UNIVERSITÀ
DI TRENTO

Lung and Liver Segmentation in Medical Images

Signal, Image and Video

Università degli Studi di Trento,
Trento, Italy

Author: Alberto Vendramini

Date Last Edited: February 5, 2025

Contents

1 Abstract

Organ segmentation is the process of identifying and delineating specific organs within medical images, typically acquired through modalities such as CT, MRI, or ultrasound. It is a crucial task in medical imaging for applications such as disease diagnosis, treatment planning, and surgical navigation.

Segmentation methods can be categorized into traditional techniques, such as thresholding, region growth, and edge detection, and modern deep learning-based approaches, such as convolutional neural networks. Although traditional methods rely heavily on predefined rules and parameters, deep learning models learn features directly from labeled data, often achieving higher accuracy and generalization.

In this work, we will focus on the segmentation of two organs: the lungs and the liver.

Lung segmentation is generally considered less challenging due to the lungs' distinct anatomical features and the noticeable contrast between lung tissue and surrounding structures, which makes boundary delineation more straightforward.

In contrast, liver segmentation presents significant challenges. These include the liver's similar intensity values to surrounding tissues, motion artifacts caused by breathing, and the variability in liver shape and size among different patients. These factors make accurate segmentation more complex and demand advanced techniques to achieve reliable results.

2 Datasets

In this work we utilized medical images (CT scans) of the abdomen area. These type of files are presented in the formats: NIfTI (Neuroimaging Informatics Technology Initiative) with extension *.nii*, NIfTI Compressed with extension *.nii.gz*, DICOM (Digital Imaging and COmmunications in Medicine) with extension *.dicom* or *.dcm*. I've used four different datasets: CT-ORG, MSD, FLARE22 and AbdomenCT-1K.

2.1 CT-ORG

This dataset consists of 140 CT scans, each with five organs labeled in 3D: lung, bones, liver, kidneys and bladder [8]. In this work this dataset is used for Lung evaluation since it is the only dataset with true labels for each organ. Data can befound clicking → here

2.2 Medical Segmentation Decathlon

The Medical Segmentation Decathlon (MSD) is a biomedical image analysis challenge where algorithms compete across multiple tasks and modalities to test the hypothesis that methods performing well on various tasks will generalize effectively to new, unseen tasks [1]. In this challenge, datasets were provided with segmentation of various organs (although diseases present in the data will be ignored in our case). This is the dataset I used the most and the one I used to evaluate the selected algorithms. Task06-Lung and Task03-Liver folders from the Google Drive repository can be found clicking → here

2.3 Former datasets

1. FLARE 2022 [7]. The dataset included 2300 CT Scans. It was eventually abandoned because other options proved to be better. Data can be found clicking → here

2. AbdomenCT-1K [6]. It is a large and diverse abdominal CT organ segmentation dataset with over 1000 CT scans presenting diseases. Replaced because it is easier to work with full CT Scans than with DICOM slices. Data can be found clicking here → Part 1, Part 2, Part 3

