Segmentation with AI

Lucía Cubero – lcubero@ing.uc3m.es

Sergio Carreras – scarrera@ing.uc3m.es

MINISTERIO
DE CIENCIA, INNOVACIÓN
Y UNIVERSIDADES

Cofinanciado por
la Unión Europea

AGENCIA
ESTATAL DE
INVESTIGACIÓN

Comunidad
de Madrid

Index

- What is medical image segmentation
- Traditional Methods
- Introduction to Artificial Intelligence
- DL Applications in Medical Imaging
- Segmentation with AI
- Total Segmentator
- Pipeline with AI Segmentation

Medical Image Segmentation

Medical Image Segmentation

- Defined as the process of contouring a medical image into meaningful regions or structures.

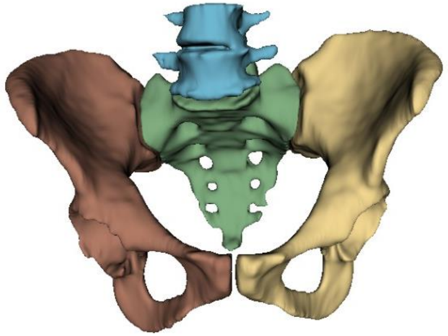
- Types
  - Binary segmentation (one organ)
  - Multi-class segmentation (multiple organs)
  - Instance segmentation (multiple instances)

Medical Image Segmentation

Original CT

Binary segmentation

Multi-class segmentation



Medical Image Segmentation

- Uses of segmentation in medical imaging
  1. Treatment planning (in radiotherapy, to contour the organs at risk)
  2. Disease quantification (to measure the tumor volumen over time)
  3. Surgical navigation

- Medical modalities used in segmentation:
  - CT
  - MRI
  - PET
  - Ultrasound

Segmentation – Traditional Methods

Traditional Methods

- Manual segmentation
  - Pros: high accuracy
  - Cons: time-consuming, subjective
- Thresholding: segments regions by selecting pixel with a certain intensity
  - Pros: fast and simple
  - Cons: poor performance with noisy images or overlapping intensities
- Region growing: starts from a “seed point” and includes neighboring pixels/voxels with similar intensity
  - Pros: captures connected structures well
  - Cons: sensitive to seed placement and noise

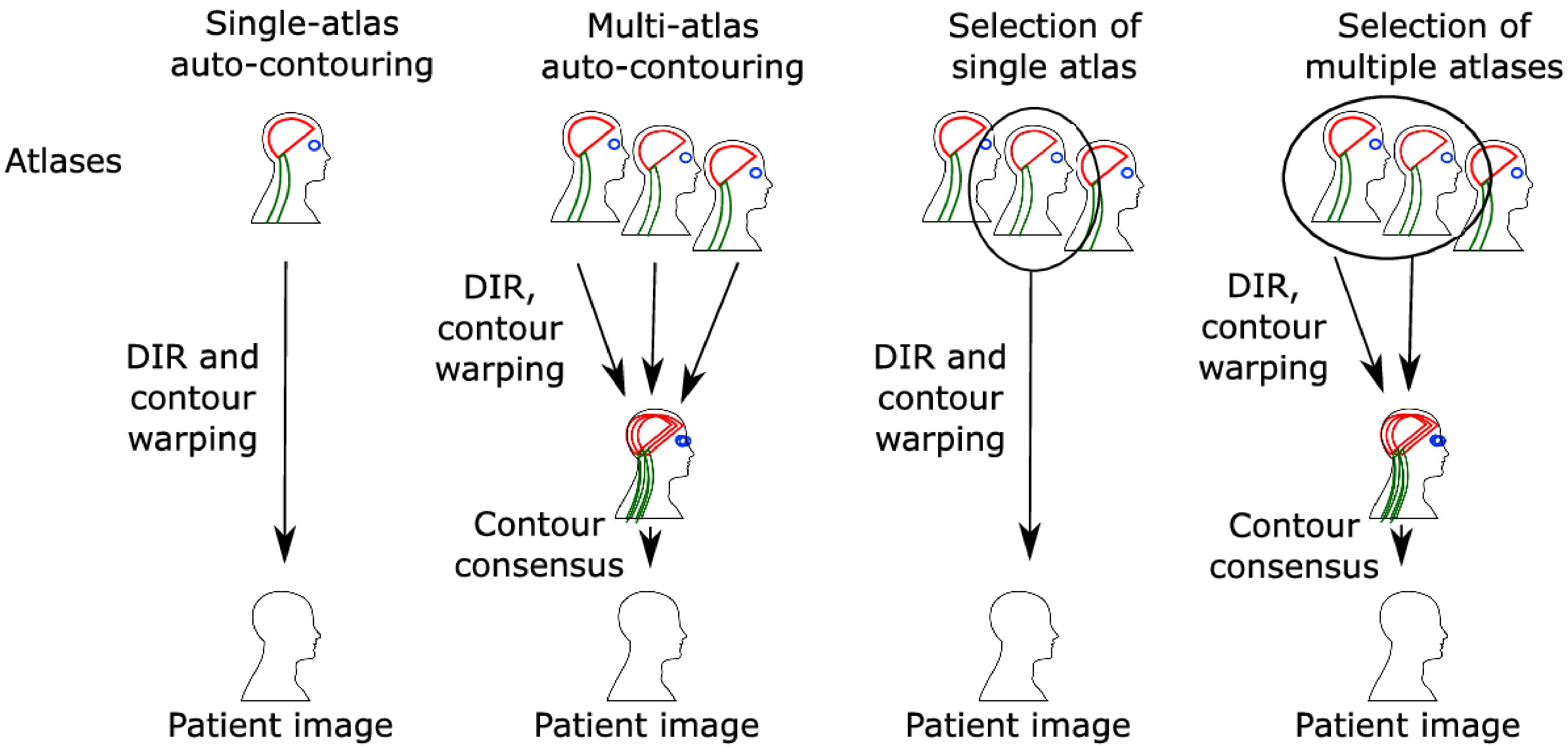
3D Slicer SOFA

Traditional Methods

- **Edge Detection:** identifies boundaries between regions by detecting sharp changes in intensity
  - Pros: useful for highlighting contours
  - Cons: struggles in low-contrast or noisy images
- **Active Contours (Snakes):** curves that evolve over time to lock onto object boundaries by minimizing an energy function
  - Pros: can refine boundaries smoothly
  - Cons: requires good initialization
- **Atlas-Based Auto-Segmentation (ABAS):** uses a pre-labeled image (atlas) and deforms it to match the target image using image registration
  - Pros: uses manual labels
  - Cons: depends on registration quality

Traditional Methods

- <u>Atlas-Based Auto-Segmentation (ABAS)</u>



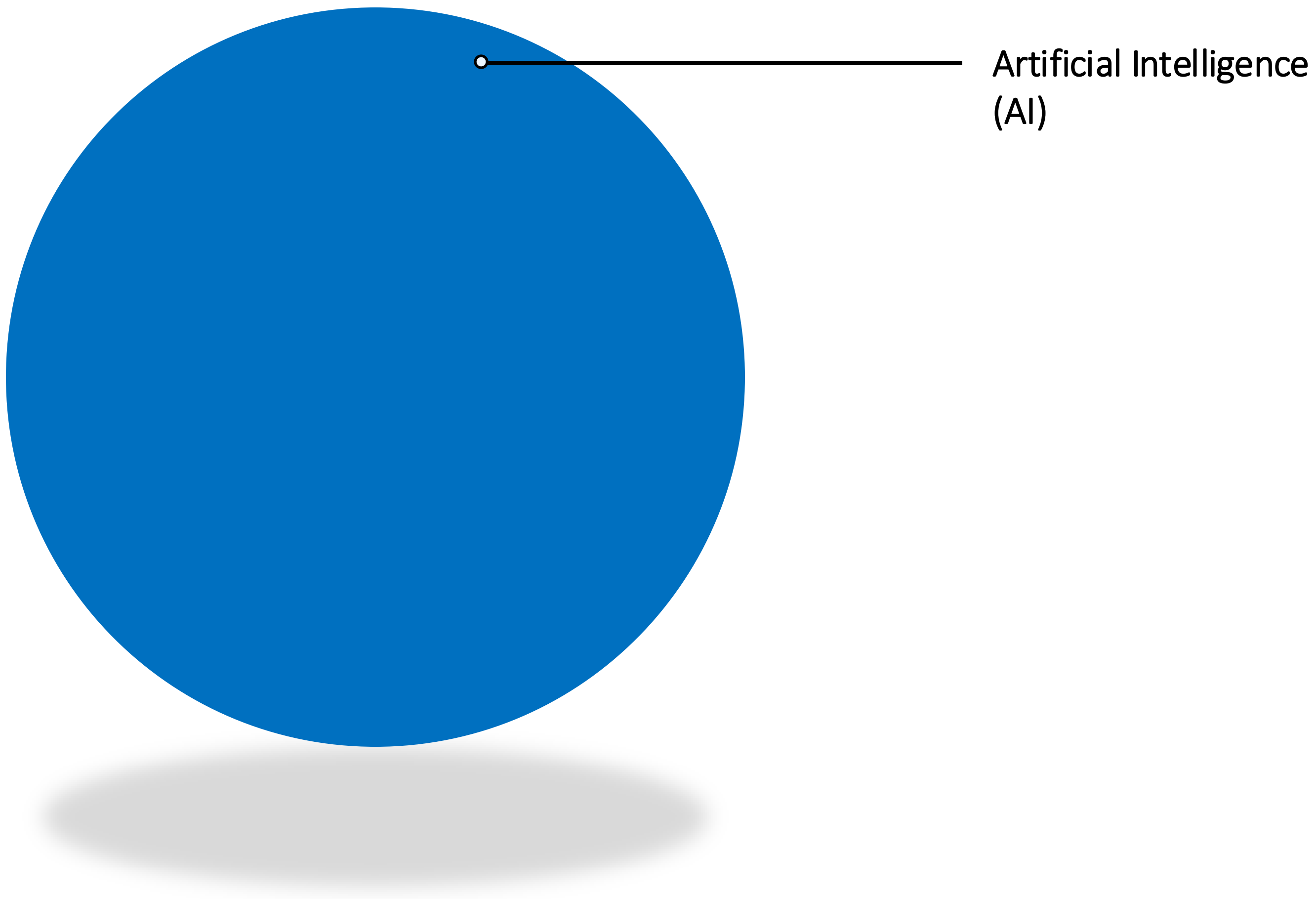
Schipaanboord, B W K et al. "Can Atlas-Based Auto-Segmentation Ever Be Perfect? Insights From Extreme Value Theory." *IEEE Transactions on Medical Imaging* 38 (2019): 99-106.

Introduction to Artificial Intelligence

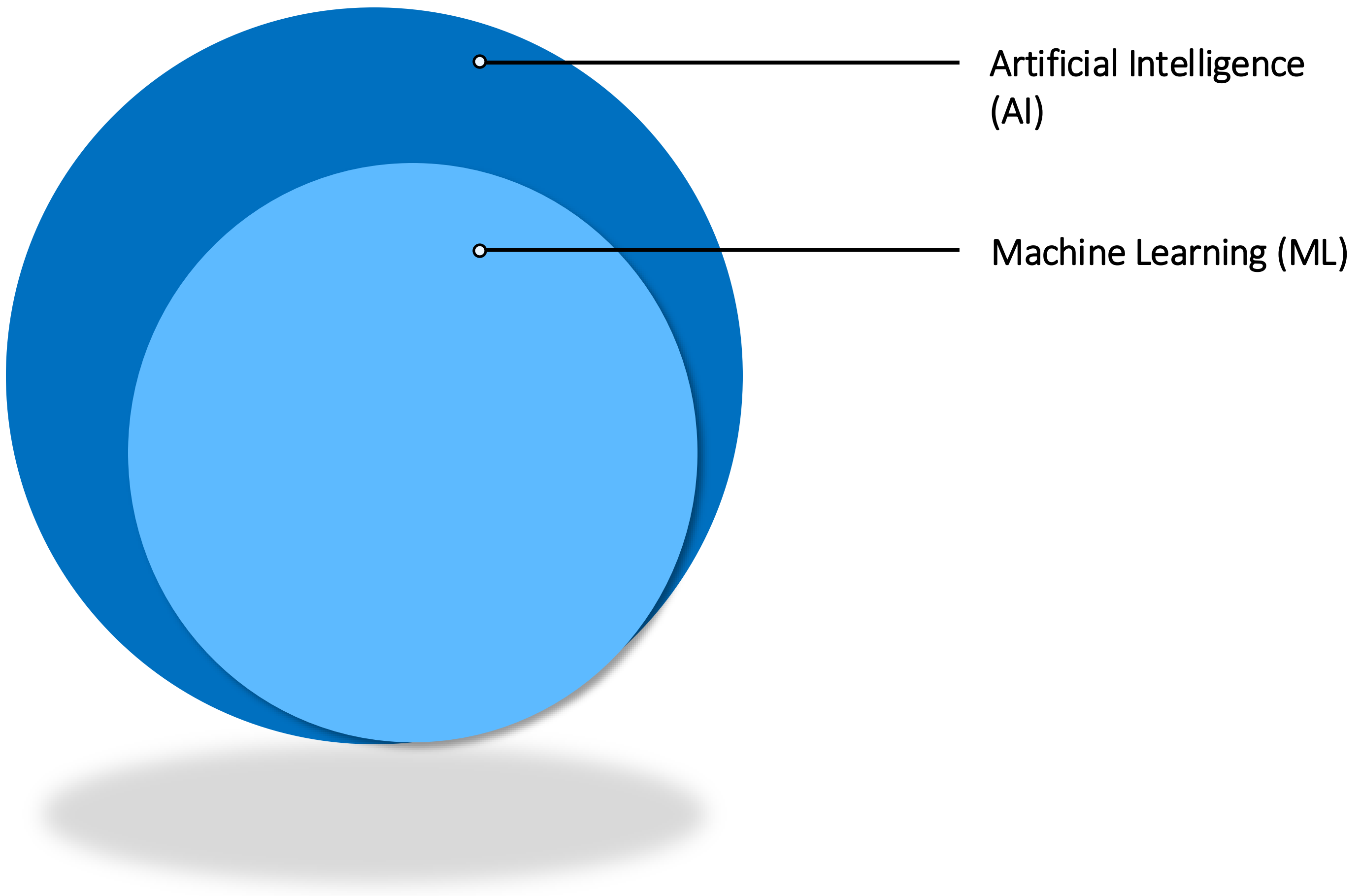
Artificial Intelligence

- **Artificial Intelligence (AI):** the ability of a machine to perform tasks that usually require human intelligence

Artificial Intelligence



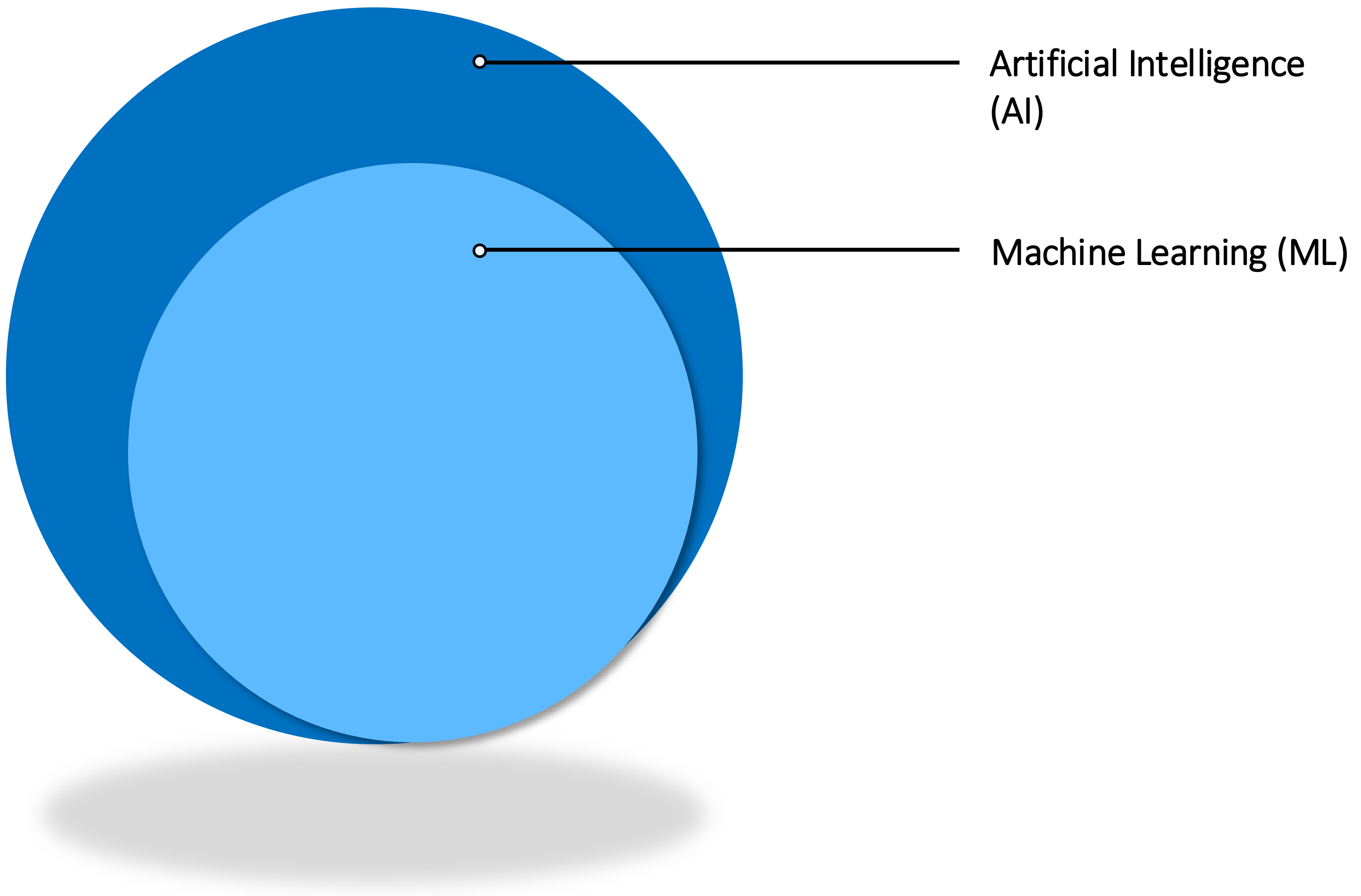
Artificial Intelligence



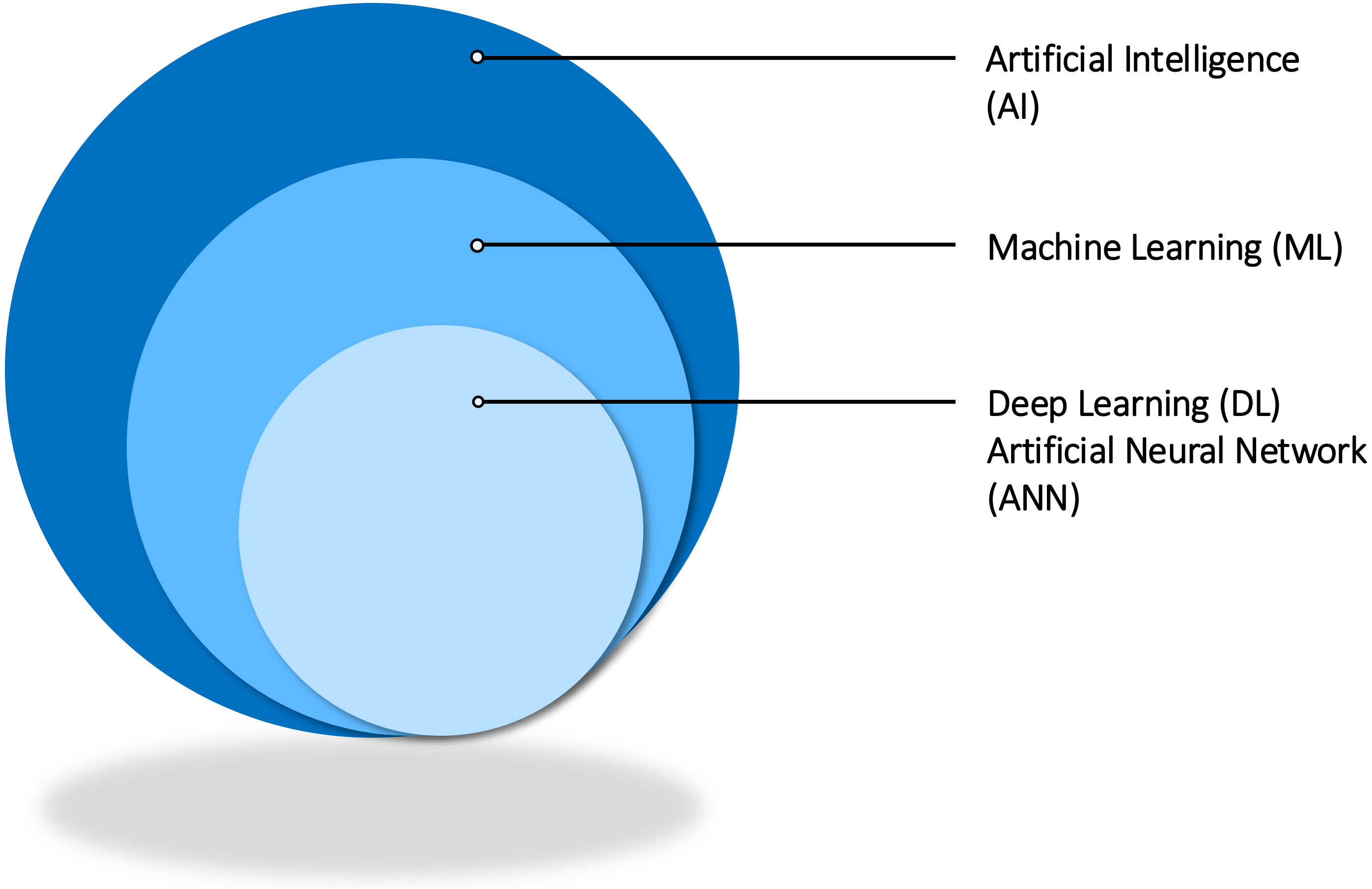
Artificial Intelligence

- **Artificial Intelligence (AI):** the ability of a machine to perform tasks that usually require human intelligence
- **Machine Learning (ML):** a subset of AI that includes algorithms that learn patterns from data without being explicitly programmed
  - Human experts select the imaging features that seem most relevant for the task

