random Forest, ... others learnt from *Pattern Recognition*) to classify candidates into masses and nonmasses