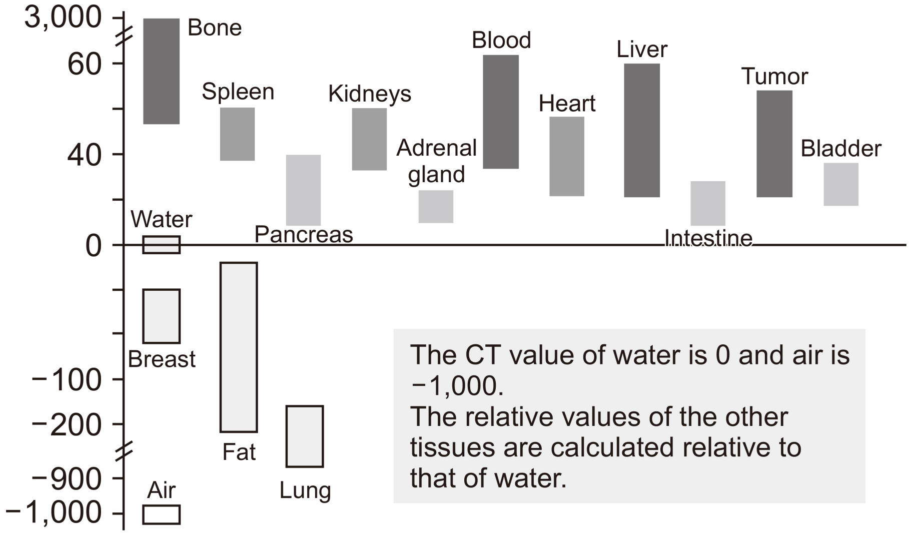
3 Lung Segmentation

Accurate lung segmentation is essential for applications like disease diagnosis (e.g., detecting pneumonia, tumors, or COVID-19 effects), treatment planning, and monitoring disease progression.

3.1 Description

This chapter addresses the problem of Lung Segmentation and proposes an algorithm based solely on image processing techniques. Lung segmentation is generally considered easier compared to other organs due to the distinct anatomical features and the high contrast between the lungs and surrounding tissues in imaging modalities like CT and MRI. The lungs are filled with air, resulting in a well-defined boundary and a unique intensity profile. Additionally, their consistent location and shape across patients make it simpler for algorithms to identify and segment them accurately.

Their Hounsfield Unit (HU) values are significantly different from those of other tissues and organs. The lungs are predominantly filled with air, giving them a characteristic HU range of approximately -600 to -900, which is far from the ranges of most other anatomical structures [4].



The CT value of water is 0 and air is −1,000.
The relative values of the other tissues are calculated relative to that of water.

Figure 3.1: Organs and tissues compared by HU [2].

This part of the project was implemented using scikit-image, SciPy, Mayavi, and NumPy for image processing, analysis, and 3D visualization.

3.2 Algorithm

The fundamentals step in the proposed algorithm are:

1. **Select voxels in HU Lung range**: Discard surely non-lung voxels using information given by HU
2. **Clean obtained mask**: apply common techniques i.e. removing components that touch borders (Clean borders), remove noise and fill gaps (Closing and Opening)
3. **Remove components whose area is under a certain constant**: we do this specifically to remove trachea, since it usually connects both the lungs and forms a single connected component in 3D (We want lungs to be separated)
4. **Find connected components in 3D**
5. **Filter connected components by their centroid**: in some CTs the patient table is showing and its area is similar to the one of the lungs, so we need to delete components that are not near the "center".
6. **Select the two biggest 3D components by volume**: after this step, if nothing went wrong, we should have two objects, left and right lungs.
7. **Fill the holes in the lungs for each slice**

3.3 Results

In this section we analyze the results of the algorithm. As evaluation metric the Dice Score is utilized [5]. The range of the dice score is $[0, 1]$ where 0 indicates no overlap, and 1 indicates complete overlap. Particularly in this case we use 3D Dice score.

The dataset used is CT-ORG, presented in Chapter 1. Since there is no training part, the whole dataset (130 files) was used for evaluation by filtering the CT where the lungs were not connected or not completely visible. Evaluation was conducted on whole 3D CT, not slices. Some CT scans were removed because they either lacked lungs or contained segmentation of only one lung. The Average Dice Score found was : $Dice_{avg} \approx 0.9355$