Artificial Intelligence



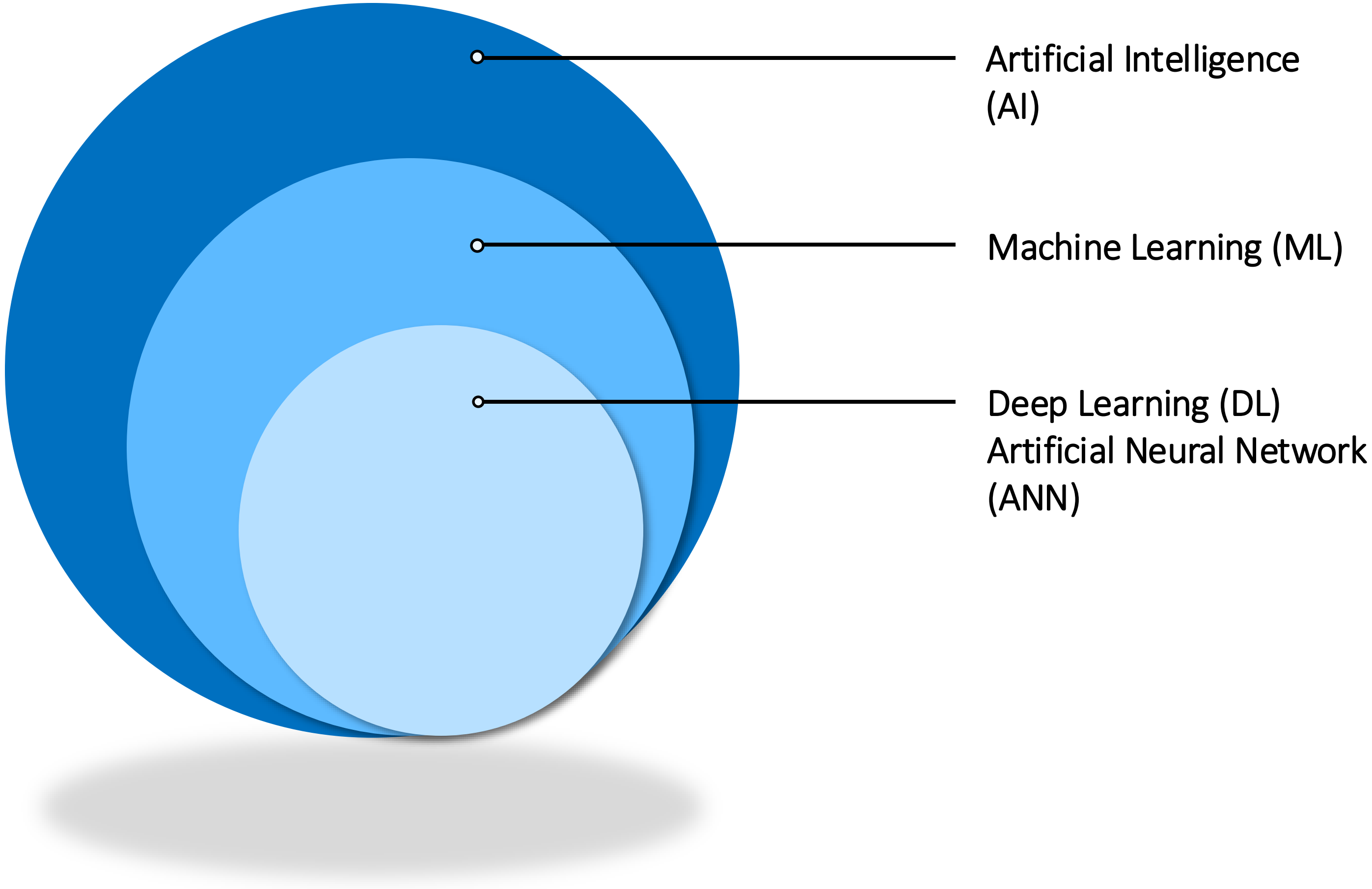
Artificial Intelligence



Artificial Intelligence

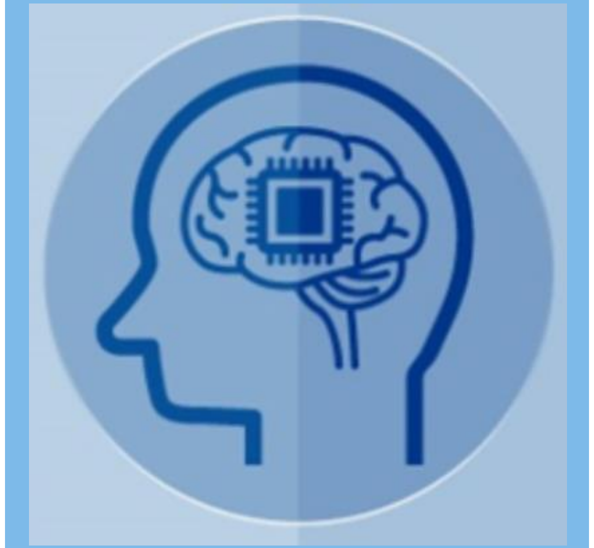
- **Artificial Intelligence (AI):** the ability of a machine to perform tasks that usually require human intelligence
- **Machine Learning (ML):** a subset of AI that includes algorithms that learn patterns from data without being explicitly programmed
  - Human experts select the imaging features that seem most relevant for the task
- **Deep Learning (DL):** a subset of ML of models that can solve complex tasks by automatically learning which features are most useful
  - Most DL models are based on Artificial Neural Networks (ANNs)
  - No manual feature selection is needed

Artificial Intelligence



Artificial Intelligence

Artificial Intelligence



Machine simulating human intelligence to perform tasks

Machine Learning

Algorithms that learn patterns from the data

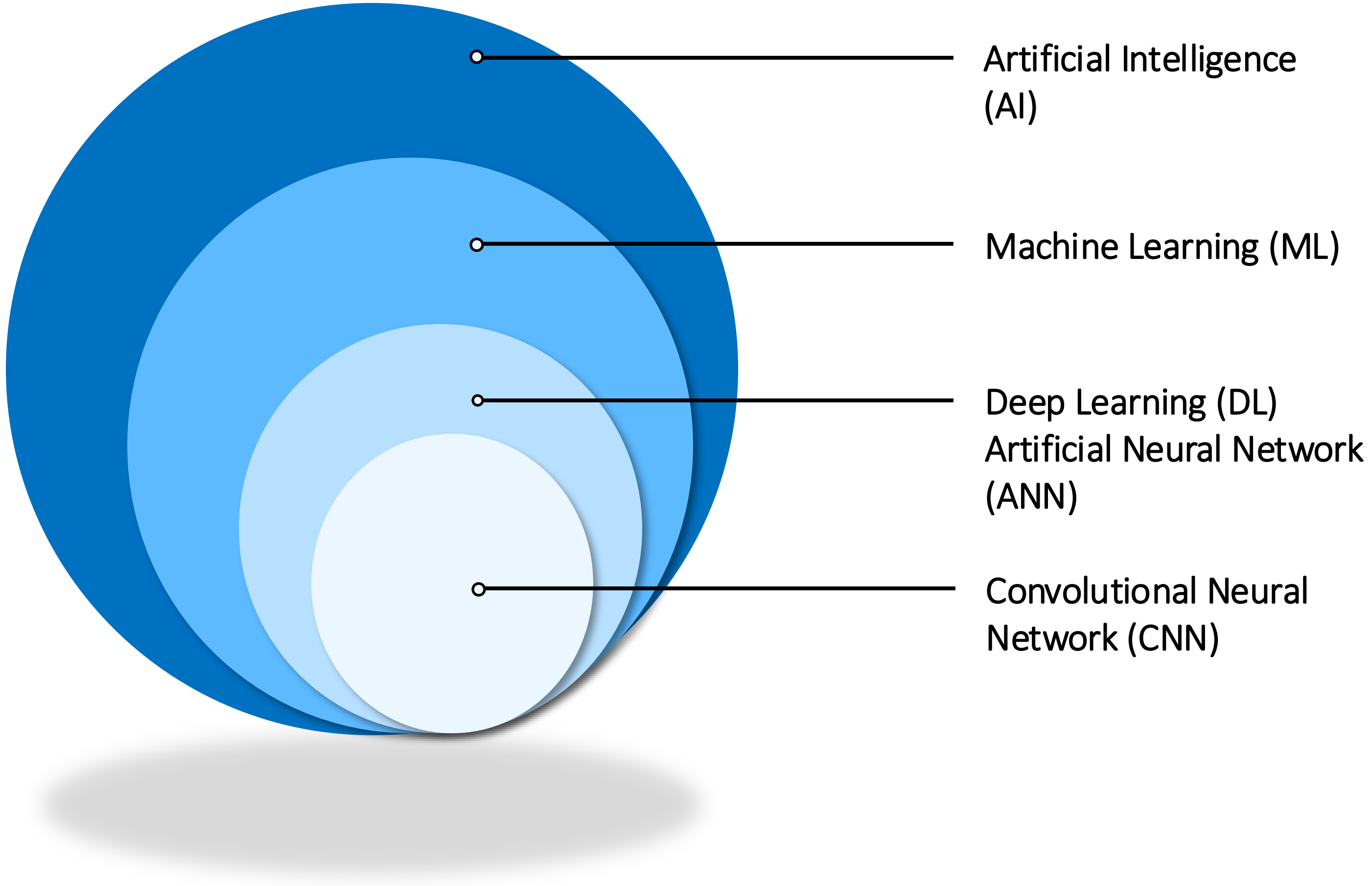
Deep Learning

Models that automatically solve complex tasks

1950 | 1960 | 1970 | 1980 | 1990 | 2000 | 2005 | 2010 | 2015 | 2020

3D Slicer + SOFA

Artificial Intelligence



Artificial Intelligence

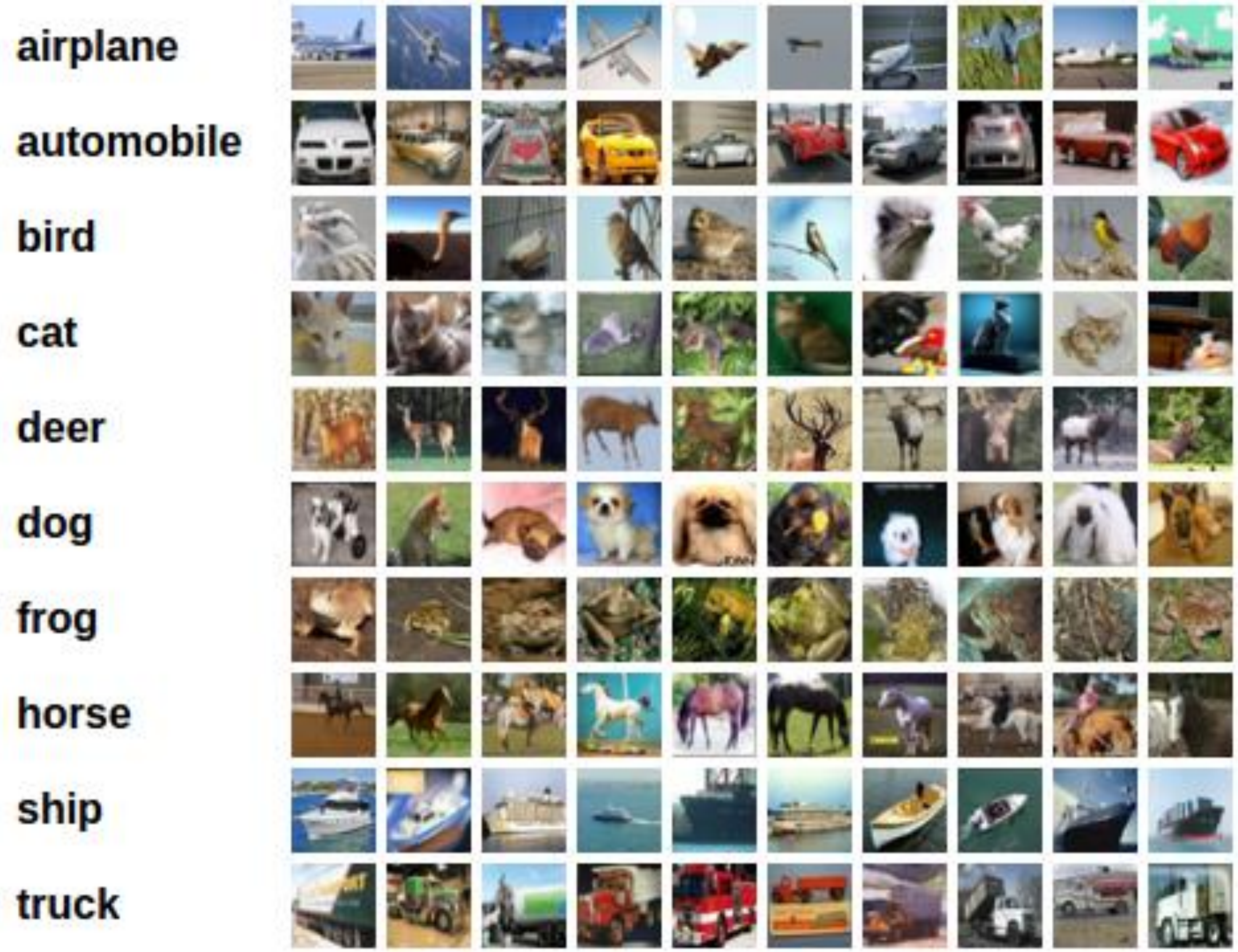
- **Artificial Intelligence (AI):** the ability of a machine to perform tasks that usually require human intelligence
- **Machine Learning (ML):** a subset of AI that includes algorithms that learn patterns from data without being explicitly programmed
  - Human experts select the imaging features that seem most relevant for the task
- **Deep Learning (DL):** a subset of ML of models that can solve complex tasks by automatically learning which features are most useful
  - Most DL models are based on Artificial Neural Networks (ANNs)
  - No manual feature selection is needed
- **Convolutional Neural Network (CNN):** a DL model that makes the explicit assumption that the inputs are images

Recent Advances in DL

- What made DL explode in the last decade?
  - Big data: more labeled and large datasets available (e.g., ImageNet, TCIA)
  - GPU acceleration: faster and more efficient computers
  - Development of DL algorithms
  - Open-source frameworks: PyTorch, TensorFlow

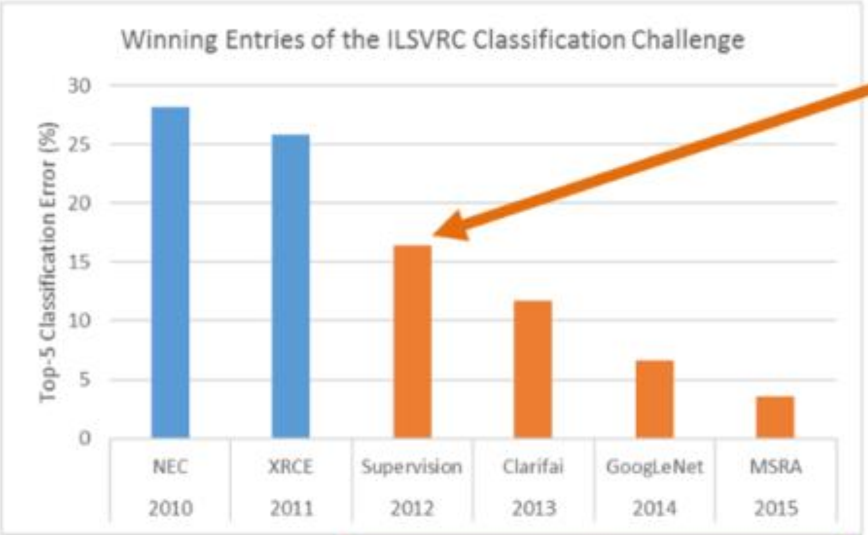
Recent Advances in DL

- ImageNet (2010)
  - Image classification
  - Massive labeled image dataset (~1.2 million images, 1000 classes)
  - Now it has expanded to 14 million images and 21000 classes



Recent Advances in DL

- **AlexNet (2012)**
  - Image classification
  - First CNN to win ImageNet
  - Introduced key concepts in DL: ReLU activation (faster training), dropout (reduces overfitting), GPU training (parallel processing)



Winning entry = convolutional neural network

In 2012, Krizhevsky et al. from the Univ. of Toronto achieved a performance breakthrough (markedly decreased error) using a deep convolutional neural network.

Since 2012, all winning entries (and most entries overall) have used convolutional neural networks.

Recent Advances in DL

- **VGGNet (2014)**
  - Image classification
  - Introduced very deep networks with 3x3 convolutions
  - Models VGG16 (16 layers) and VGG19 (19 layers) became standard backbones
  - Popular feature extractor

- 2014-2015
  - **GoogLeNet / Inception**
  - **ResNet**
  - **DenseNet**

Recent Advances in DL

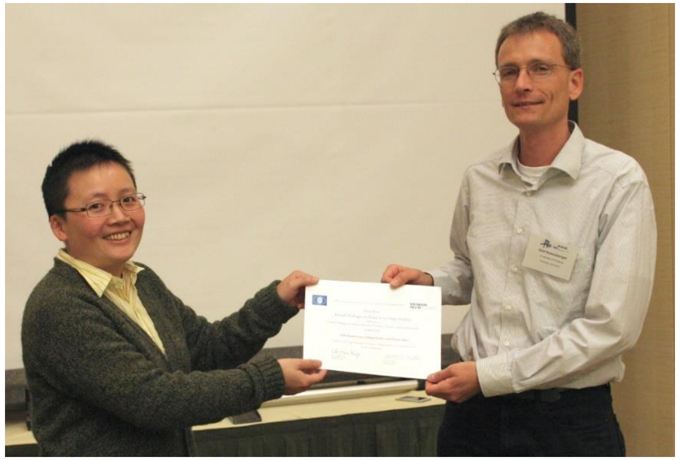
- **UNet (2015)**
  - Biomedical image segmentation
  - Encoder-decoder structure with skip connections
  - It is still the gold-standard for 2D / 3D medical image segmentation

U-Net: Convolutional Networks for Biomedical Image Segmentation

Olaf Ronneberger, Philipp Fischer, and Thomas Brox

>100,000 times cited (Google Scholar)

Winner of the Dental X-Ray Image Segmentation Challenge



Winner of the Cell Tracking Challenge

DL Applications in Medical Imaging

Applications in Medical Imaging

- Why apply DL to medical imaging?
  - Medical imaging is a data-rich field
  - Many challenges appear in daily clinic:
    - Increasing workload for radiologists

3D Slicer

SOFA

Applications in Medical Imaging

- Why apply DL to medical imaging?
  - Medical imaging is a data-rich field
  - Many challenges appear in daily clinic:
    - Increasing workload for radiologists
    - Need for fast, accurate, and consistent interpretation, diagnosis, and decision-making
    - Clinician intra and inter-variability
  - DL can help in solving these problems
  - **DL can assist, not replace**

Applications in Medical Imaging

- **Medical image reconstruction**: reconstruction from low dose data, fast acquisitions.. in CT and MR
- **Medical image enhancement**: denoising, super-resolution, MR bias field correction, image harmonization
- **Medical image segmentation**
- **Medical image registration**
- **Computer-aided detection and diagnosis (CAD):** localize bounding box containing object and diagnose
- **Surgical video analysis:** skill assessment, tool detection, guidance, navigation

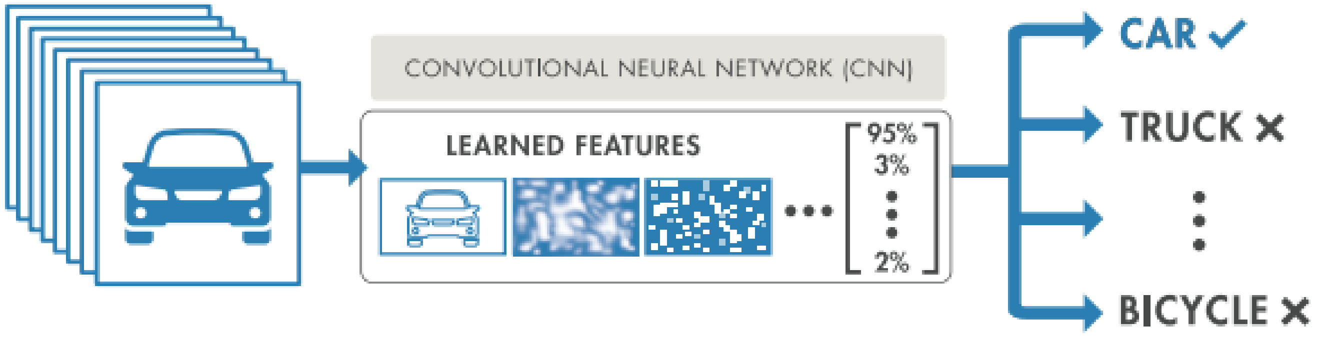
Challenges in Medical Imaging

- Medical images can be 3D (CT, MR, PET, …), which requires huge computational resources
- Data labeling is very subjective, due to inter- and intra-clinician variabilities
- Acquiring large medical databases is very complex, and they are usually limited to one medical center and country
- Clinicians do not have any education on DL, and many are against the implementation of DL in the clinic

Segmentation with AI

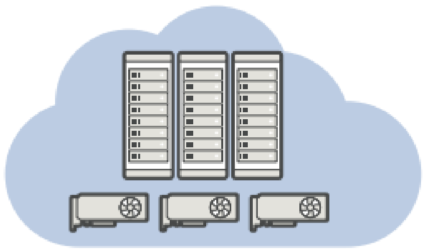
Segmentation with DL

- Features are learned from the data – no feature extraction



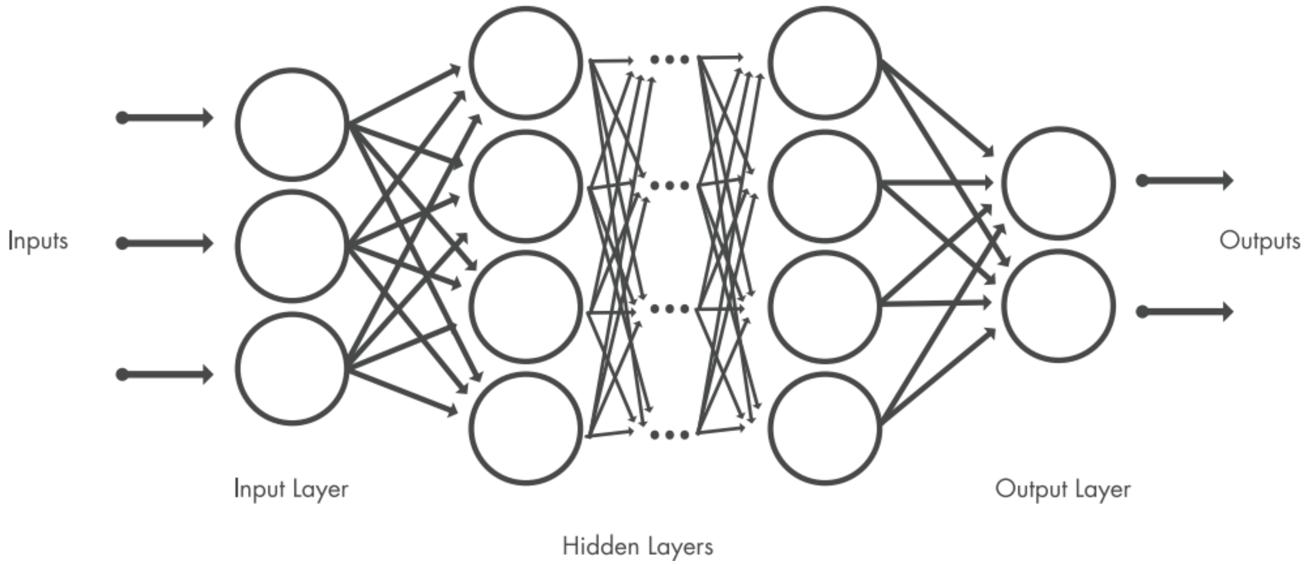
- Training demands computer power
  - Solved thanks to GPUs and Open Toolkits

TensorFlow Keras PyTorch Caffe NVIDIA



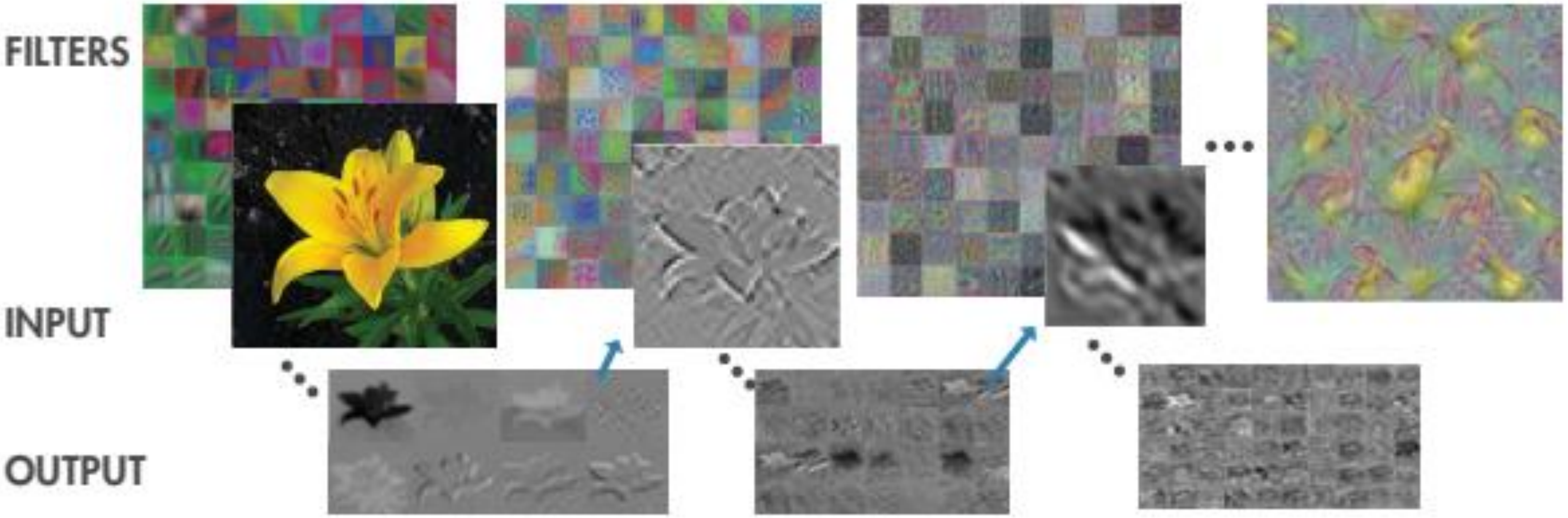
Segmentation with DL

- From training data, the network understands the specific features associated with a category.
- Network learns directly from the data—we have no influence over what features are being learned
  - Multiple nonlinear processing layers
  - Simple elements operating in parallel



Segmentation with DL – CNNs

- When working with images, Convolutional Neural Layers learn to extract features from images



Segmentation with DL – CNNs

A convolutional neural network (CNN, or ConvNet) is one of the most popular algorithms for deep learning with images and video.

