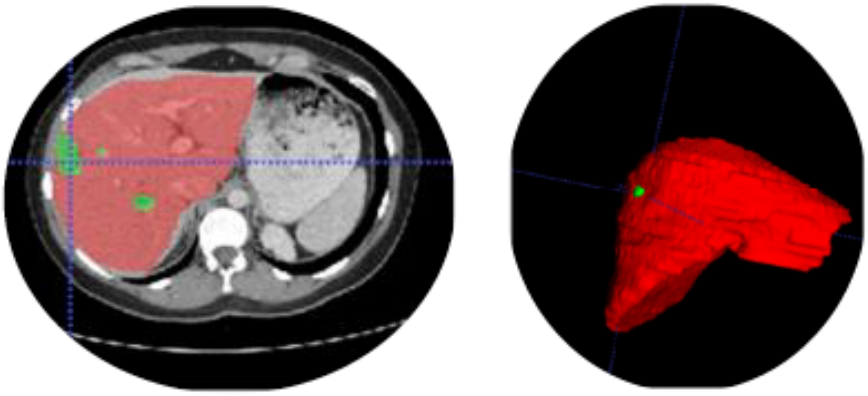
Liver Project

Advanced Medical Image Processing

March 2019

1. Introduction



A fully automatic technique for segmenting the liver and localizing its unhealthy tissues is essential in many clinical applications, such as pathological diagnosis, surgical planning, and postoperative assessment. However, it is still a very challenging task due to the complex background, fuzzy boundary, and various appearance of both liver and liver lesions.

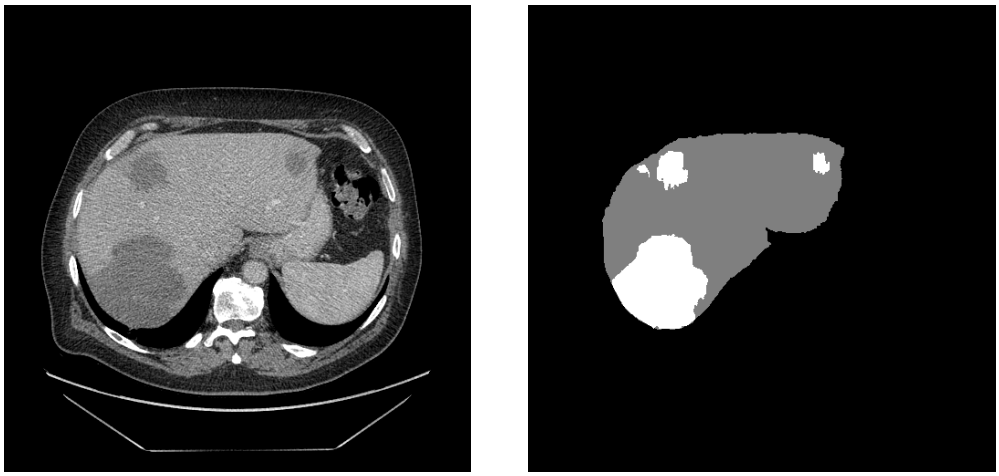
2. Tasks

In this project you'll develop automatic algorithms to segment liver and liver lesions in abdominal CT scans. To achieve this goal you'll utilize the power of deep learning algorithms.

The project consists of the following tasks:

1. Liver segmentation
2. Lesions segmentation

3. Data



The training data set consists of 11 CT scans:

- 'Data' directory contains CT images converted to 'png' format
- 'Segmentation' directory contains segmentation masks:
  - Pixels with value 127 indicate liver
  - Pixels with value 255 indicate liver lesion

You can download the data from the following link: data link.

Using of another data to train the model is not allowed.

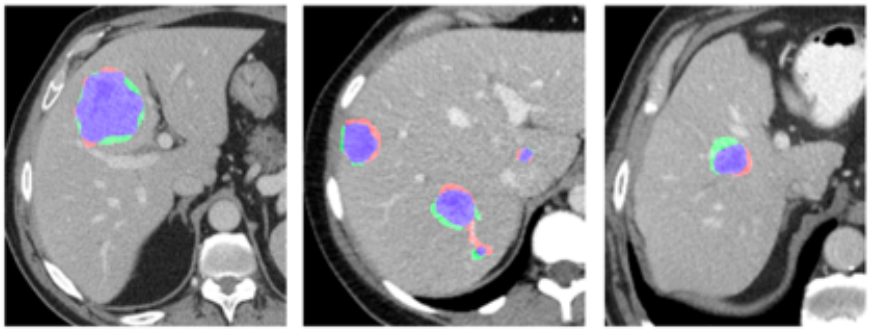
4. Evaluation metrics

The following metrics will be used to evaluate the segmentation accuracy:

- Dice similarity coefficient (Dice):

$$Dice = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{1}$$

  TP, FP, and FN denote the number of true positive, false positive, and false negative pixels respectively. Dice computes a normalized overlap value between the produced and ground truth segmentation.



- Precision (positive predictive rate):

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

  Precision expresses the proportion between the true segmented pixels and all pixels the model associates with liver / lesions.

- Recall (sensitivity):

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

  Recall expresses the proportion between the true segmented pixels and all liver / lesions pixels.

5. Implementation

You will implement algorithms using Python with one the deep learning packages (Tensorflow, PyTorch, Keras etc)

6. Important Dates

- 14.03.2019: release of training data
- 26.05.2019: release of test data

- 2.06.2019: submission deadline
- 6.06.2019: short projects presentation (about 10 minutes per project)
- 13.06.2019: best projects presentation

7. Submission Details

You will submit the 'zip' file with:

- Report in 'pdf' format that shortly summarizes the algorithm details (maximum 5 pages), it should not include the code, but only description of methods and experiments
- Directory with the name 'sources' that contains source files
- Directory with the name 'testseg' that contains segmentation masks of the test data set

Please be accurate with the specified directories names.

The 'zip' file should not include training data set or test CT images.

The grade of the project will be calculated according to the quality of the test set results, breadth and depth of the work and originality.

Submission is in pairs.

Mail for questions: eytankats@mail.tau.ac.il