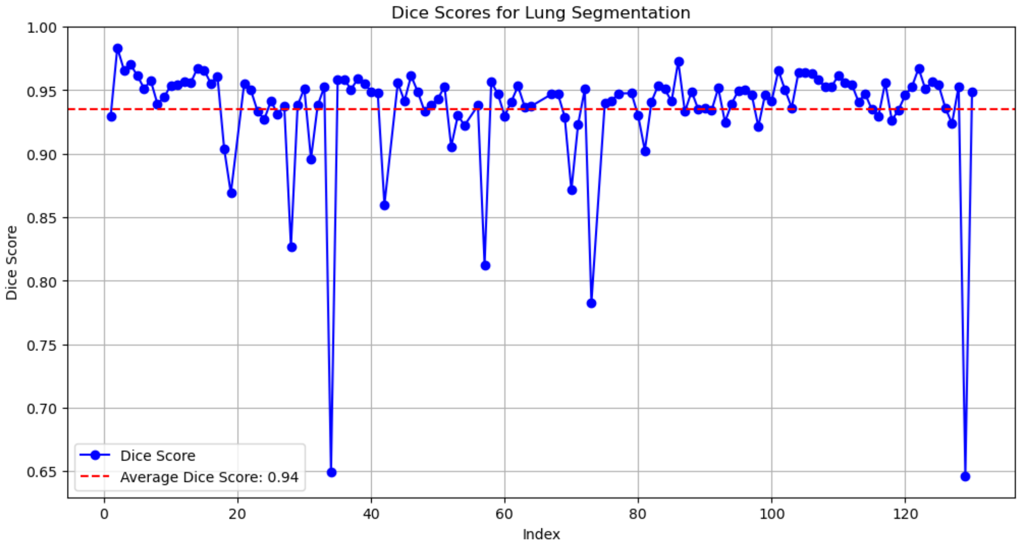


Figure 3.2: Results found on CT-ORG dataset

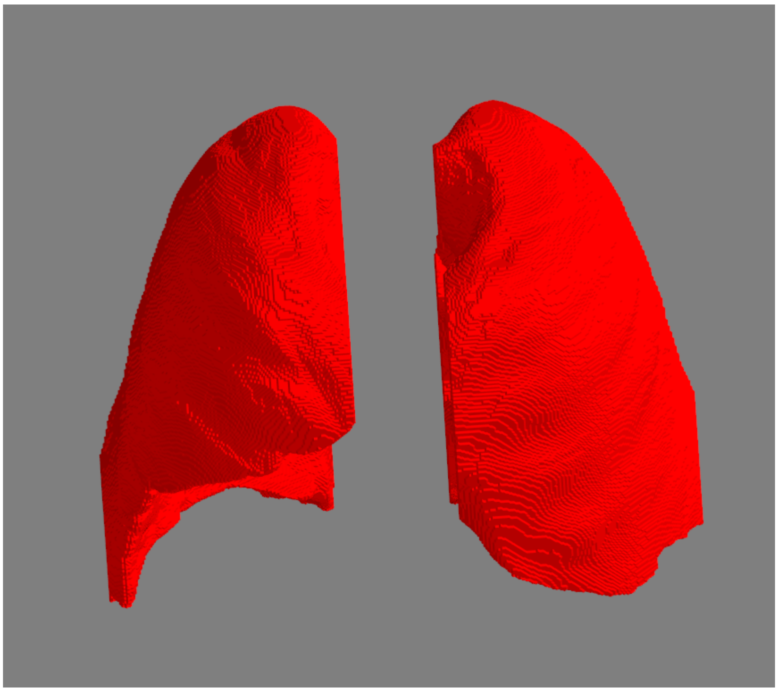


Figure 3.3: 3D plot of a Lung segmentation using the above described algorithm.

4 Liver Segmentation through Medical Image Analysis

Liver segmentation is an essential step in medical image analysis, contributing significantly to the diagnosis and treatment of liver-related diseases. It provides key insights for various clinical applications, such as surgical planning, monitoring of tumor progression, and guiding interventions like biopsies. Unlike lung segmentation, which is often more straightforward due to the organ's well-defined boundaries, liver segmentation is more challenging. The liver's irregular shape, its proximity to neighboring organs like the stomach and kidneys, and the low contrast between the liver and surrounding tissues in medical images make the task difficult.