Like other neural networks, a CNN is composed of an input layer, an output layer, and many hidden layers in between.

Feature Detection Layers

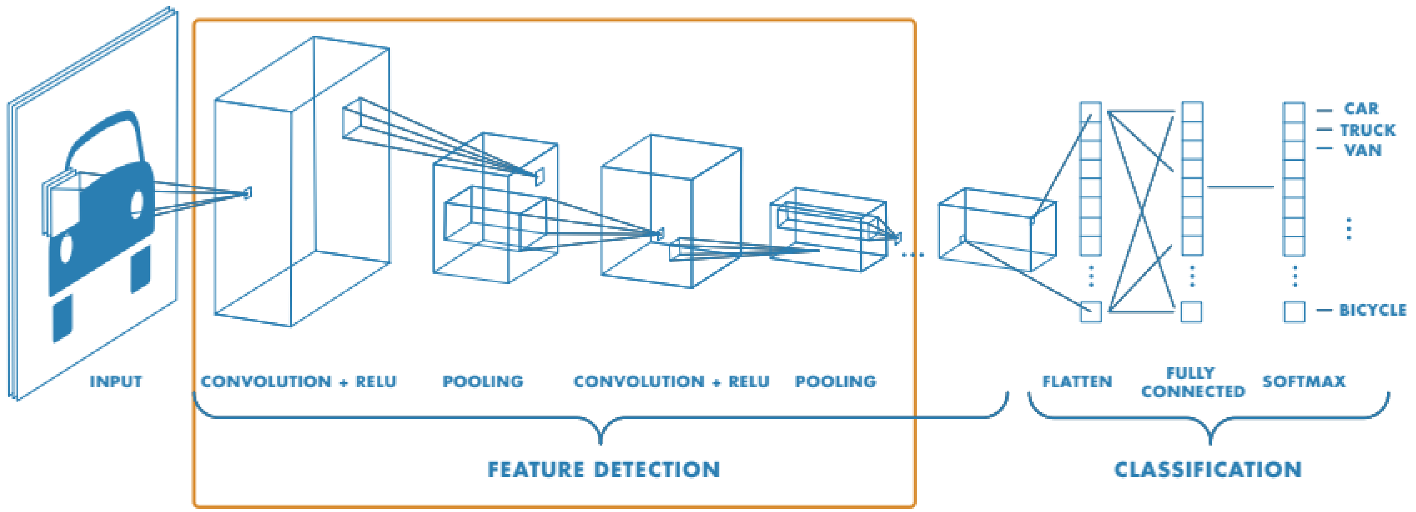
These layers perform one of three types of operations on the data: convolution, pooling, or rectified linear unit (ReLU).

Convolution puts the input images through a set of convolutional filters, each of which activates certain features from the images.

Pooling simplifies the output by performing nonlinear downsampling, reducing the number of parameters that the network needs to learn about.

Rectified linear unit (ReLU) allows for faster and more effective training by mapping negative values to zero and maintaining positive values.

These three operations are repeated over tens or hundreds of layers, with each layer learning to detect different features.



Segmentation with DL – CNNs

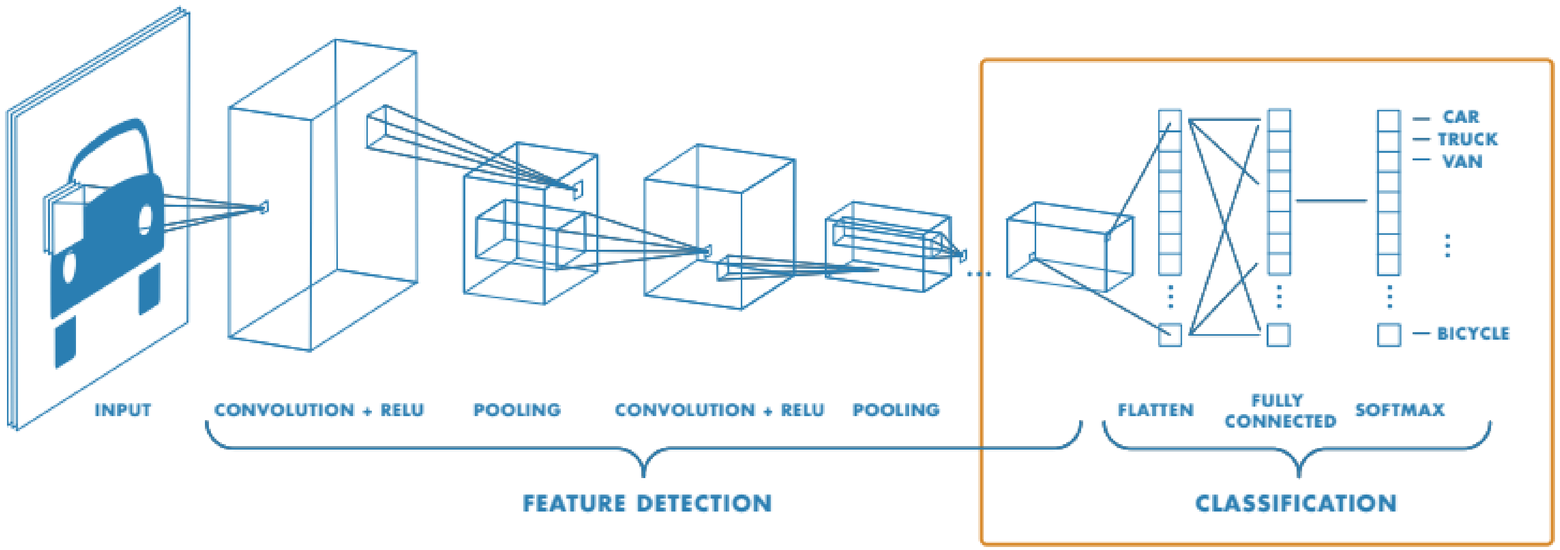
Classification Layers

After feature detection, the architecture of a CNN shifts to classification.

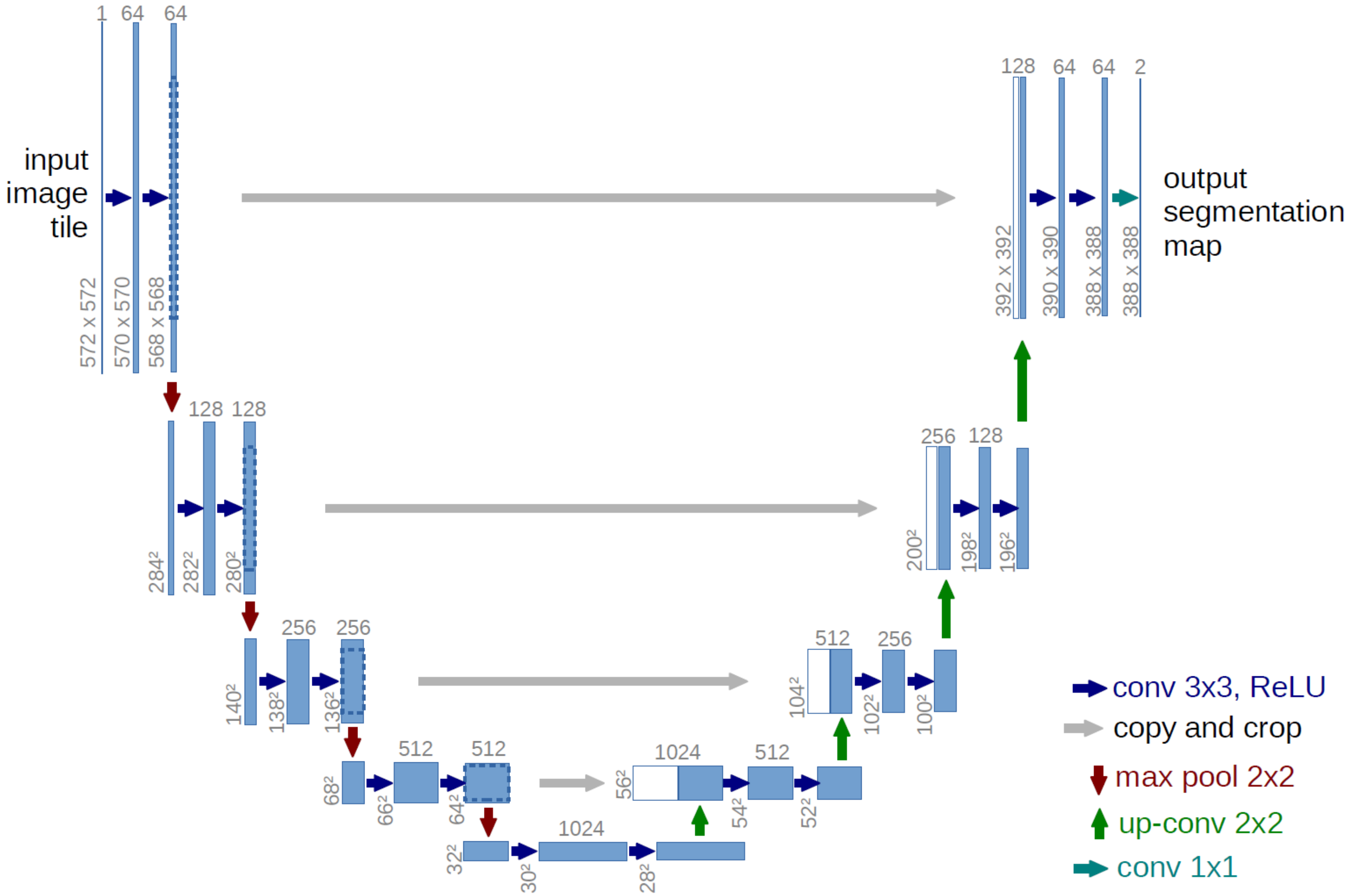
The next-to-last layer is a **fully connected layer** (FC) that outputs a vector of K dimensions where K is the number of classes that the network will be able to predict. This vector contains the probabilities for each class of any image being classified.

The final layer of the CNN architecture uses a **softmax** function to provide the classification output.

There is no exact formula for selecting layers. The best approach is to try a few and see how well they work—or to use a pretrained network.

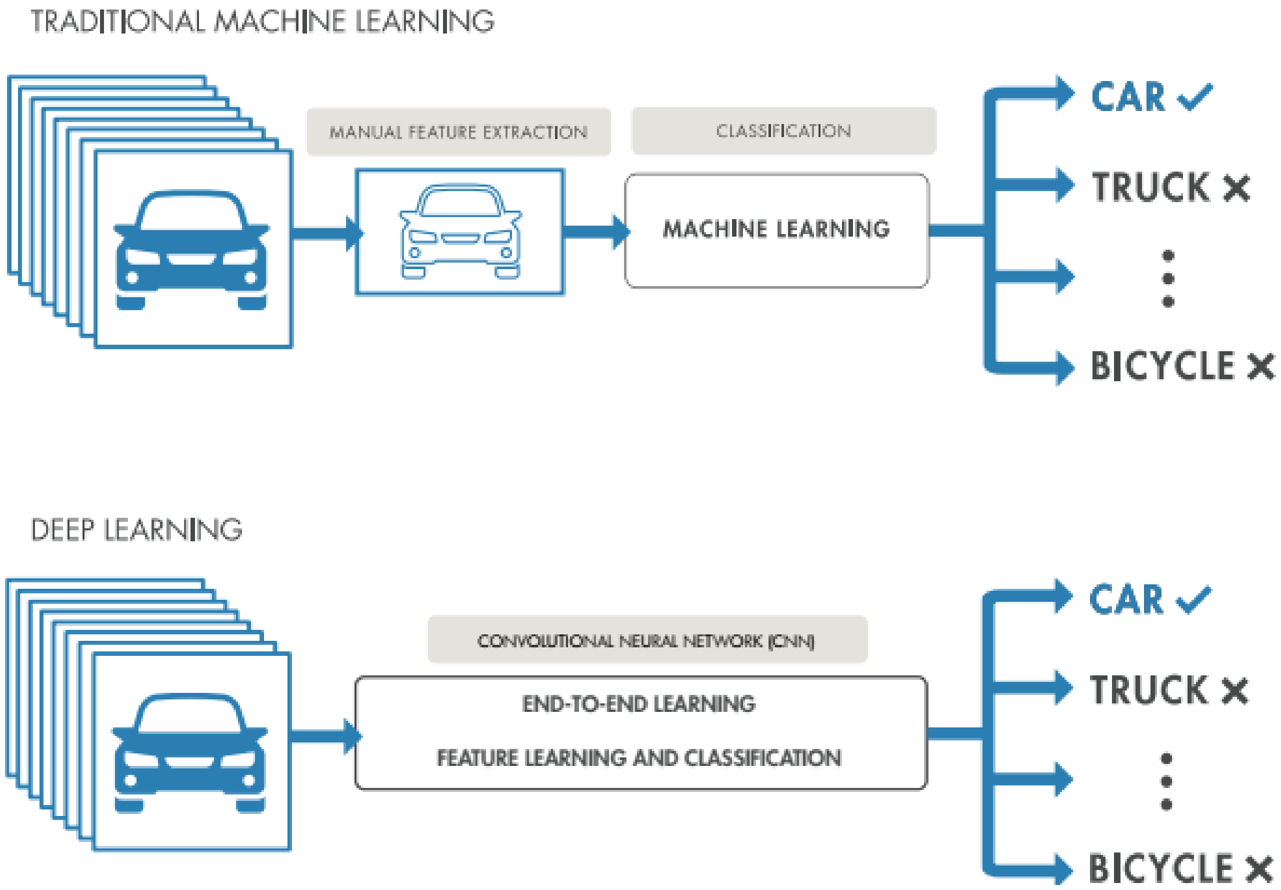


UNet Architecture



ML vs DL

3D Slicer + SOFA



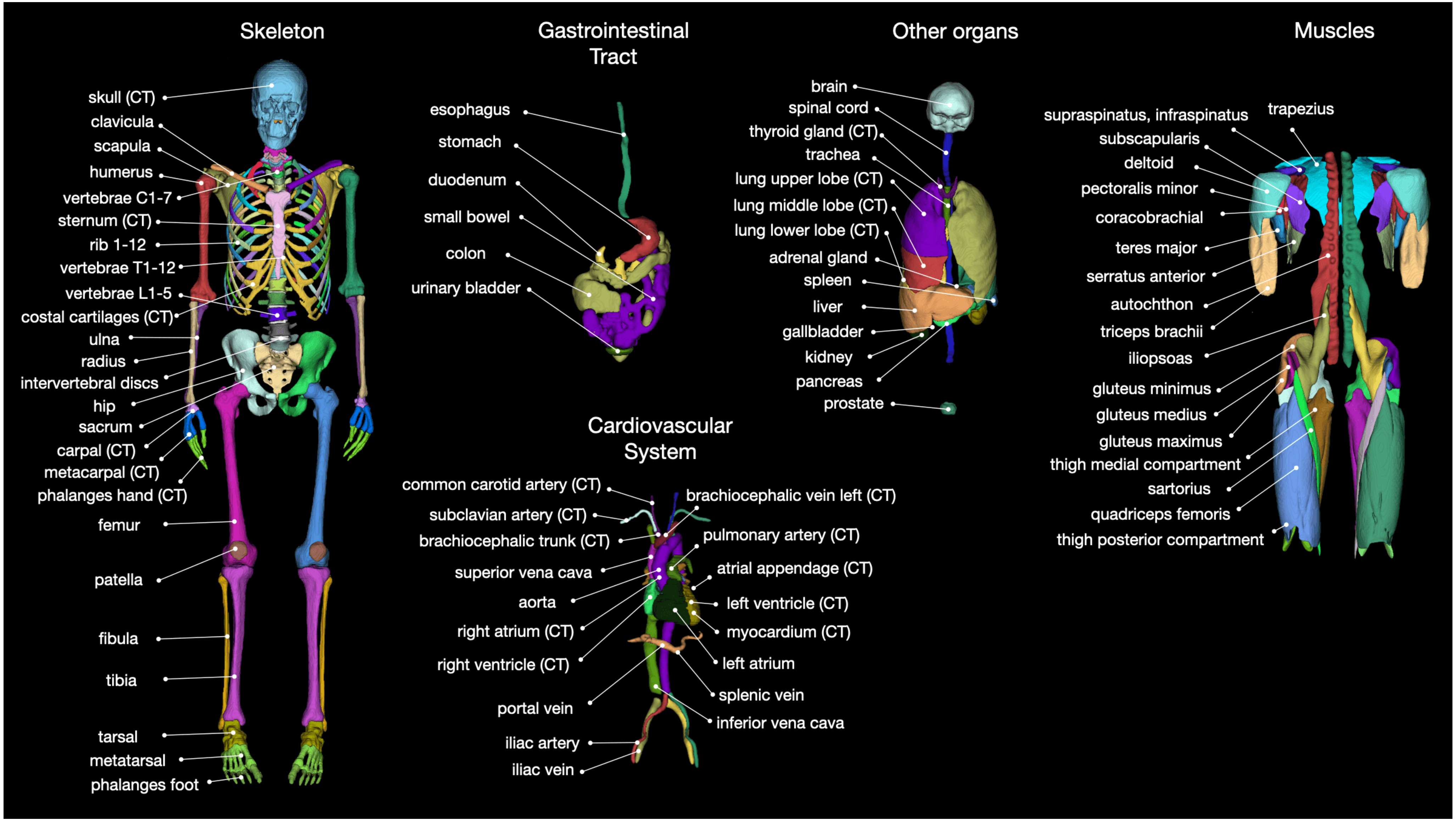
Machine Learning	Deep Learning
Good results with small data sets	Requires large data sets
Quick to train a model	Computationally intensive
Need to try different features and classifiers to obtain best results	Learns features and classifiers automatically

Total Segmentator

Total Segmentator

- Model: nnU-Net (basically a 3D U-Net that automatically configures many parameters based on your data, GPU, requirements, …)
- It segments:
  - 117 classes in CT images
  - 50 classes in MR images
- Trained on a wide range of different CT and MR images (different scanners, institutions, protocols,...) and therefore should work well on most images.
- A large part of the training dataset can be downloaded from Zenodo (1228 subjects for CT, and 616 subjects for MR).
- Available extension for 3D Slicer.
- You can also try the tool online at totalsegmentator.com

Total Segmentator



Pipeline of AI Segmentation

Pipeline of AI Segmentation

<u>Objective</u> : DL model to segment automatically & efficiently all OAR on CT

Pipeline of AI Segmentation

Objective : DL model to segment automatically & efficiently all OAR on CT

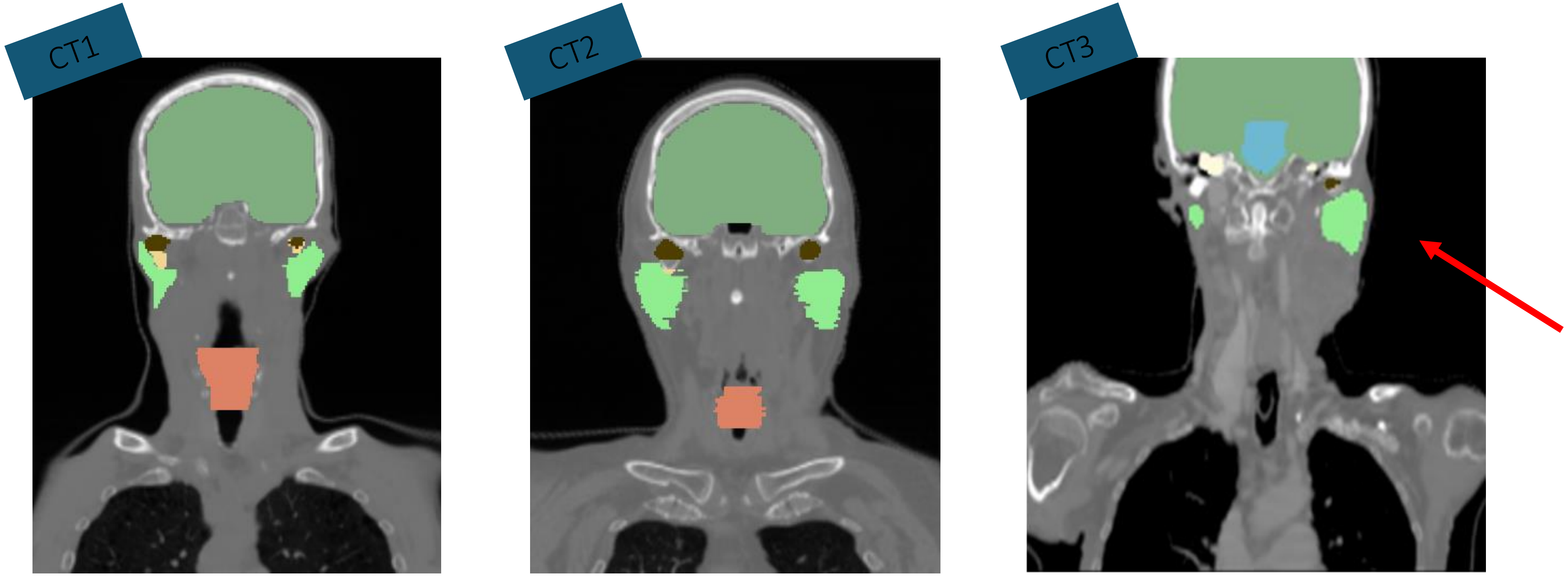
Step 1. Create a database to train and test the model

- It should be heterogeneous : sex, race, age, tumor type / size / location, multiple medical centers
- Challenges with ground truth (manual) labels:
  - Different segmentation protocols
  - Inter- and intra-observer variability
  - Complicated due to poor contrast
  - Tumors disrupt “normal” anatomy
- Number of patients : it is complicated to gather a large database due to medical data access limitations & accurate labeling