4.1 Description

In this chapter, we focus on three primary approaches for liver segmentation: Exploration, Flood, and Watershed. These methods are evaluated for their accuracy and robustness in segmenting the liver region from medical images, especially in the presence of noise, varying intensities, and complex anatomical structures. Each approach is designed to tackle the difficulties posed by liver segmentation by considering different aspects of image processing, including region-growing techniques, intensity thresholds, and gradient-based methods. Unlike with lung segmentation, we separate the CT in slices and work with them singularly.

This part of the project was implemented using OpenCV (cv2), scikit-image, SciPy, and NumPy for image processing and analysis.

4.2 Algorithms

4.2.1 Exploration

In this part the exploration technique was used for segmentation, employing a classical method like Breadth-First Search (BFS). This approach was applied to a mask obtained by using various filters, aiming to isolate and segment the region of interest more accurately. The BFS-based exploration start from a point labeled as liver (weakly supervised approach) and allows for the efficient identification and connection of relevant areas, ensuring a more robust segmentation compared to a simple thresholding approach based on filters.

1. **Apply a median filter**

2. **Filter based on HU voxel values**

3. **Start BFS from given labeled points**: given a voxel, possibility to move in each direction (x,y,z). Move can be done if destination is in filtered mask.

4. **Mark each voxel as visited**: the obtained mask has 1 in visited voxels and 0 in all the others.

5. **Fill binary holes**

4.2.2 Flood

The idea now is to improve the quality of the Exploration segmentation, so we introduce tolerance. In this part of the work, segmentation was performed using the flood fill technique with tolerance. This method was applied to a mask obtained through various filters, allowing for a more flexible region-growing process by considering intensity variations. Unlike exploration-based approaches, flood fill with tolerance expands the segmented region dynamically based on local intensity similarities, ensuring a more adaptive and accurate segmentation compared to strict thresholding methods.

1. **Apply a median filter**

2. **Filter based on HU voxel values**

3. **Find boundaries and inclusion mask**

4. **Start flooding with certain tolerance from given points labeled as liver**

5. **Aggregate flooding masks (OR pixelwise operation)**

6. **Filter flooding result with exclusion mask**

7. **Fill binary holes**

Note: Step 5 can be modified in order to improve performance. In this case it seemed that the algorithm was under-estimating the area, so computing the OR of different masks led to bigger areas and higher dice score.

4.2.3 Watershed

In this part of the work, segmentation was performed using the watershed algorithm. This method was applied to a mask obtained through various filters, treating the image as a topographic surface where regions grow from predefined markers until they meet at boundaries. Unlike exploration-based approaches, watershed segmentation is particularly effective in separating adjacent structures by leveraging gradient information, ensuring a precise delineation of the region of interest even in complex scenarios.

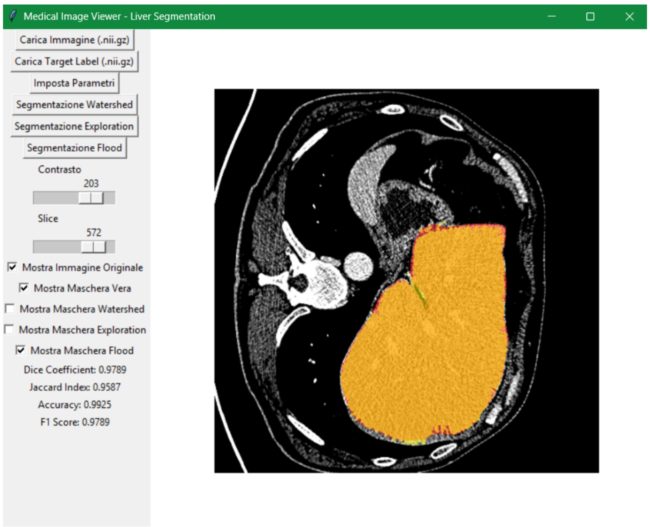
1. **Apply a median filter**

2. **Filter based on HU voxel values**

3. **Standardize and find gray scale image**

4. **Find background value and threshold the image**
5. **Fill binary holes**
6. **Opening**: to remove small noises
7. **Find non-relevant areas**: by dilation and distance thresholding
8. **Apply watershed algorithm**
9. **Find connected components**
10. **New mask equal 0 in biggest component (background)**
11. **Find new connected components and select biggest one**

Note: Beside this, a small GUI was implemented in order to show visually the results.