Pipeline of AI Segmentation

Objective : DL model to segment automatically & efficiently all OAR on CT

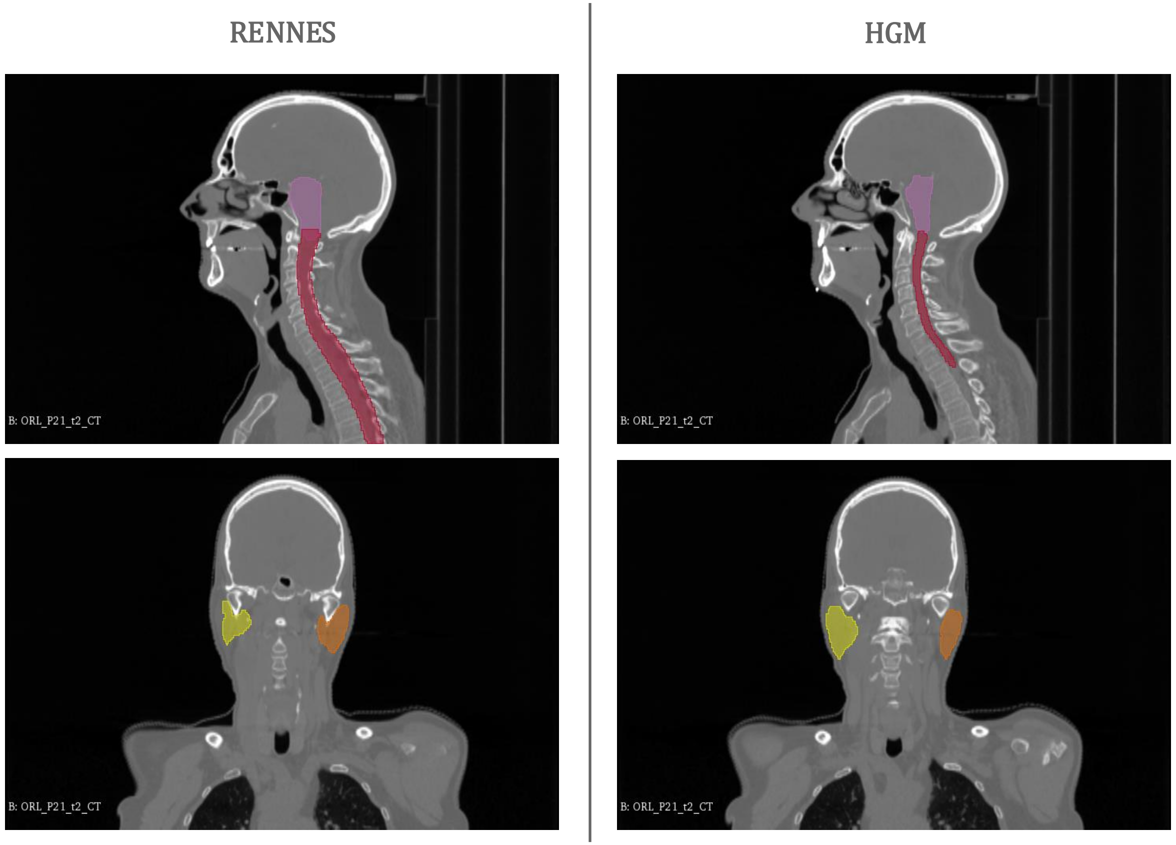
Step 1. Create a database to train and test the model



Pipeline of AI Segmentation

Objective : DL model to segment automatically & efficiently all OAR on CT

Step 1. Create a database to train and test the model



Pipeline of AI Segmentation

3D Slicer + SOFA

Objective : DL model to segment automatically & efficiently all OAR on CT

Step 2. Preprocessing

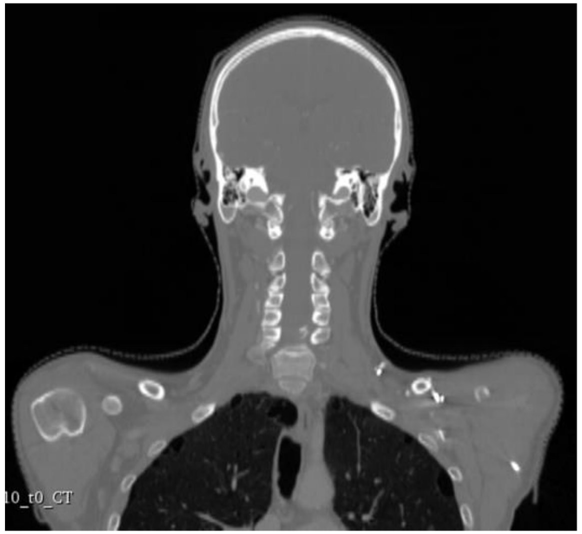
- If CT images have been acquired in different machines → they have different resolution, size, quality, … → IMAGE HOMOGENIZATION
- Ground truth segmentation curation is needed to ensure LABEL HOMOGENIZATION
- Larger images take larger training times, and most of the GPUs don't have enough memory → Downsample images (with an unavoidable loss of resolution)
- Data augmentation : rotation, random crops, random brightness shifts, noise

Pipeline of AI Segmentation

3D Slicer + SOFA

Objective : DL model to segment automatically & efficiently all OAR on CT

Step 2. Preprocessing



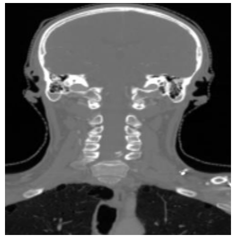
512 x 512 x 178

Cropping

158 x 190 x 214

Downsample

79 x 95 x 107

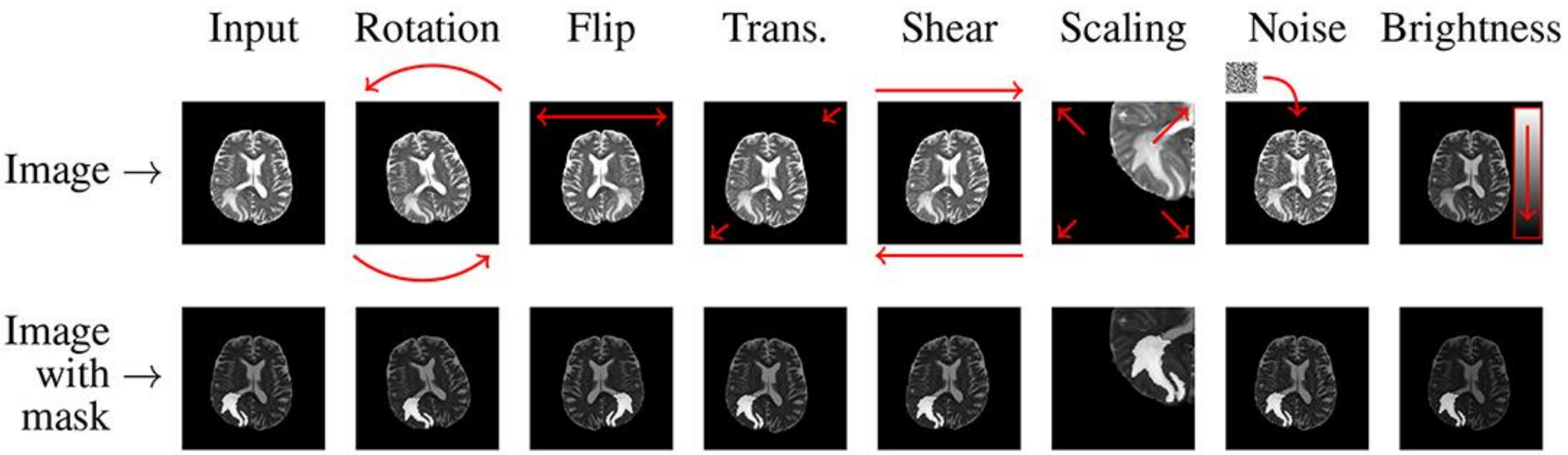


98 x 98 x 98

Pipeline of AI Segmentation

Objective : DL model to segment automatically & efficiently all OAR on CT

Step 2. Preprocessing



Pipeline of AI Segmentation

Objective : DL model to segment automatically & efficiently all OAR on CT

Step 3. Training

- Model architecture: U-Net, nnU-Net, DenseVNet, SwinUNETR
- Loss function: Dice, Cross-Entropy
- Hyperparameter tuning
- 5-fold cross-validation → Longer training times

Pipeline of AI Segmentation

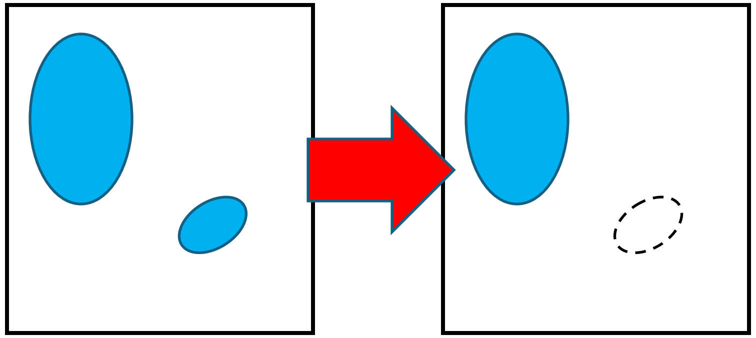
3D Slicer SOFA

Objective : DL model to segment automatically & efficiently all OAR on CT

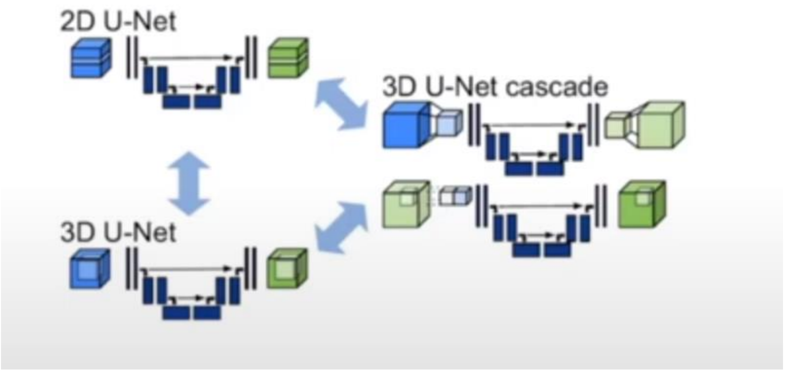
Step 4. Evaluation

- Post-processing

Postprocessing
Suppression of non-maximum components. Apply or not?



Ensembling
Which model (or combination of models) to choose from trained 2D, 3D and Cascade networks



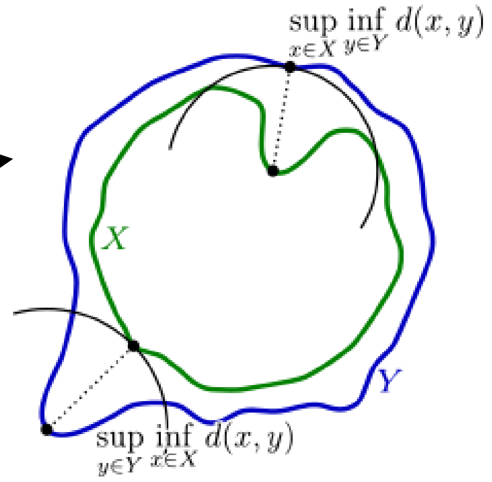
Pipeline of AI Segmentation

Objective : DL model to segment automatically & efficiently all OAR on CT

Step 4. Evaluation

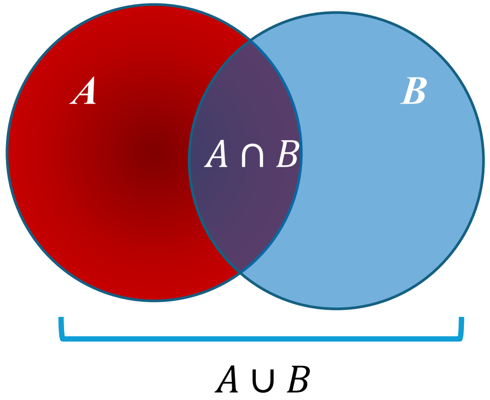
- Evaluation metrics
  - Geometrical: Dice Score Coefficient, Jaccard Index
  - Distance: Average Surface Distance, Hausdorff Distance
  - Clinical Acceptability

$$d_H(S_1, S_2) = max\left\{ \max_{s_2 \in S_2} \min_{s_1 \in S_1} d(s_1, s_2), \max_{s_1 \in S_1} \min_{s_2 \in S_2} d(s_2, s_1) \right\}$$



$$D = \frac{2|A \cap B|}{|A| + |B|}$$

$$J = \frac{|A \cap B|}{|A \cup B|}$$



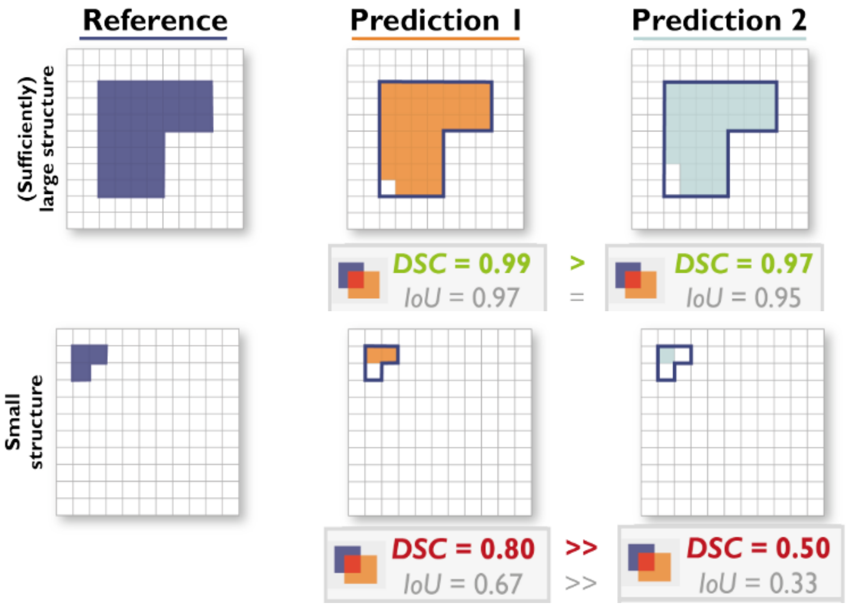
$$J = \frac{D}{(2 - D)} \qquad D = \frac{2J}{(1 + J)}$$

Pipeline of AI Segmentation

Objective : DL model to segment automatically & efficiently all OAR on CT

Step 4. Evaluation

- Evaluation metrics
  - Geometrical: Dice Score Coefficient, Jaccard Index
  - Distance: Average Surface Distance, Hausdorff Distance
  - Clinical Acceptability



AI Segmentation in 3D Slicer

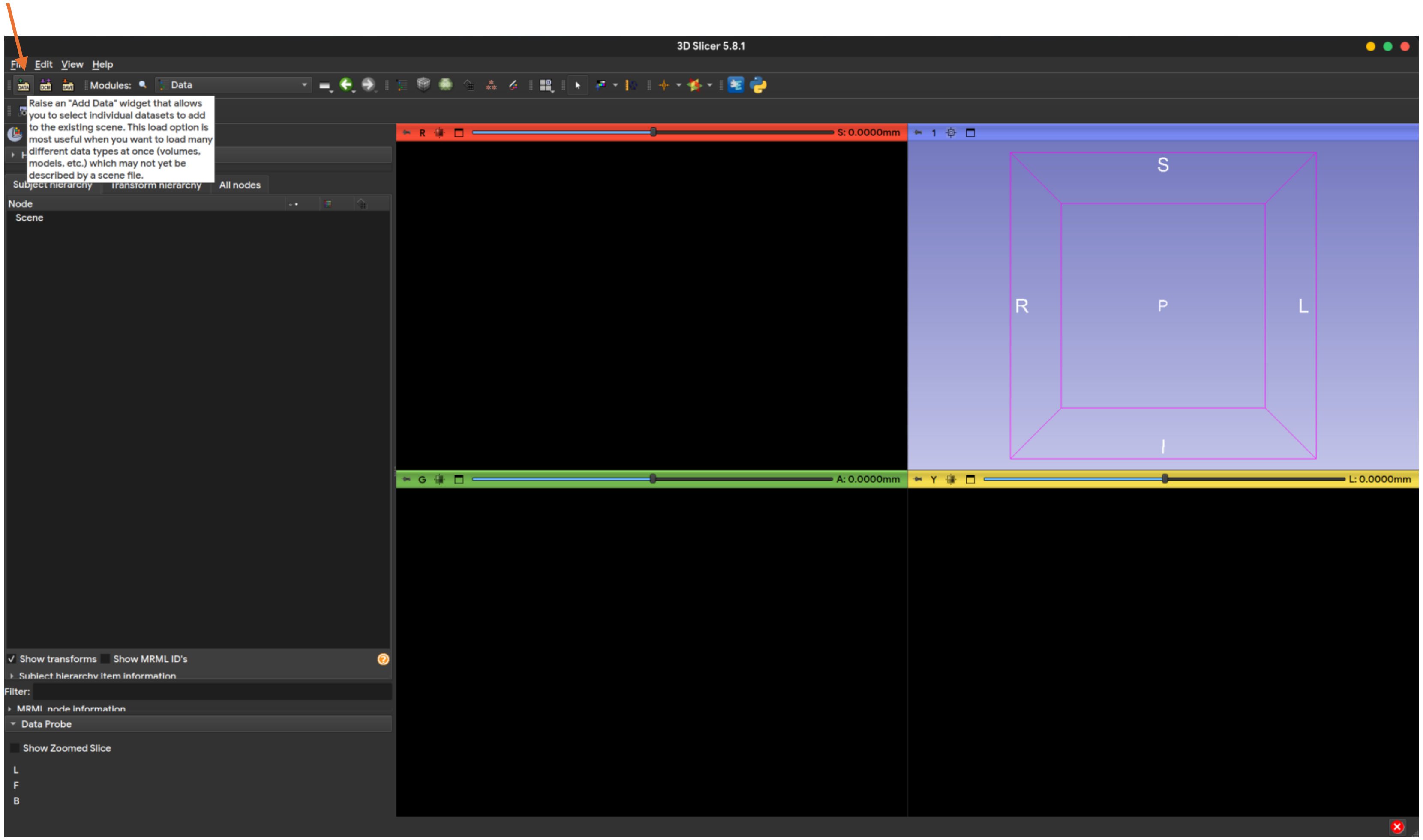
State of AI Segmentation in 3D Slicer

State of AI Segmentation in 3D Slicer

- AI Segmentation in 3D Slicer is messy because...
  - There are **no native** extensions
  - Extensions are just **wrappers** of deep learning models
  - Each extension/model has their own libraries/version of libraries
  - Integration with the GUI is sloppy (nnInteractive for example)

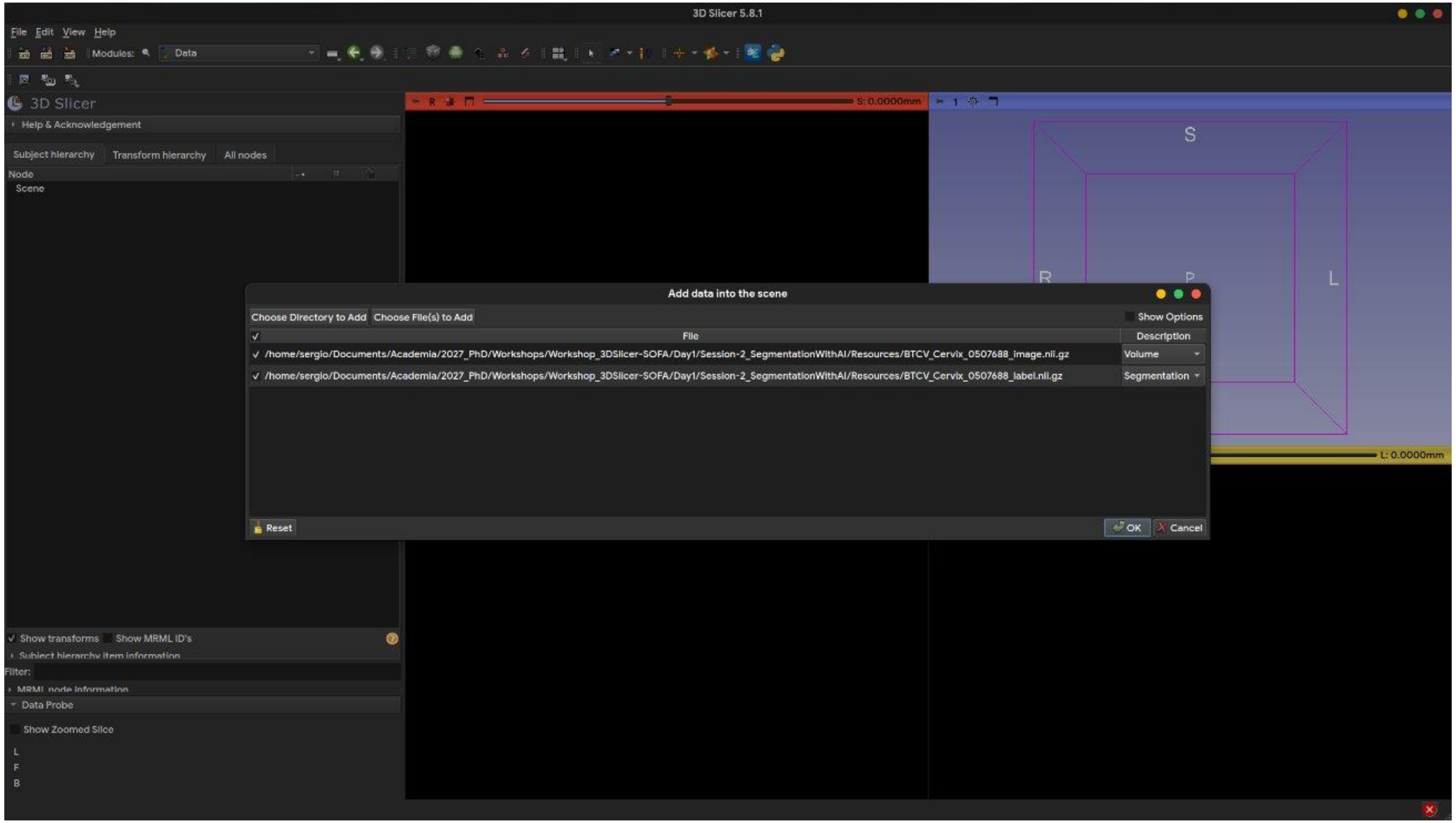
Classic Segmentation in 3D Slicer

- Loading data



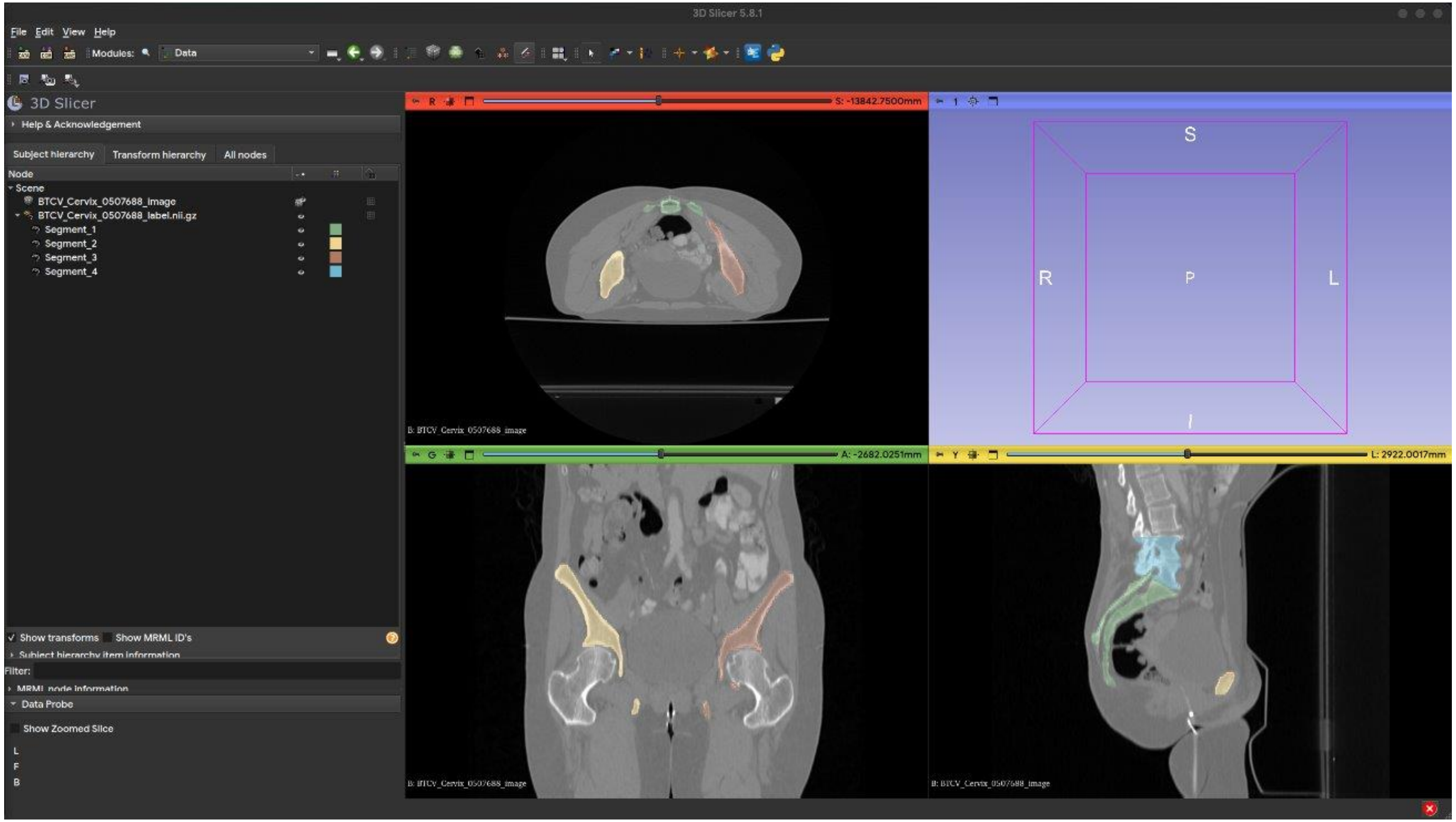
Classic Segmentation in 3D Slicer

- Loading data
  - Select the image as "Volume" and the label as "Segmentation"