Medical Image Viewer - Liver Segmentation
Carica Immagine (.nii.gz)
Carica Target Label (.nii.gz)
Imposta Parametri
Segmentazione Watershed
Segmentazione Exploration
Segmentazione Flood

Contrasto 203

Slice 572

☑ Mostra Immagine Originale
☑ Mostra Maschera Vera
□ Mostra Maschera Watershed
□ Mostra Maschera Exploration
☑ Mostra Maschera Flood

Dice Coefficient: 0.9789
Jaccard Index: 0.9587
Accuracy: 0.9925
F1 Score: 0.9789

Figure 4.1: Developed GUI present in Github repository.

4.3 Results

The performance of the three proposed liver segmentation approaches—Exploration, Flood, and Watershed—was evaluated using the Dice similarity coefficient.

Figure 4.2 summarizes the average evaluation Dice scores obtained for each method:

- **Watershed:** $Dice_{avg} \approx 0.9152$
- **Flood:** $Dice_{avg} \approx 0.9192$

- **Exploration:** $Dice_{avg} \approx 0.8678$

Evaluation was conducted on slices with largest labeled area in each CT. The results indicate that the Flood and Watershed methods outperform the Exploration approach. The Flood method achieved the highest Dice score, demonstrating its effectiveness in handling intensity variations while maintaining accurate region-growing behavior. The Watershed algorithm performed comparably well, particularly in cases where clear boundaries were present in the medical images.

The Exploration approach, while useful for initial segmentation, was less effective due to its reliance on a predefined starting point and its sensitivity to noise in the filtered mask. The Breadth-First Search (BFS) strategy provided robust connectivity but struggled in cases where the liver region exhibited irregular intensity distributions or weak boundary definitions.

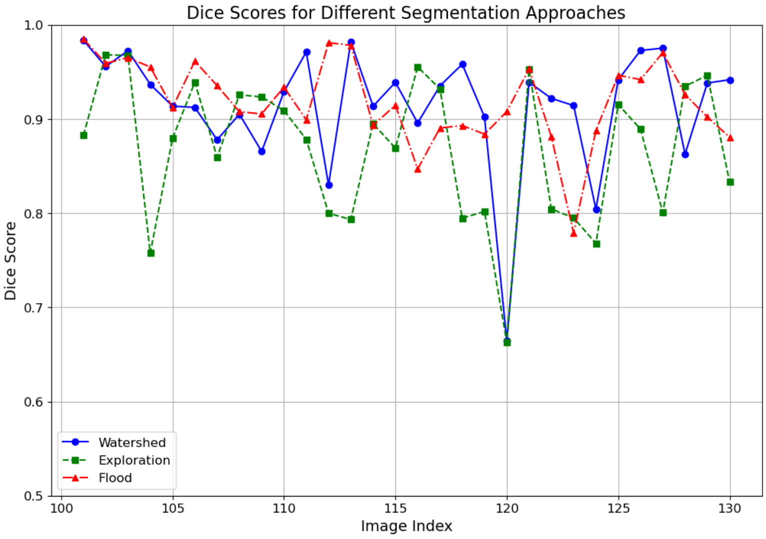


Figure 4.2: Comparison on different evaluation cases.

5 Liver Segmentation through U-Net

In this chapter we analyze a complex solution using a U-Net, this part of the project was inspired by a paper published by Giri et al [3]. I decided to include this approach (even if it does not fully comply with the course objectives) to demonstrate how more complex methods outperform the simpler processing techniques proposed earlier. The dataset used in this chapter is MSD, the same for Chapter 4.