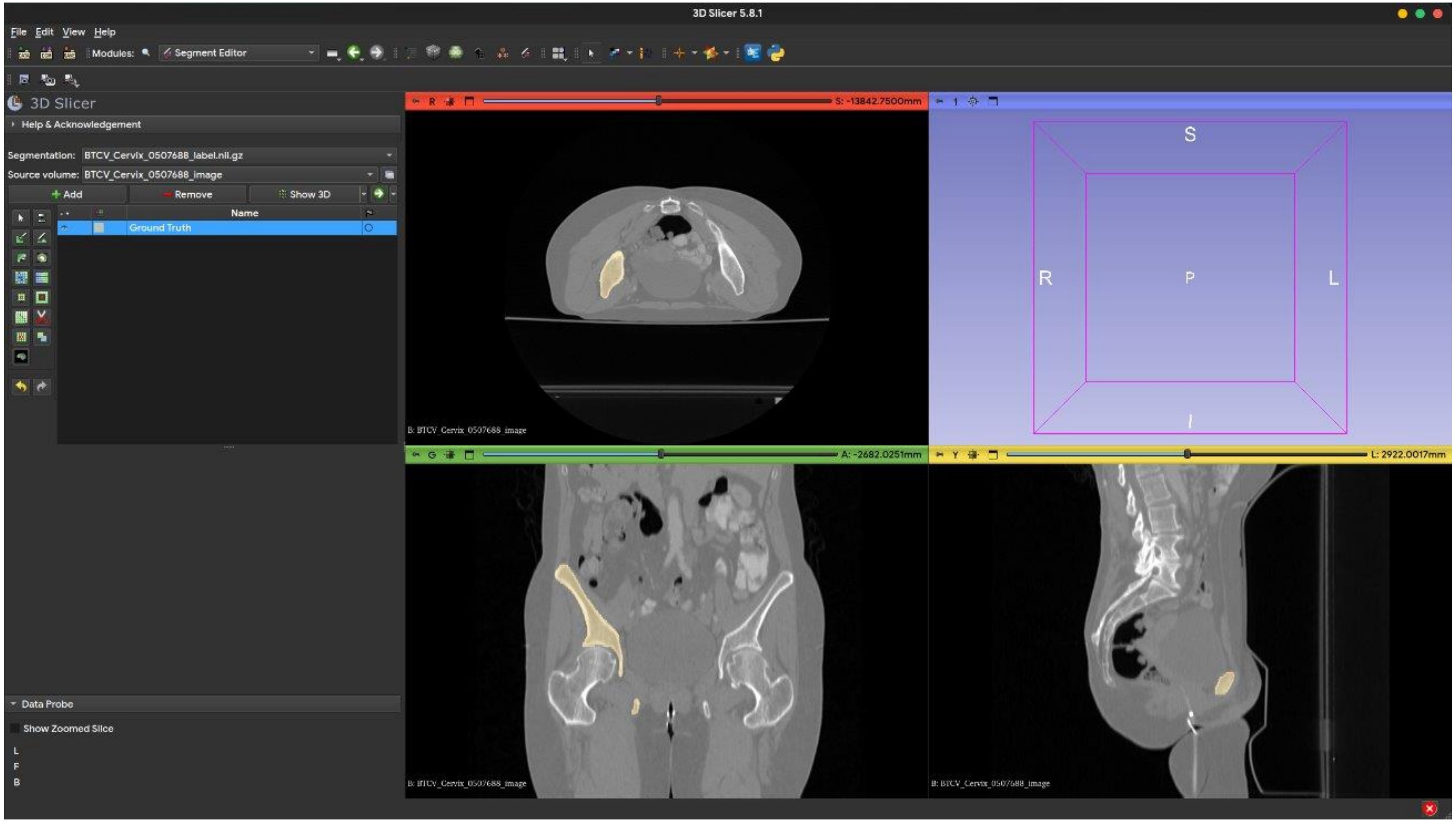
Classic Segmentation in 3D Slicer

- Loading data



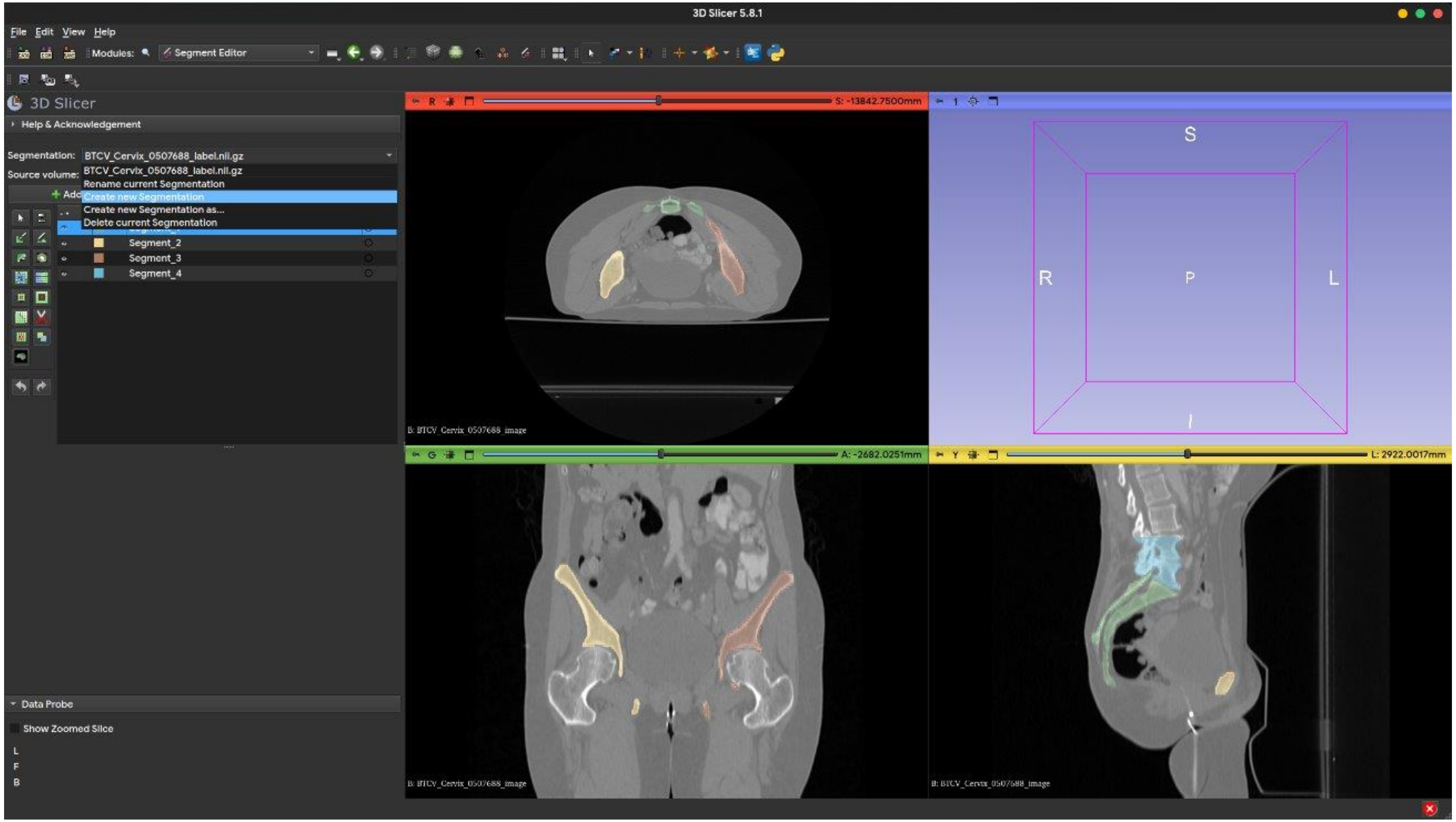
Classic Segmentation in 3D Slicer

- Loading data
  - Let's remove all of them except label 2 (right hip), which we will rename