5.1 Description

U-Net is a type of convolutional neural network (CNN) architecture that was originally designed for biomedical image segmentation. It was introduced by Olaf Ronneberger, Philipp Fischer, and Thomas Brox in their 2015 paper [9]. The architecture is particularly well-suited for tasks where the goal is to produce a pixel-wise segmentation of an image, meaning that each pixel in the input image is assigned a label or class.

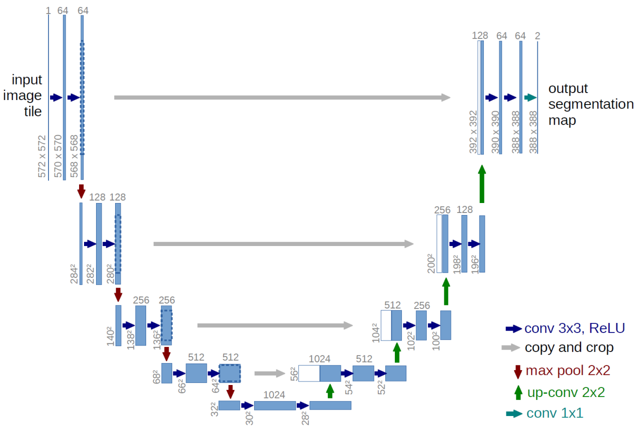


Figure 5.1: U-Net architecture [9]

This part of the project was implemented using Torch and MONAI.

5.2 Algorithm

The dataset MSD is not composed of CT scans of the same size. The first preprocessing consists on splitting each image in chunks of 64 slices, ensuring all slices are used for training.

The U-Net used has 5 levels, at each level the channels duplicate starting from 16. The loss function selected is the Dice Loss. In this case the number of epoch used is 600. Training was conducted on an old video card (NVIDIA GTX 980) with CUDA and cuDNN acceleration, alternating training and testing on each epoch.

Before training and testing, CT pass through a set of preprocessing steps:

1. **Data Loading**: The image and segmentation files are loaded in supported formats, such as NIfTI or DICOM, preserving their spatial properties.
2. **Channel-First Format**: The data is converted to a channel-first format, ensuring compatibility with deep learning models that expect channel-first input.
3. **Reorientation to Standard Coordinate System**: The images and segmentations are aligned to the standard RAS (Right-Anterior-Superior) coordinate system.
4. **Intensity Normalization**: The intensity values of the images are scaled to a fixed range, improving model stability.
5. **Foreground Cropping**: The volume is cropped to remove background regions.
6. **Resizing to Fixed Spatial Dimensions**: The image and segmentation are resized to a predefined spatial size. In this case 128x128x64, this is also limited because of the low computational power of my computer.
7. **Conversion to Tensor**: The processed data is converted into PyTorch tensors, making it suitable for deep learning pipelines.