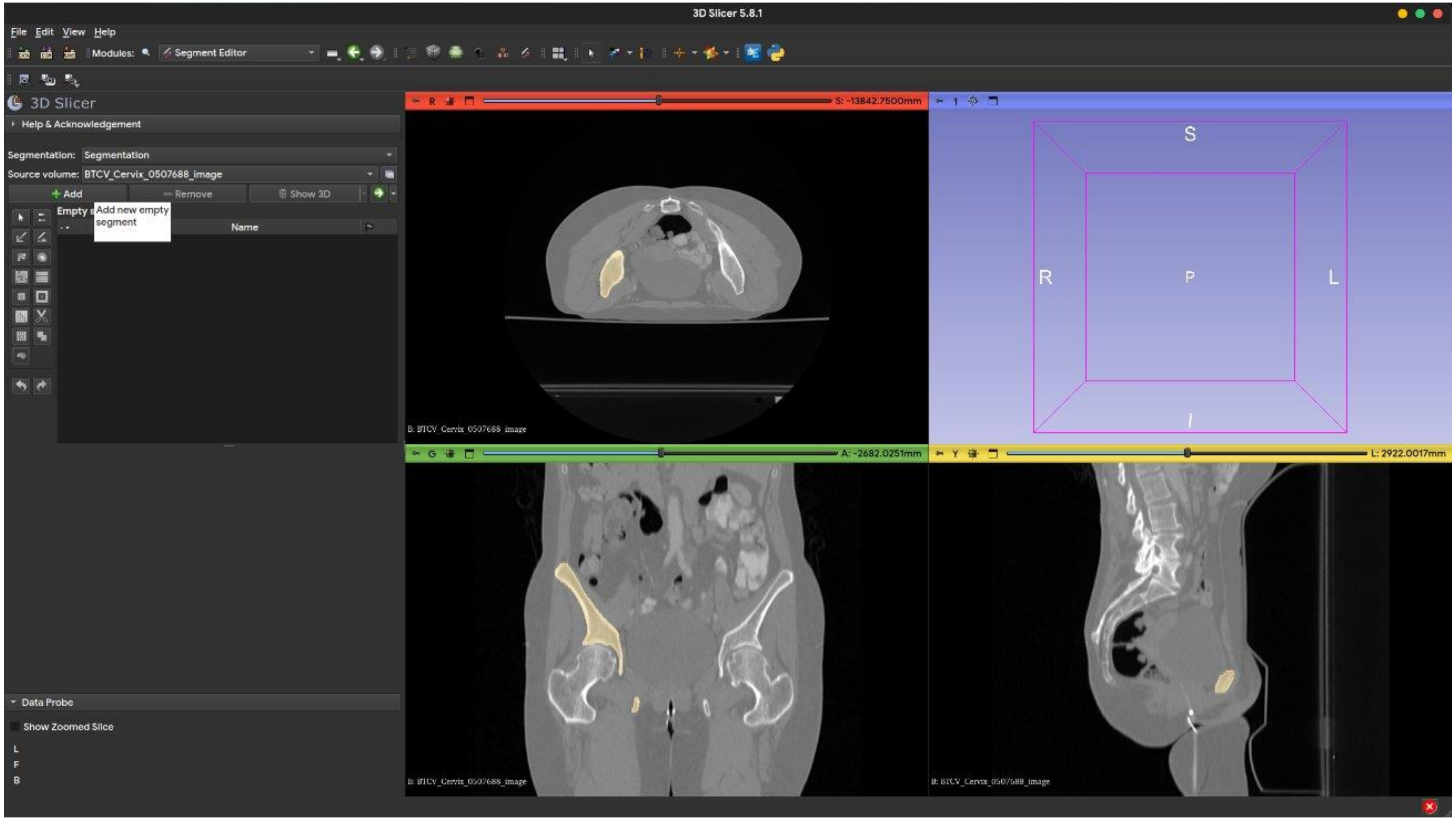
Classic Segmentation in 3D Slicer

- For every method we will create a new segmentation object



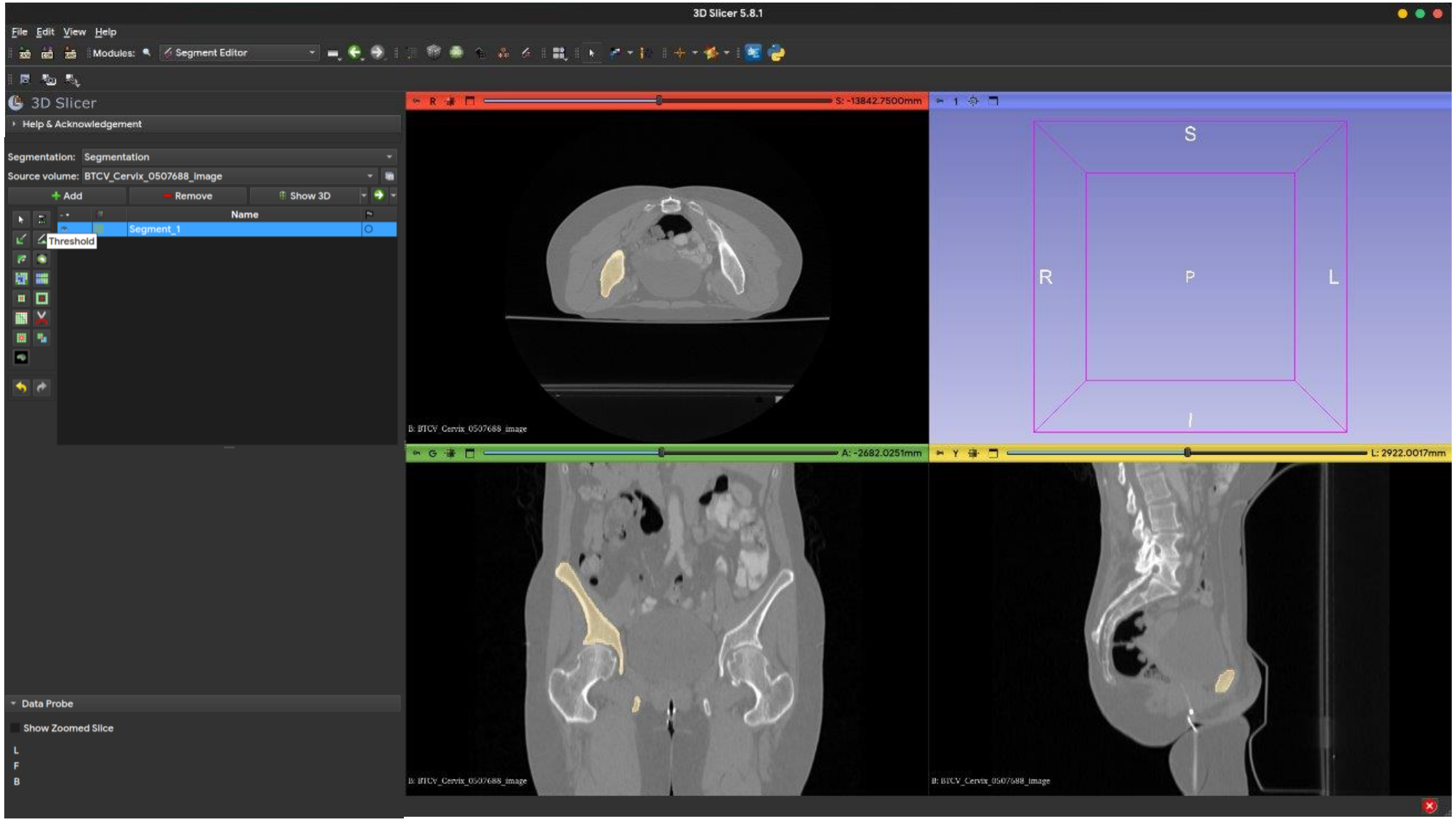
Classic Segmentation in 3D Slicer

- Thresholding
  - Let's add a new segment



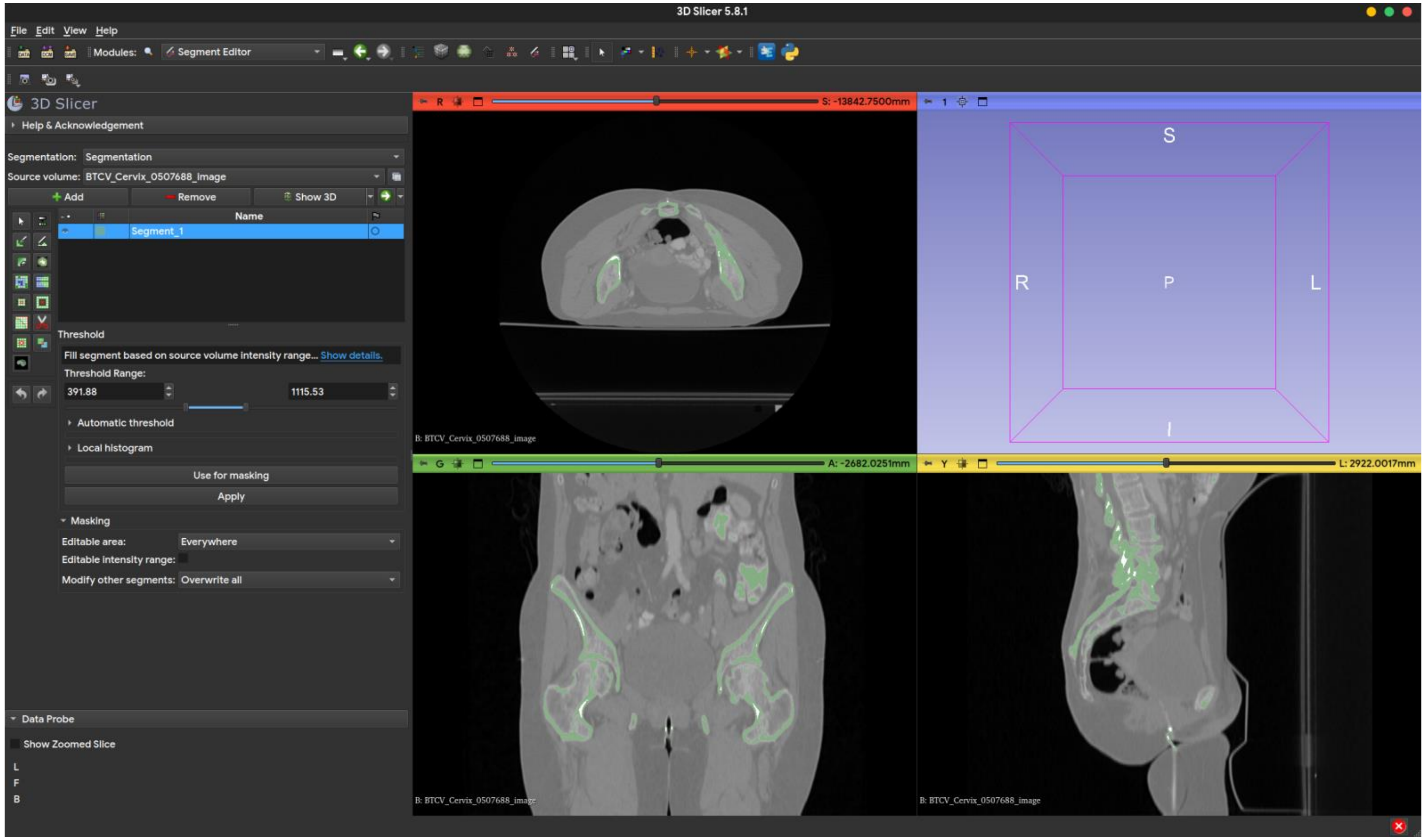
Classic Segmentation in 3D Slicer

- Thresholding
  - Choose Thresholding



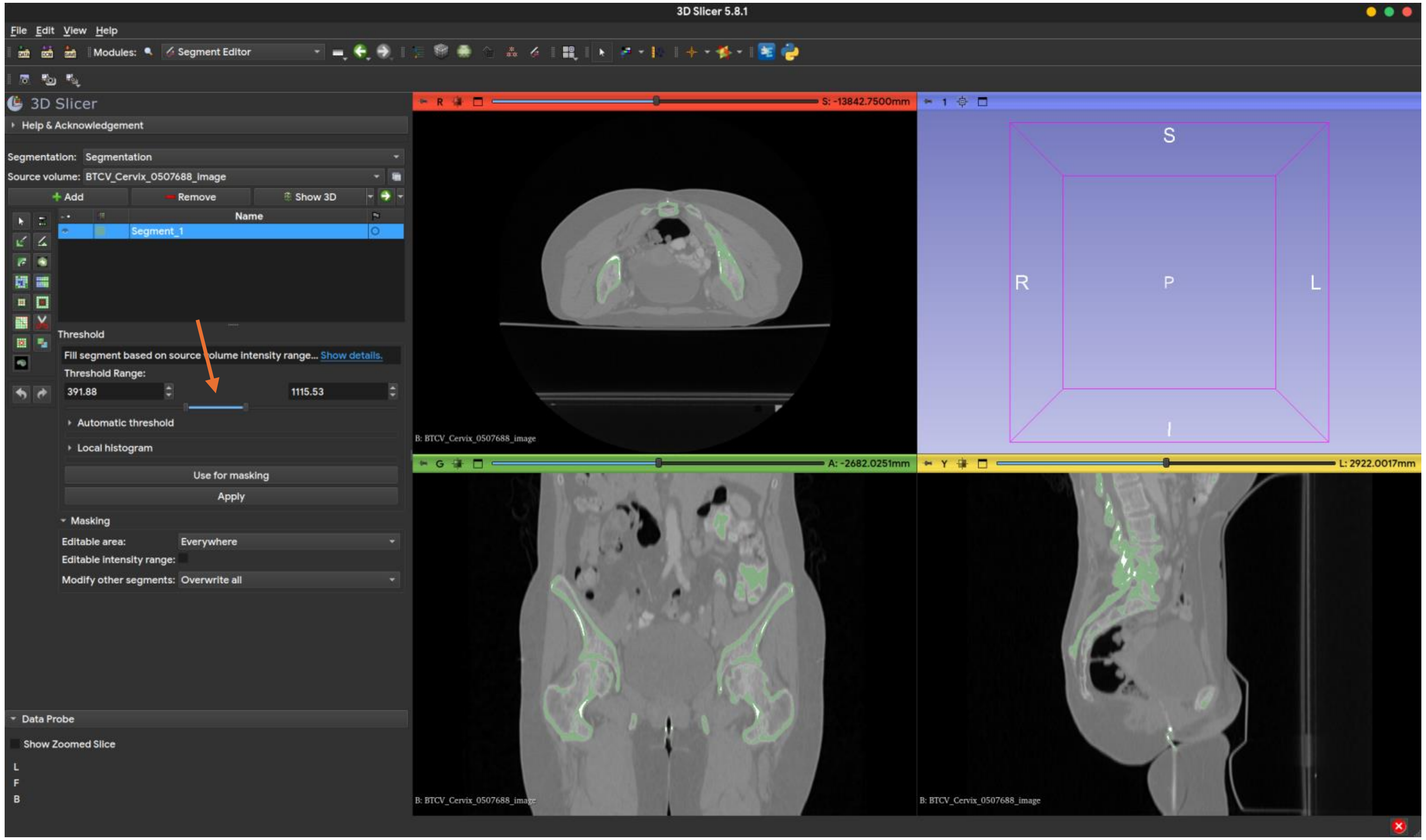
Classic Segmentation in 3D Slicer

- Thresholding
  - Select an adequate threshold and click apply



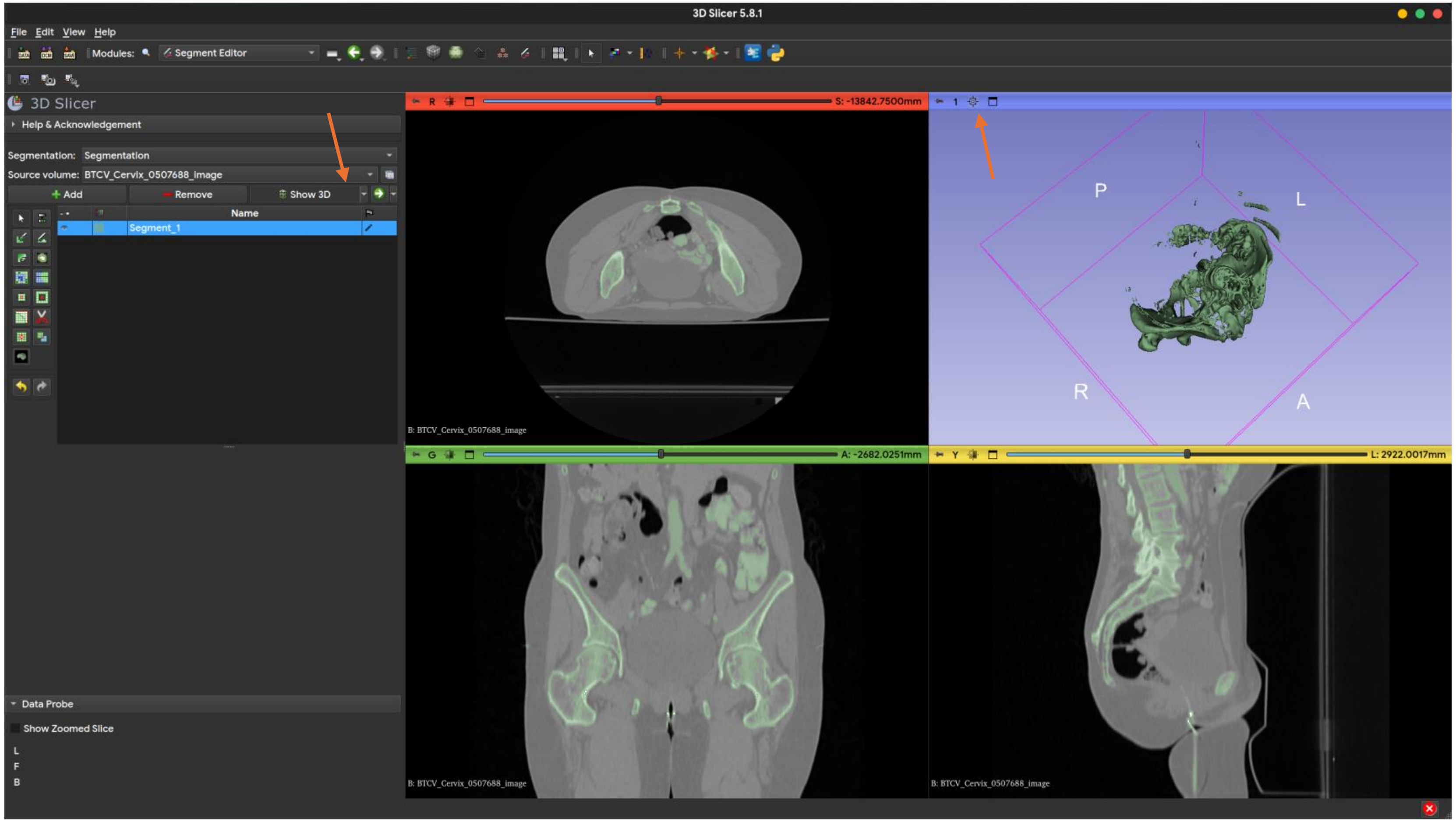
Classic Segmentation in 3D Slicer

- Thresholding
  - Select an adequate threshold and click apply



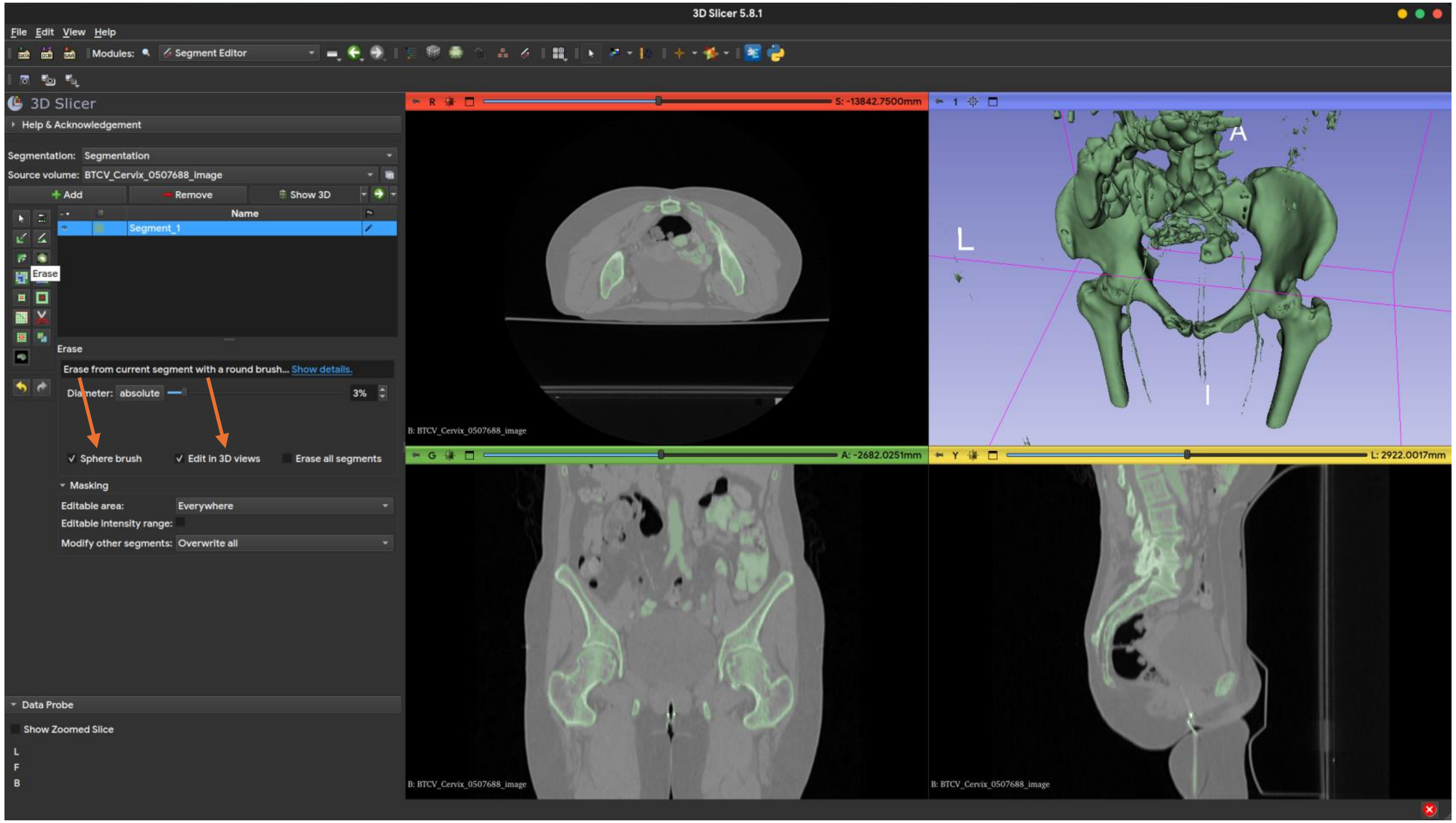
Classic Segmentation in 3D Slicer

- Thresholding
  - See the segmentation in 3D (center the view after clicking "Show 3D")



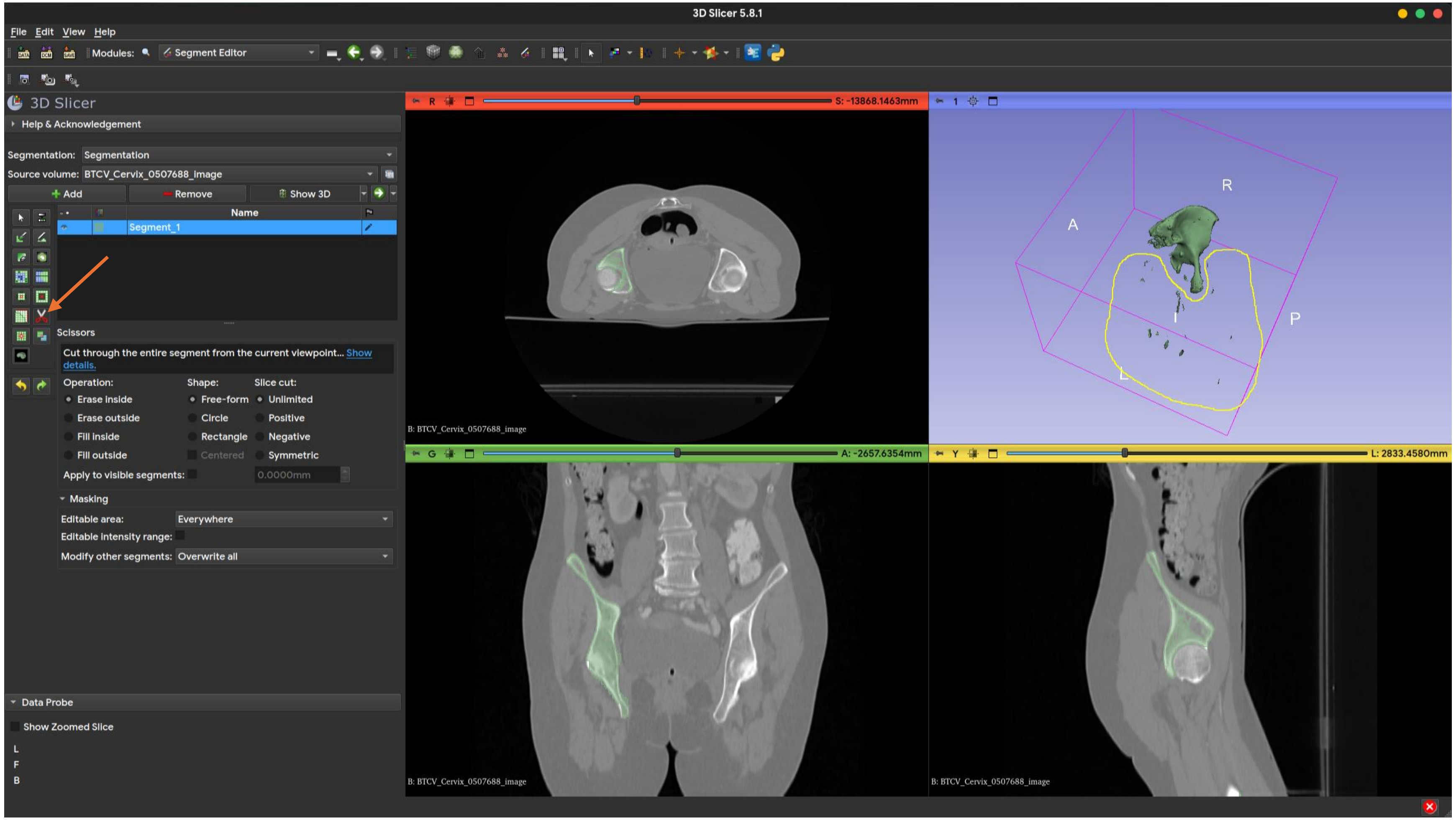
Classic Segmentation in 3D Slicer

- Refining segmentation
  - We can erase the parts we do not want



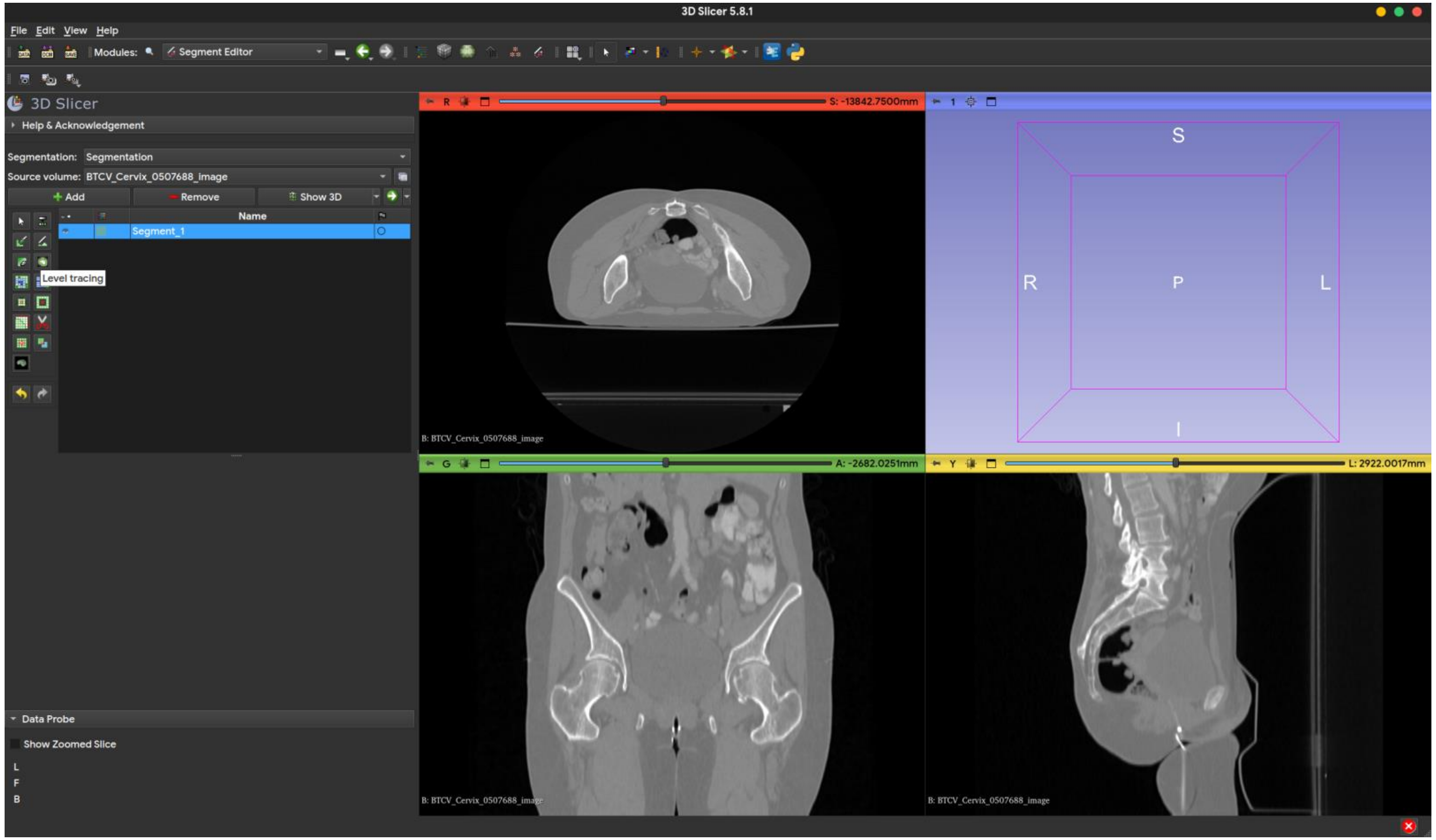
Classic Segmentation in 3D Slicer

- Refining segmentation
  - In the 3D view we can use the "Scissors" to remove chunks



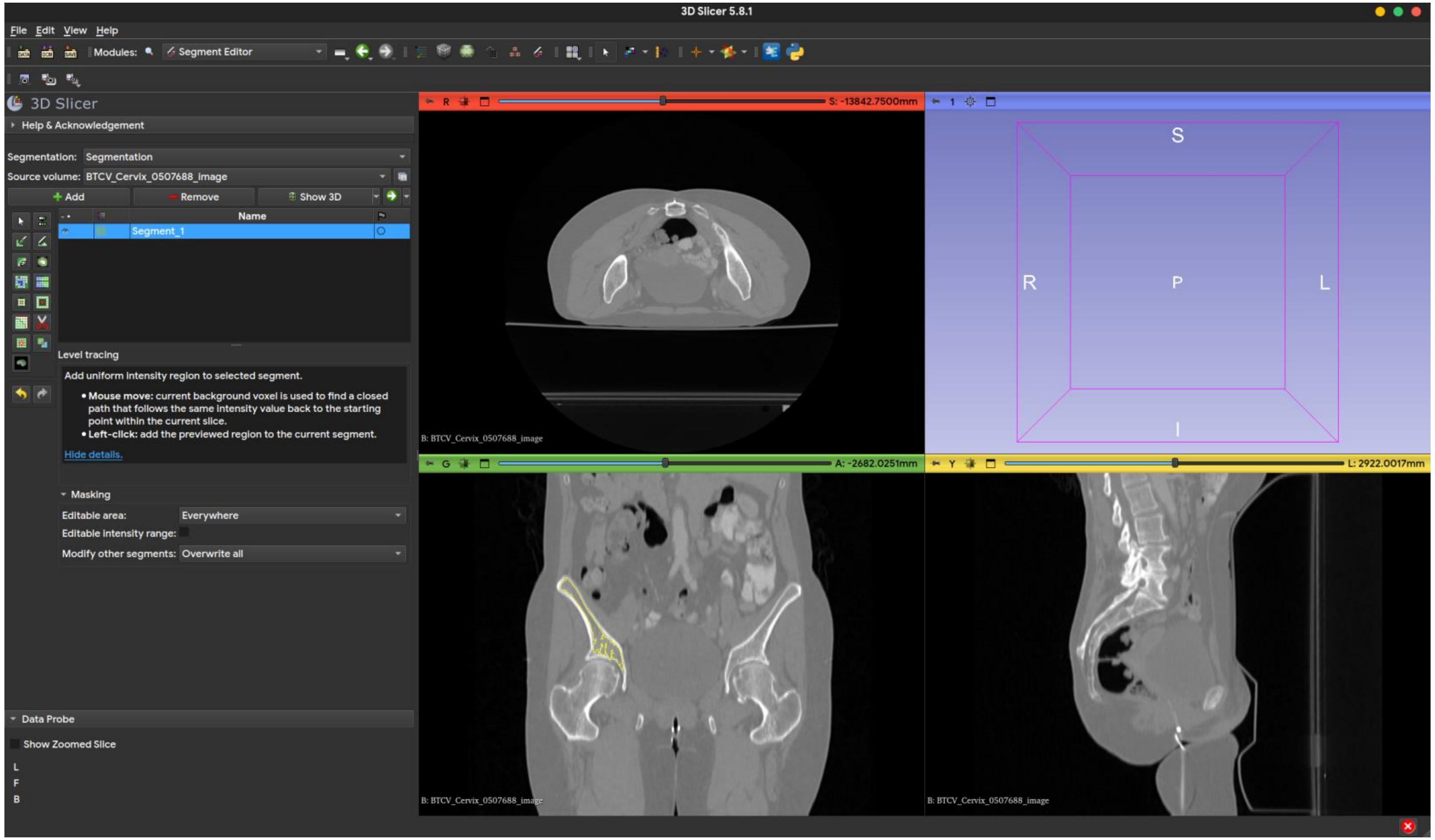
Classic Segmentation in 3D Slicer

- Level tracing
  - Simply put the mouse on top of the ROI and let the intensities create areas



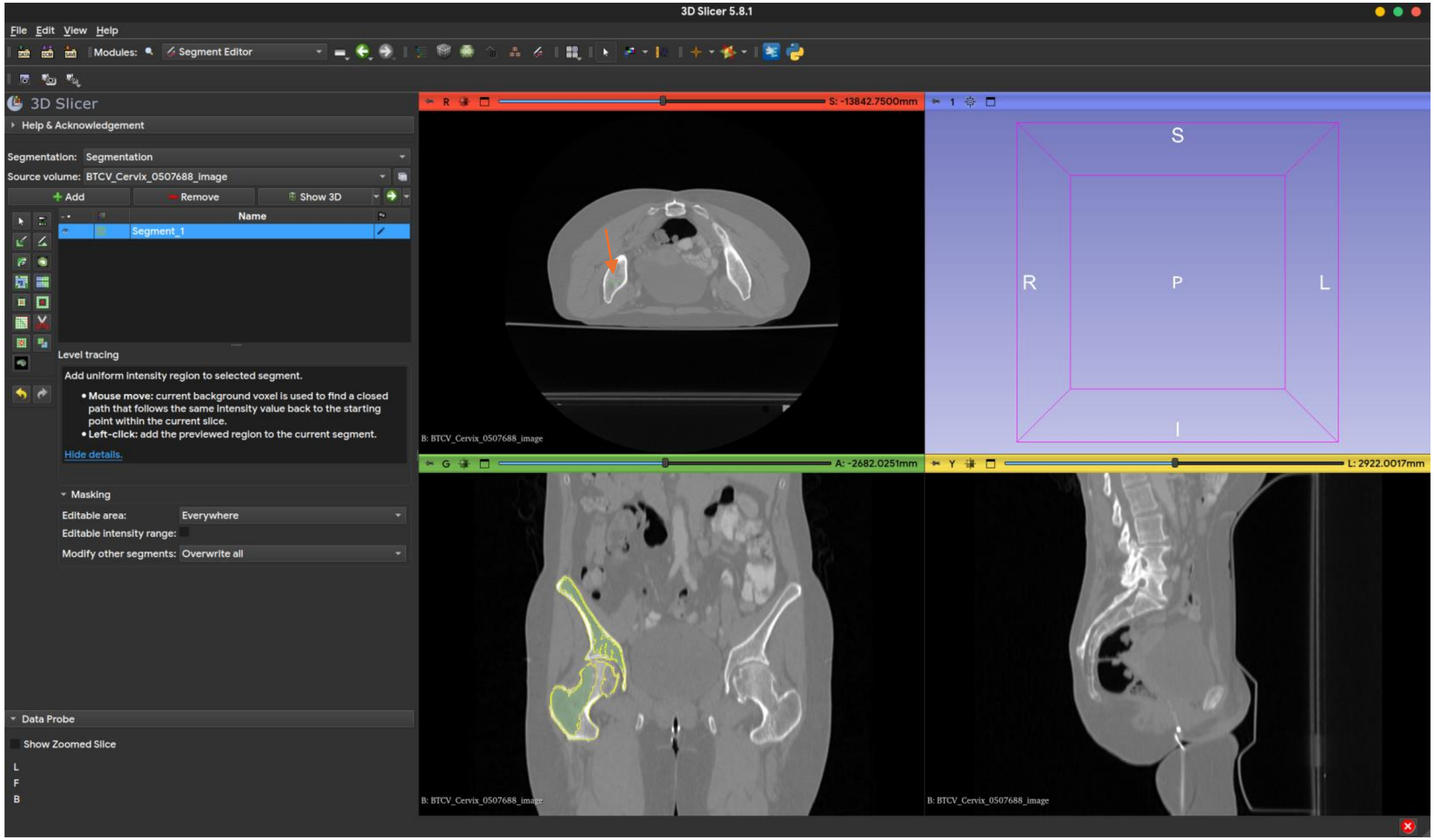
Classic Segmentation in 3D Slicer

- Level tracing
  - Simply put the mouse on top of the ROI and let the intensities create areas