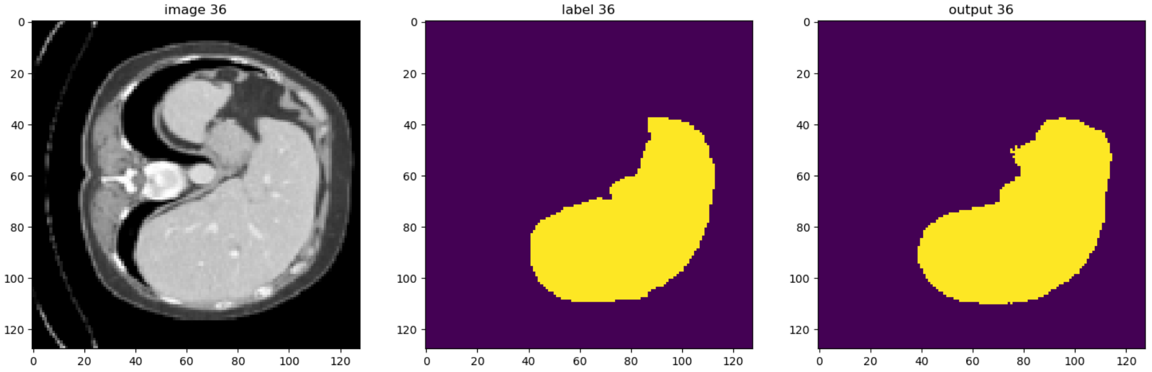


Figure 5.2: Prediction of the trained U-Net of a slice in 128x128 resolution

5.3 Results

In this section we analyze the results obtained with the trained U-Net. The first thing to say is that these results are highly limited by my computer's computational power and my time. The U-Net was trained on 100 samples of the MSD dataset and evaluated on the remaining 30. The data size increased a lot separating each CT in 64 slides, roughly 5 times. The training phase took around 2500 minutes ≈ 1 day and 18 hours. The evaluation average dice score found was $Dice_{avg} \approx 0.9298$, found on slices with largest labeled area for each chunk. Surely training the U-Net with more computational power and time, leads to better results outclassing image processing methods. From the Figure 5.4 it is possible to see that U-Net, compared to Chapter 2 approaches, is more stable on prediction results.

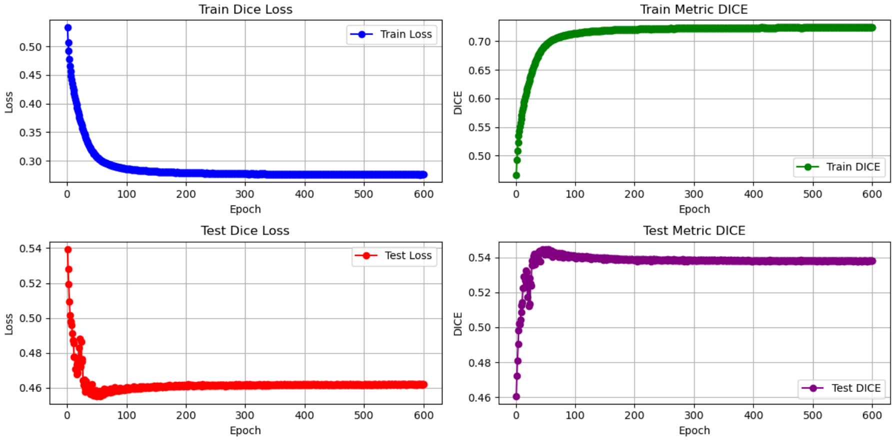


Figure 5.3: Training phase results on 600 epochs

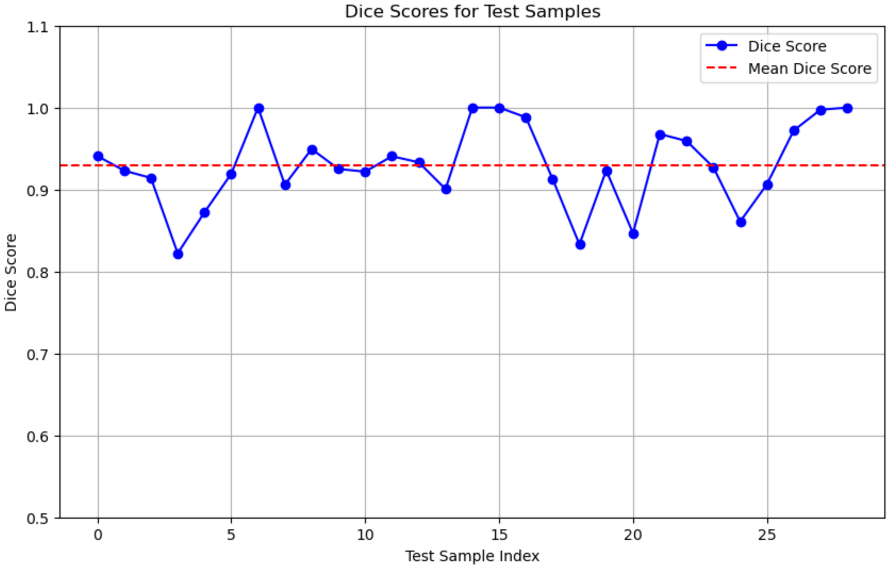


Figure 5.4: Evaluation score for each CT.