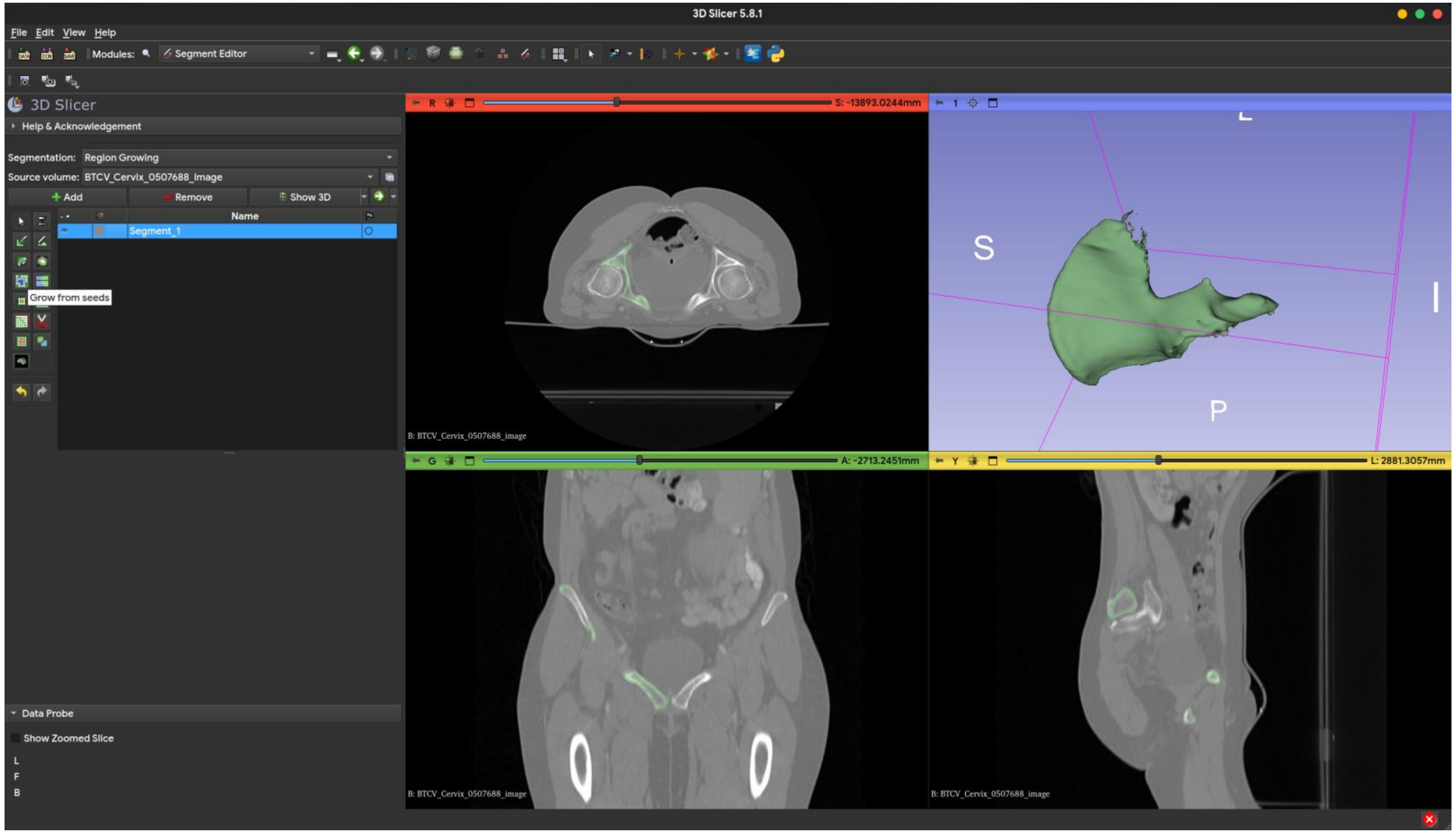
Classic Segmentation in 3D Slicer

- Level tracing
  - The main limitation is that this mode works only per slice



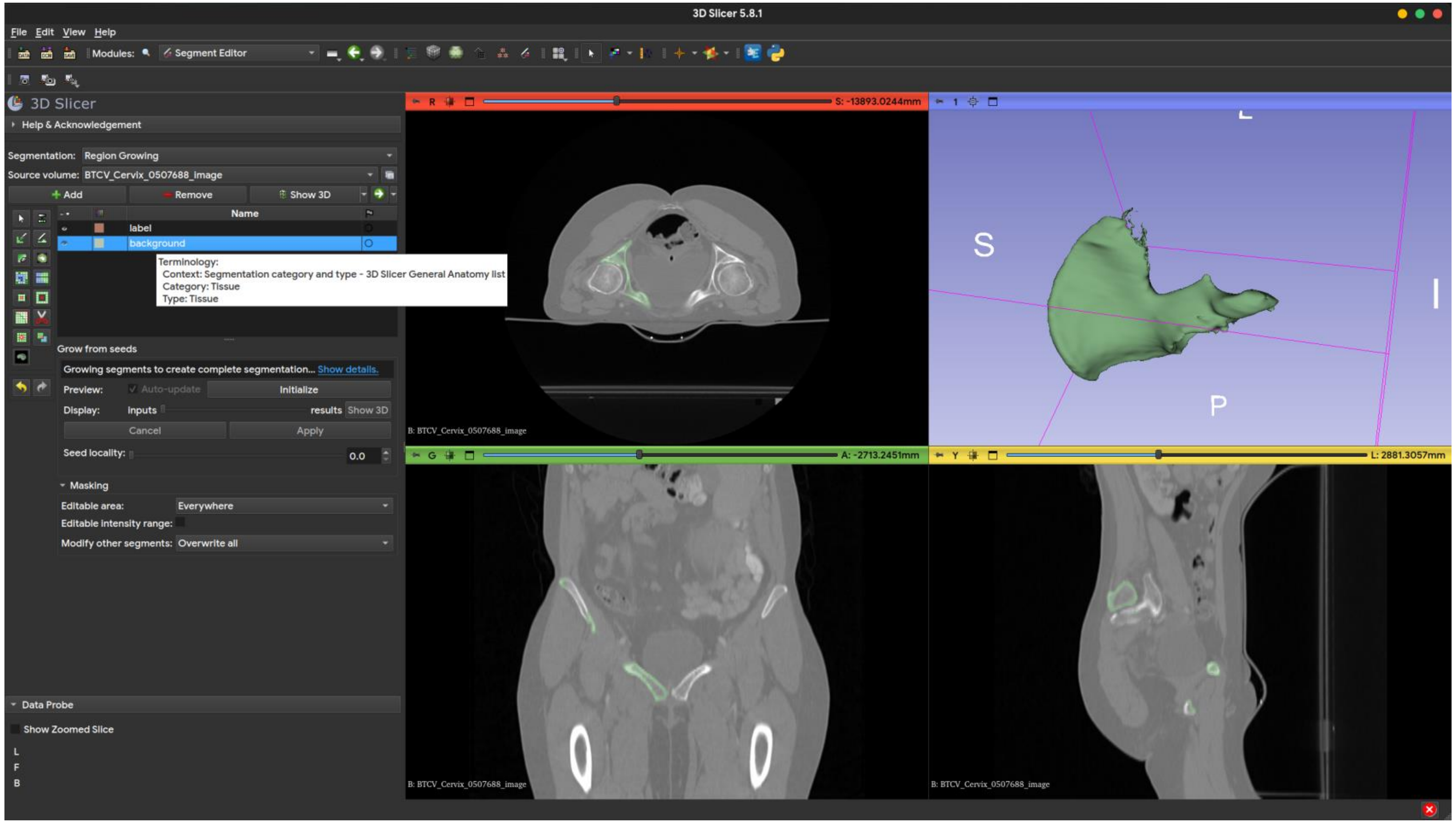
Classic Segmentation in 3D Slicer

- Region growing
  - After creating a new segmentation, create two segments



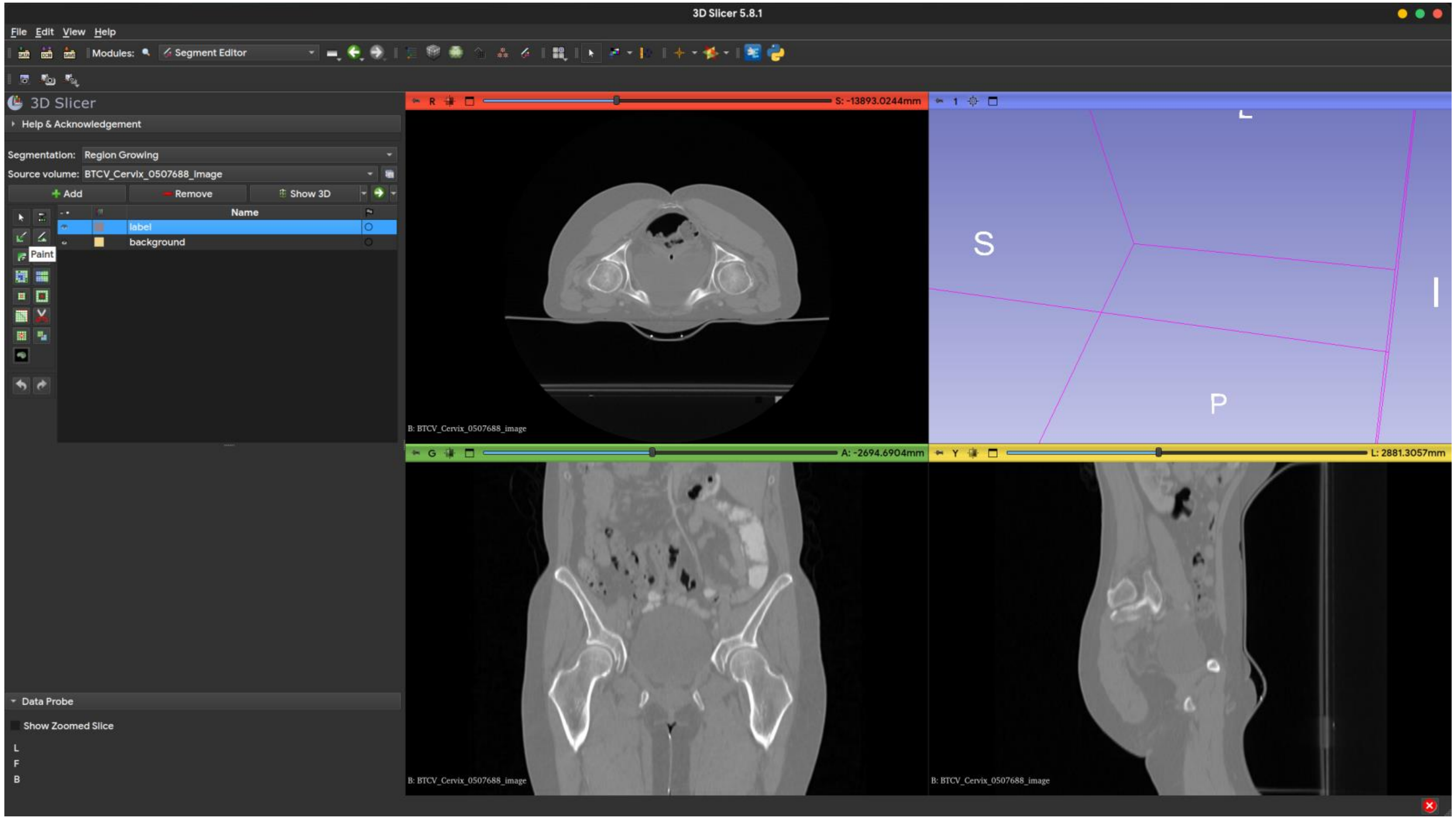
Classic Segmentation in 3D Slicer

- Region growing
  - After creating a new segmentation, create two segments



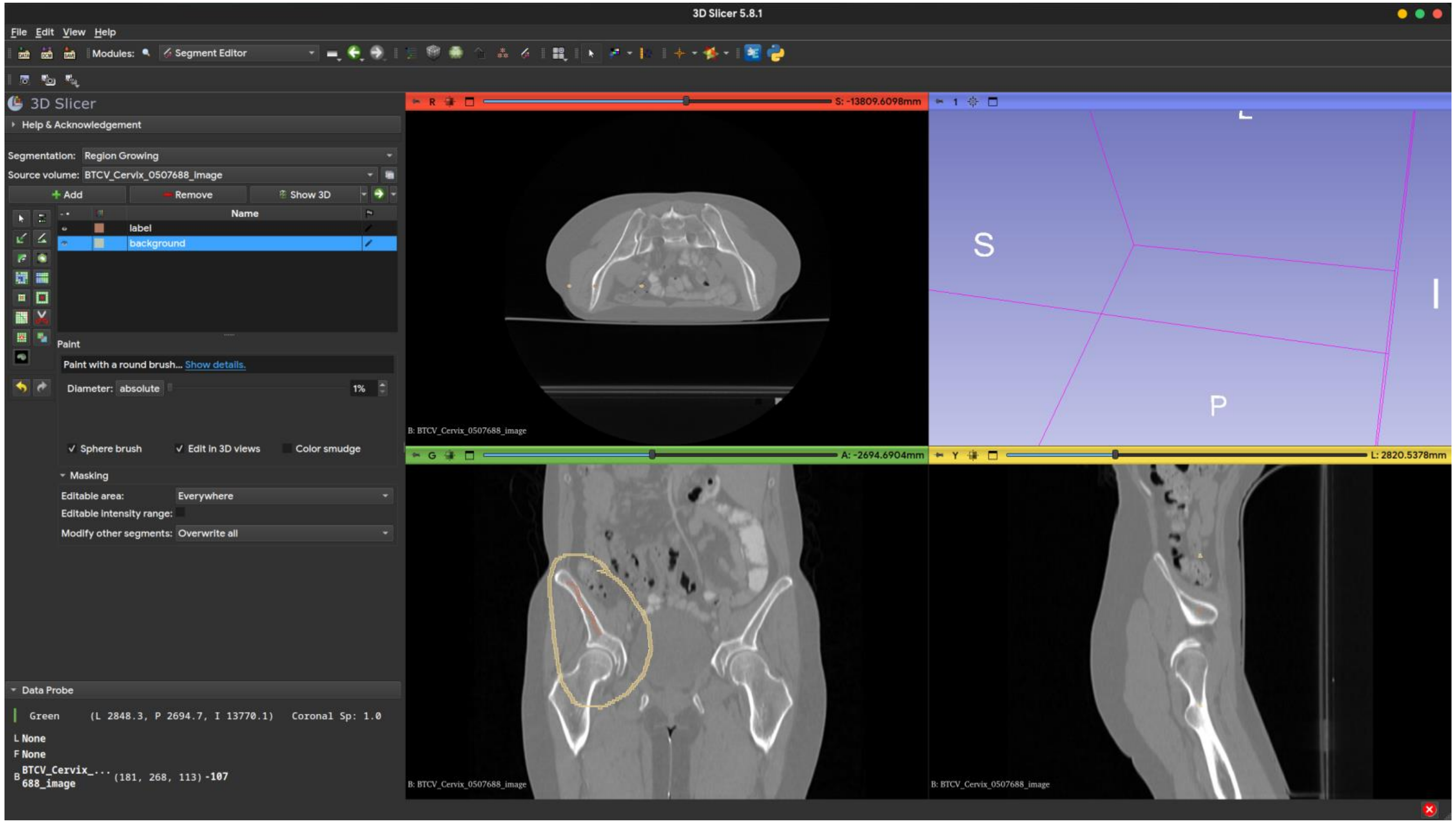
Classic Segmentation in 3D Slicer

- Region growing
  - Using the paintbrush, paint 'seeds' on the hip and background with each color



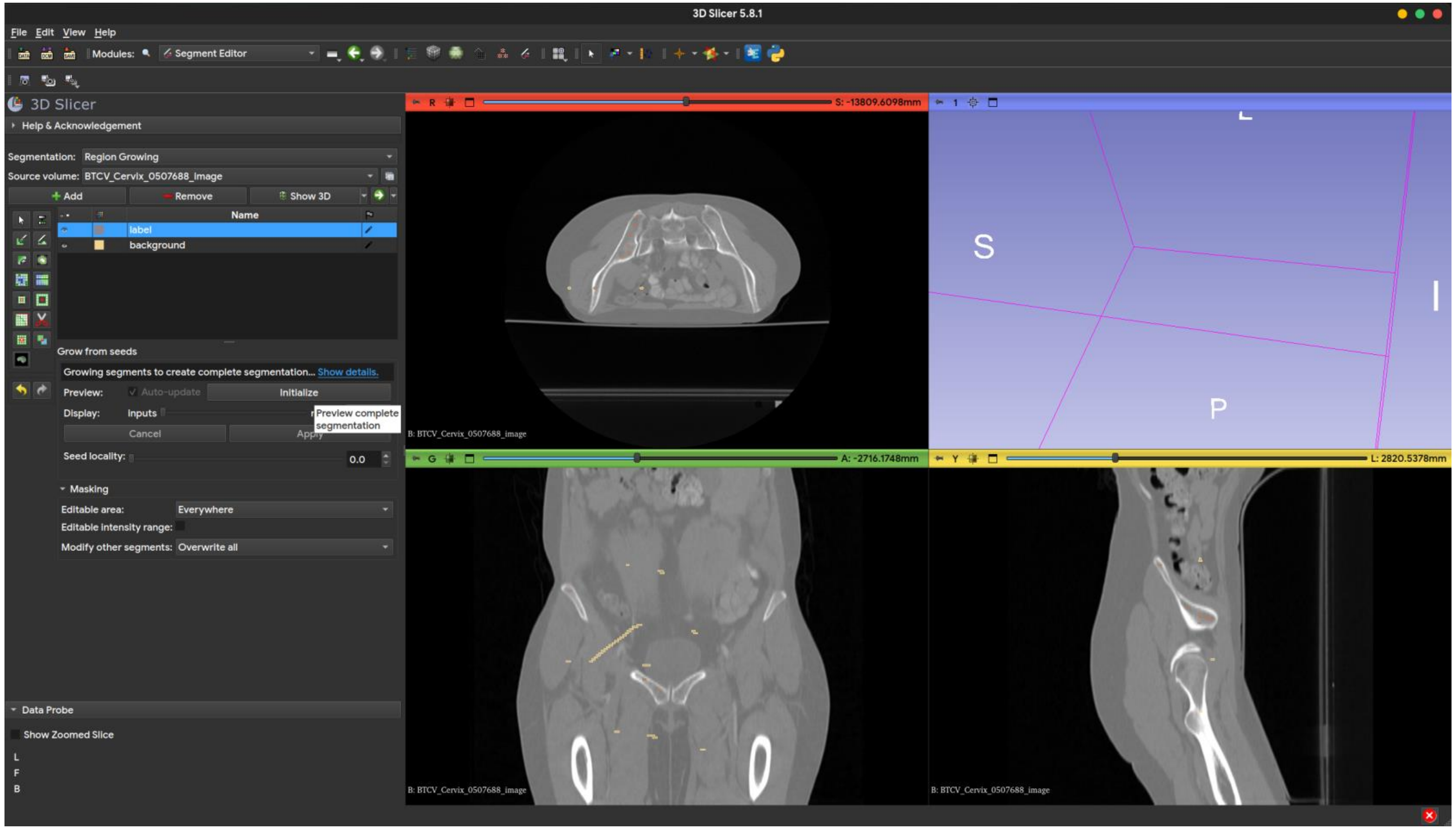
Classic Segmentation in 3D Slicer

- Region growing
  - Using the paintbrush, paint 'seeds' on the hip and background with each color