6 Conclusions and Future Improvements

In this work, we explored various approaches for lung and liver segmentation in medical images, ranging from traditional image processing techniques to advanced deep learning models. For lung segmentation, a rule-based algorithm leveraging Hounsfield Unit (HU) values and morphological operations achieved an average Dice score of 0.9355. For liver segmentation, three image-processing-based algorithms were implemented. While these methods provided reasonable results, they highlighted the challenges posed by the liver's irregular shape, low contrast, and proximity to other organs. Meanwhile U-Net-based approach achieved a mean Dice score of 0.9298, outperforming the traditional methods. This underscores the power of deep learning in handling complex segmentation tasks.

Overall, this project demonstrated that while traditional methods can be effective for simpler tasks like lung segmentation, deep learning models like U-Net are better suited for more challenging problems like liver segmentation. The results also highlighted the importance of dataset quality, pre-processing and post-processing in achieving accurate segmentation.

There are several areas for improvement that could enhance the performance and applicability of the segmentation methods:

- The parameters of the algorithms can be adjusted to achieve optimal results, based on the specific characteristics of the dataset.
- Using larger and more diverse datasets.
- Trying advanced deep learning architectures such as 3D U-Net, Attention U-Net, or Transformer-based models, SAM could further improve segmentation accuracy, especially for complex organs.
- Use different post and pre-processing techniques such as morphological operations could refine the segmentation masks and reduce errors.

Additional informations on how to run the code and the project can be found in the Github repository here: https://github.com/avendramini/Lung_Liver_Segmentation.git

Bibliography

[1] Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Annette Kopp-Schneider, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M Summers, et al. The medical segmentation decathlon. *Nature communications*, 13(1):4128, 2022.

[2] Katherine A Estes, Jacob C Peterson, Adam M Szlachetka, and R Lee Mosley. X-ray, positron emission, and single photon emission tomographic bioimaging. *Current Laboratory Methods in Neuroscience Research*, pages 271–292, 2014.

[3] Jayant Giri, T Sathish, Taukeer Sheikh, Neeraj Sunheriya, Pallavi Giri, Rajkumar Chadge, Chetan Mahatme, and A Parthiban. Automatic liver segmentation using u-net deep learning architecture for additive manufacturing. *Interactions*, 245(1):90, 2024.

[4] K Greenway, A Campos, H Knipe, et al. Hounsfield unit. *Radiopaedia.org*, 2025. Reference article, Accessed on 01 Feb 2025.

[5] Nghi Huynh. Understanding evaluation metrics in medical image segmentation, 2020. Accessed: 2025-01-27.

[6] Jun Ma, Yao Zhang, Song Gu, Cheng Zhu, Cheng Ge, Yichi Zhang, Xingle An, Congcong Wang, Qiyuan Wang, Xin Liu, et al. Abdomenct-1k: Is abdominal organ segmentation a solved problem? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10):6695–6714, 2021.

[7] MICCAI. Flare22 - fast and low-resource semi-supervised abdominal organ and tumor segmentation challenge, 2022.

[8] Blaine Rister, Darvin Yi, Kaushik Shivakumar, Tomomi Nobashi, and Daniel L Rubin. Ct-org, a new dataset for multiple organ segmentation in computed tomography. *Scientific Data*, 7(1):381, 2020.

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, pages 234–241. Springer, 2015.