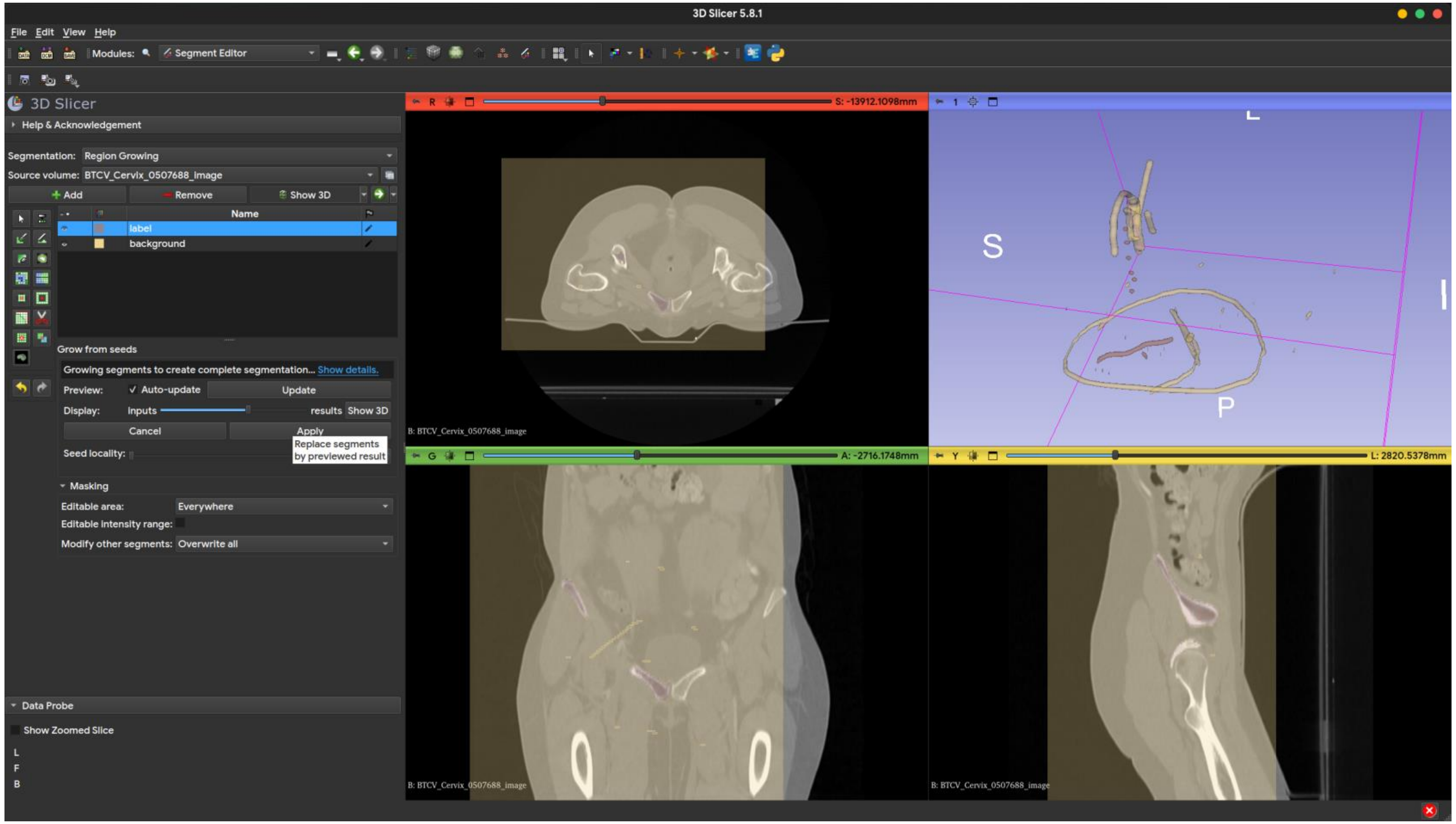
Classic Segmentation in 3D Slicer

- Region growing
  - Select "initialize" to see the preliminary results



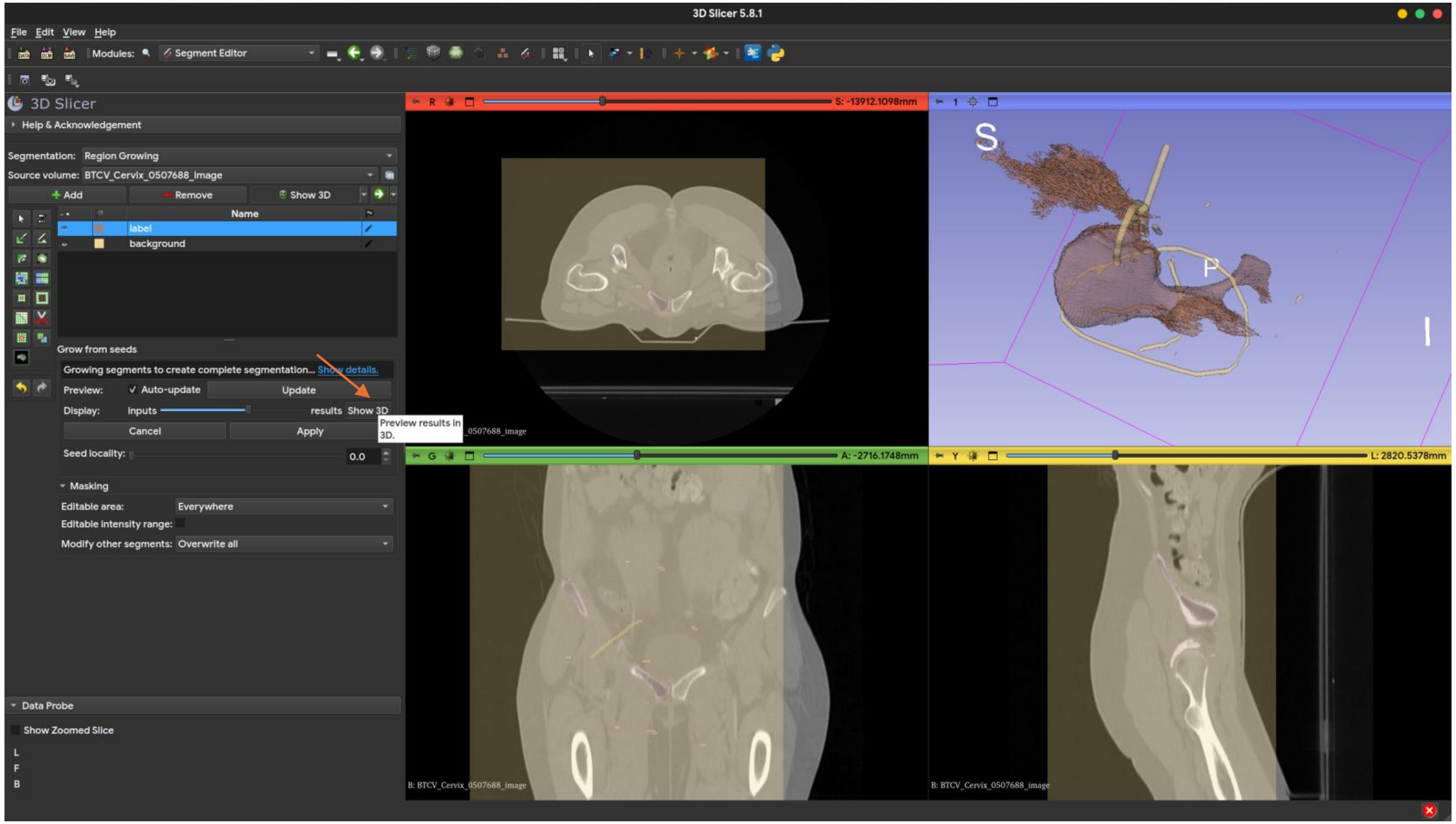
Classic Segmentation in 3D Slicer

- Region growing
  - Select "initialize" to see the preliminary results



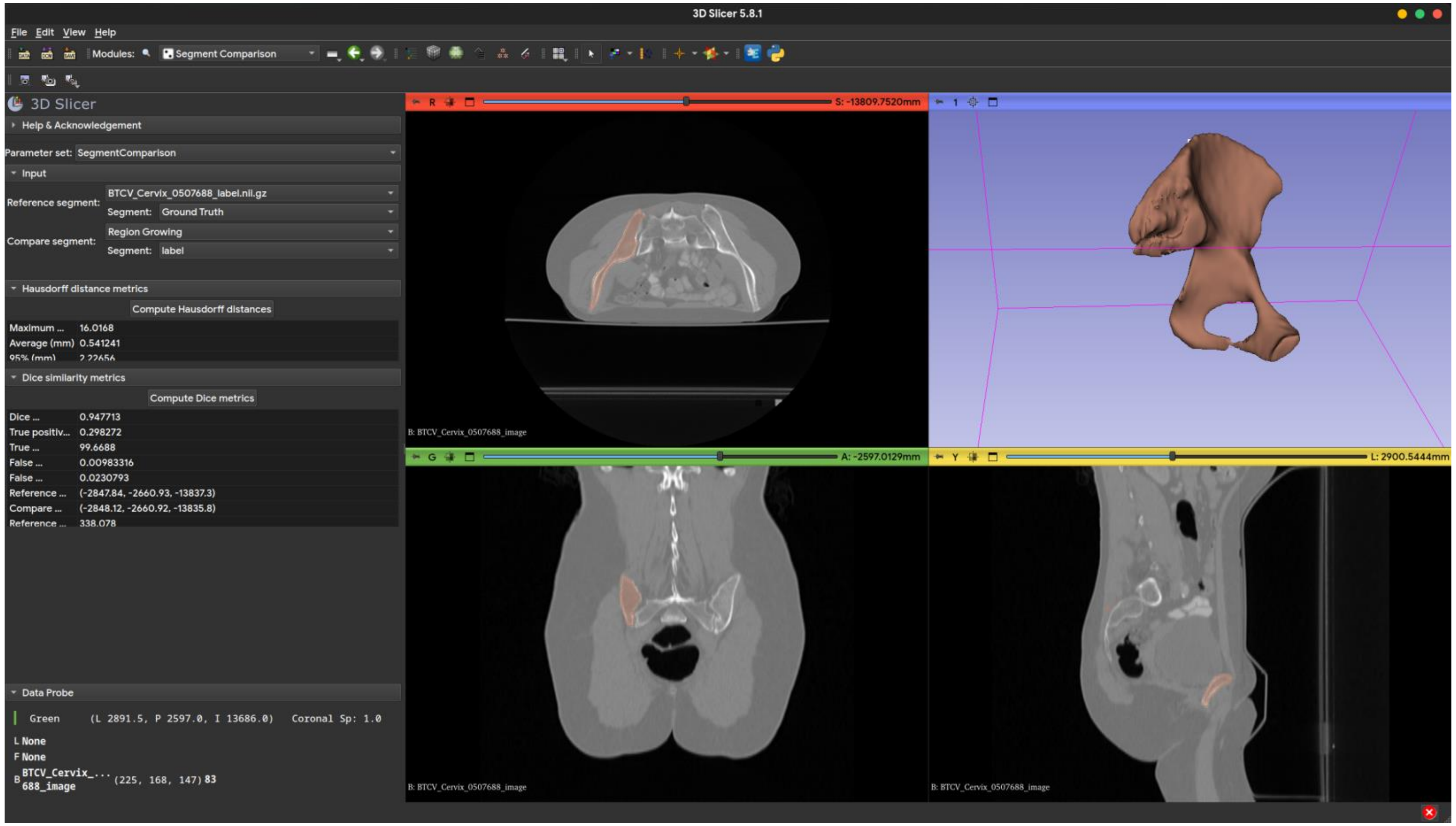
Classic Segmentation in 3D Slicer

- Region growing
  - You can see the preliminary 3D



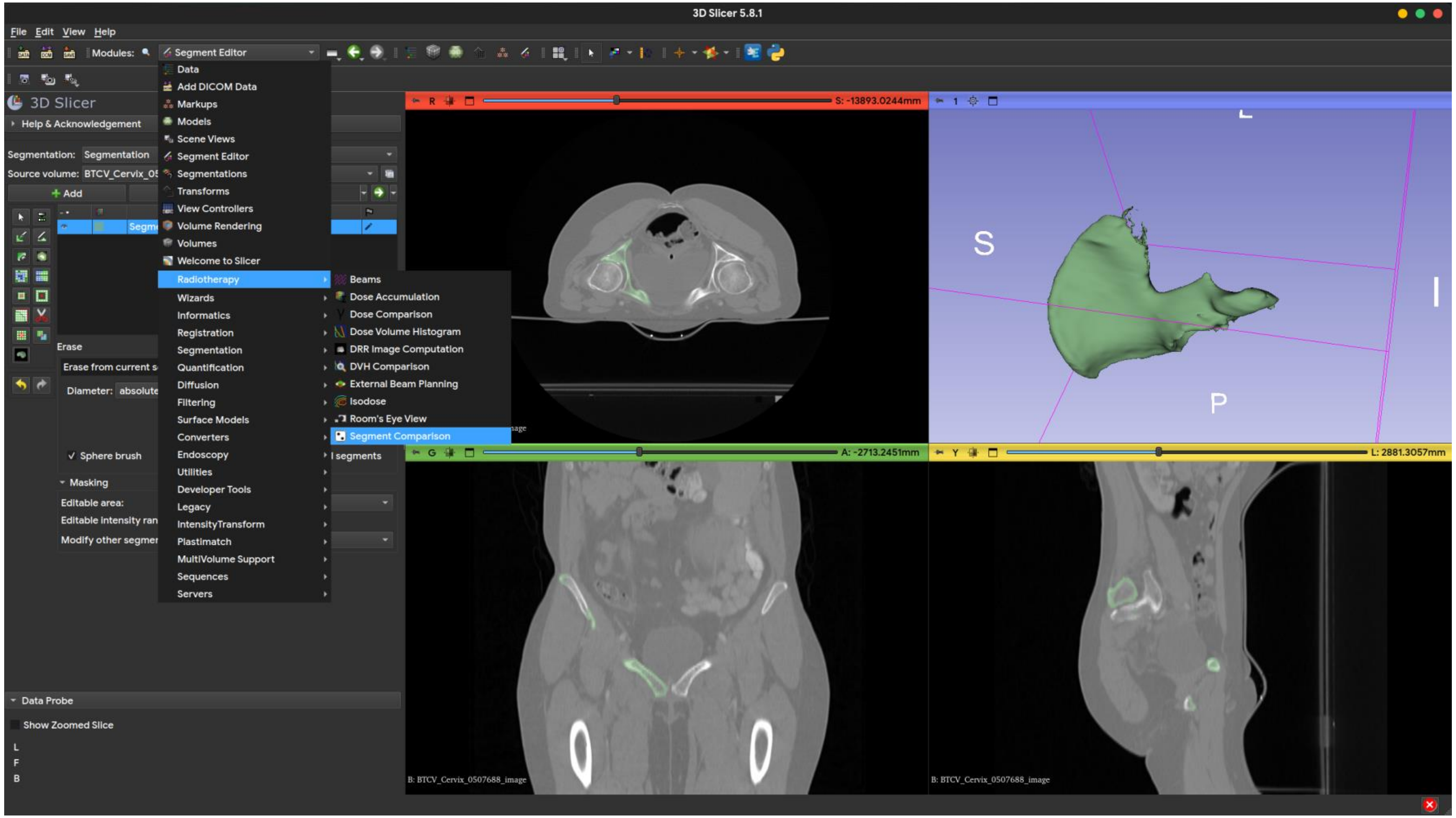
Classic Segmentation in 3D Slicer

- Region growing
  - Use the paintbrush to refine the segmentation



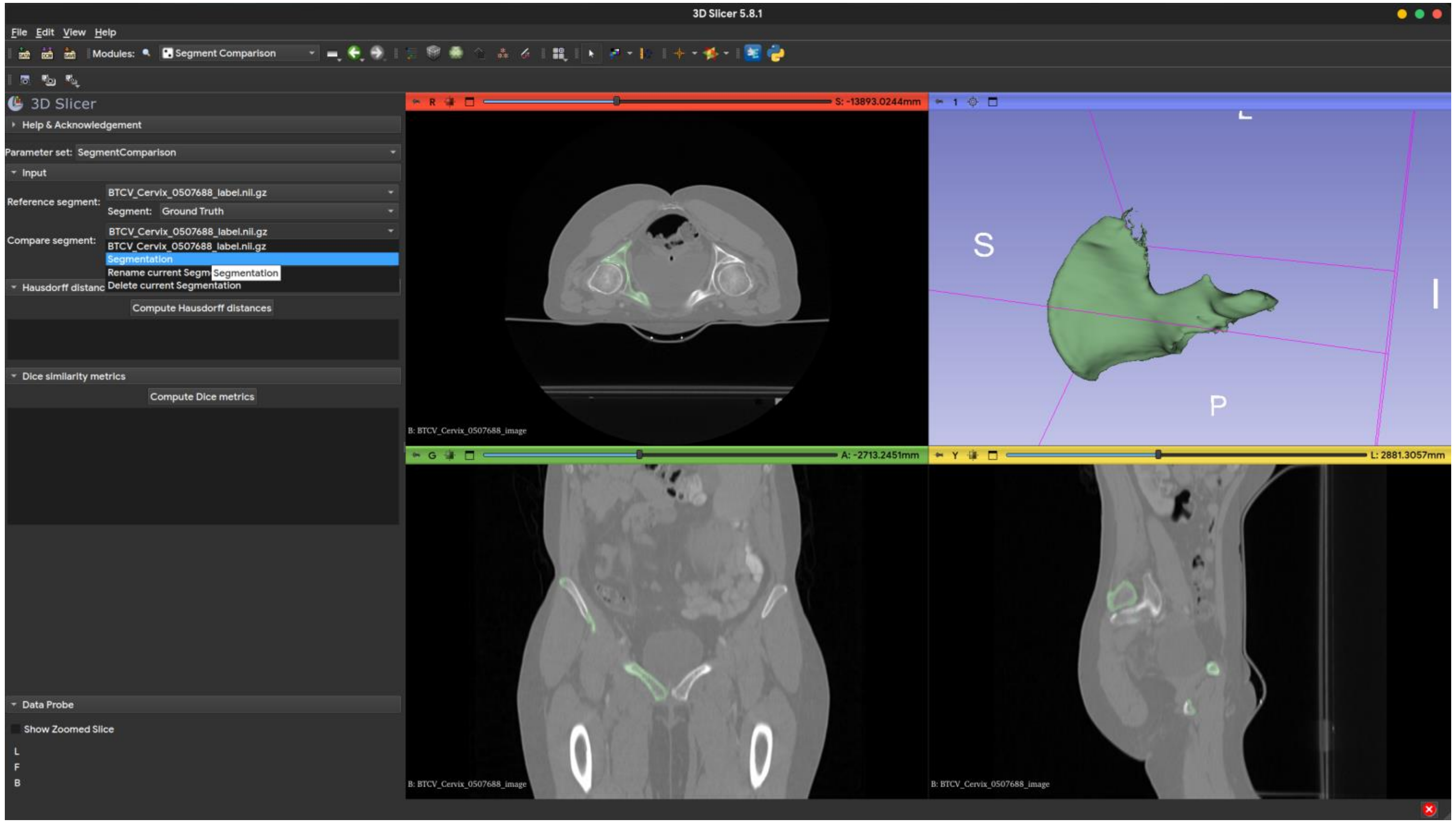
Classic Segmentation in 3D Slicer

- Evaluating the segmentation



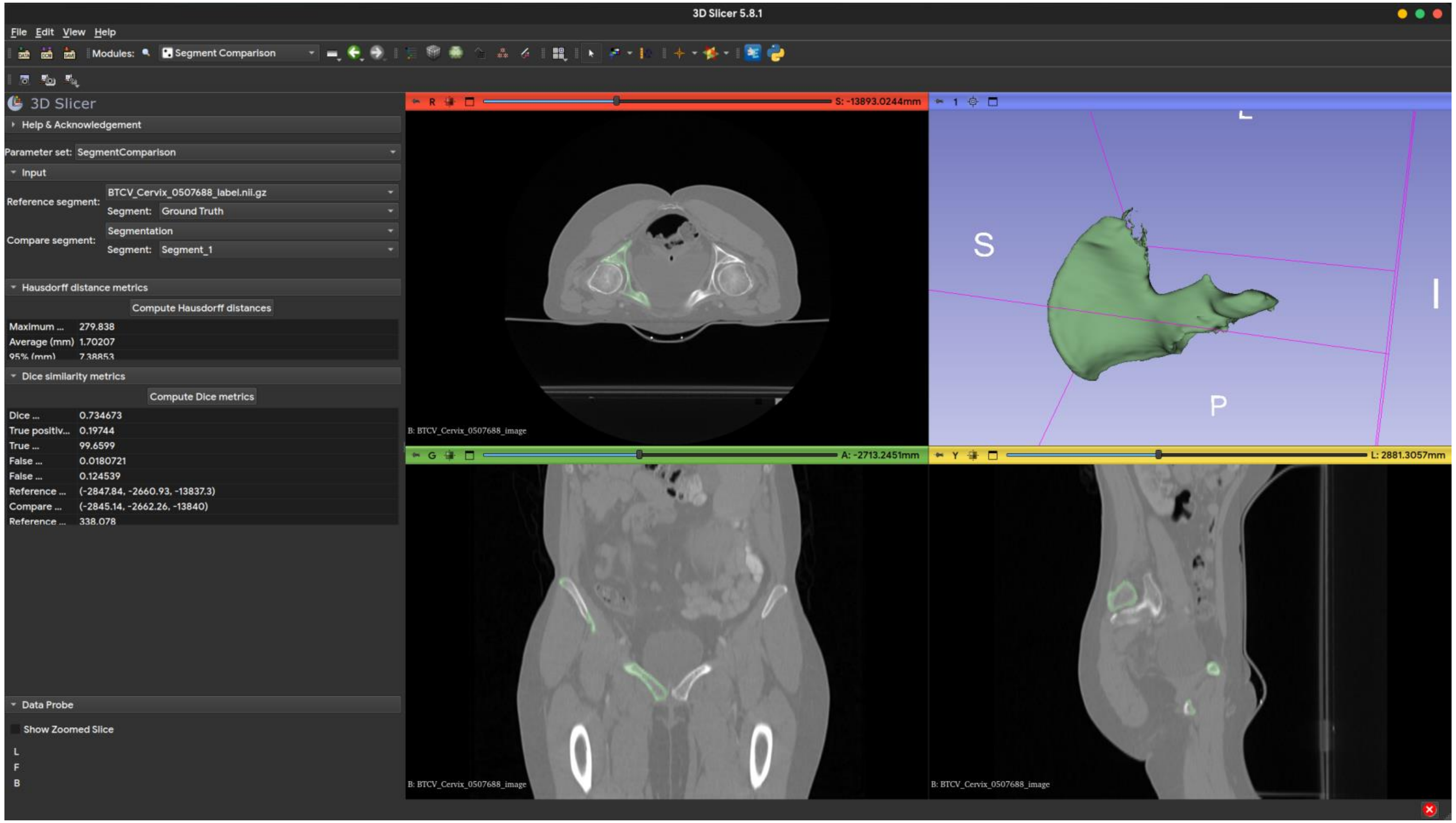
Classic Segmentation in 3D Slicer

- Evaluating the segmentation
  - Select the ground truth as the reference and the manual as compare



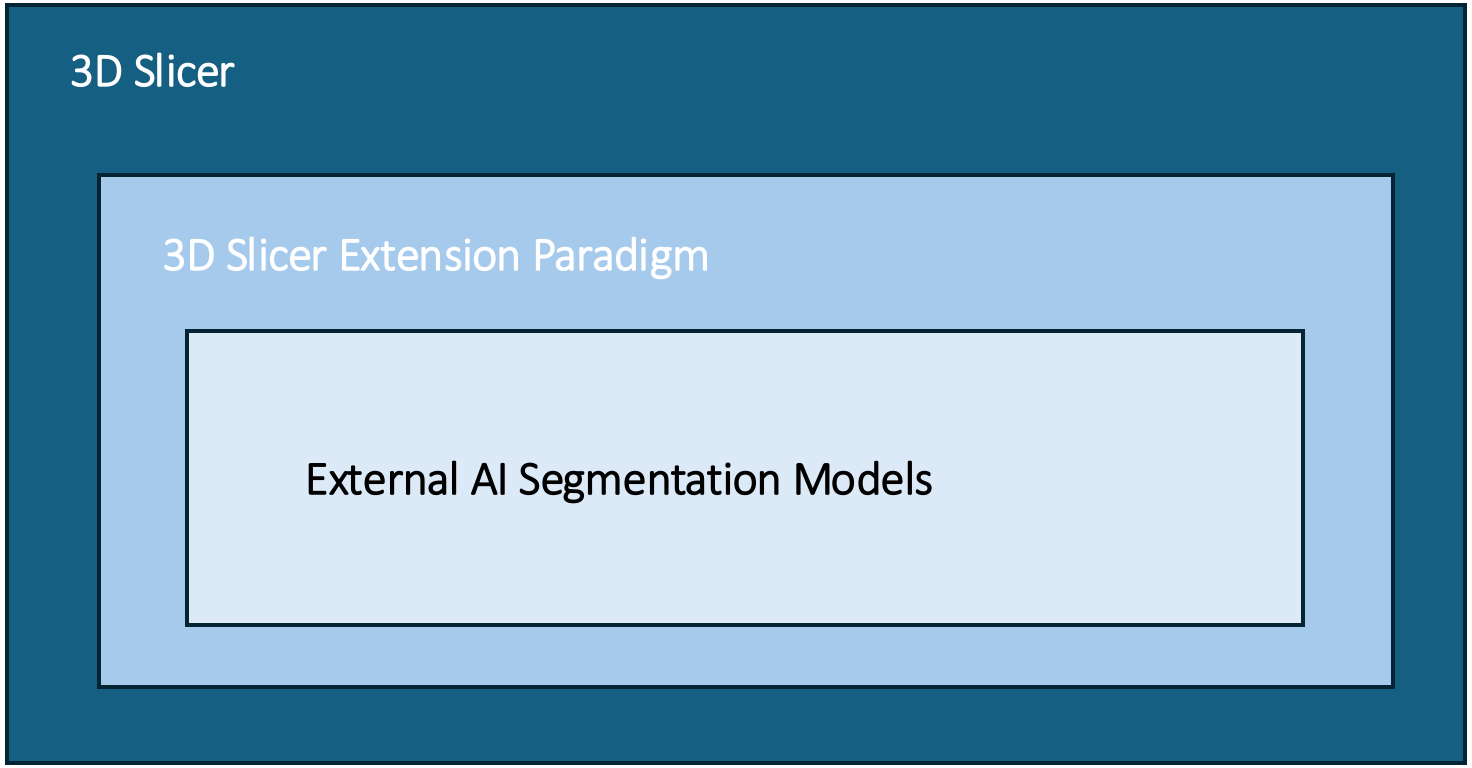
Classic Segmentation in 3D Slicer

- Evaluating the segmentation
  - Compute the DICE coefficient and the Hausdorff distance



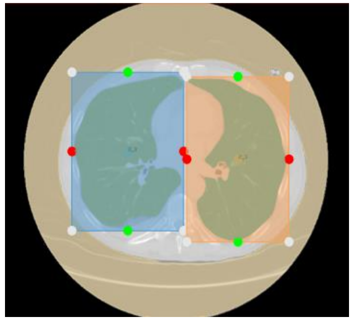
AI Segmentation in 3D Slicer

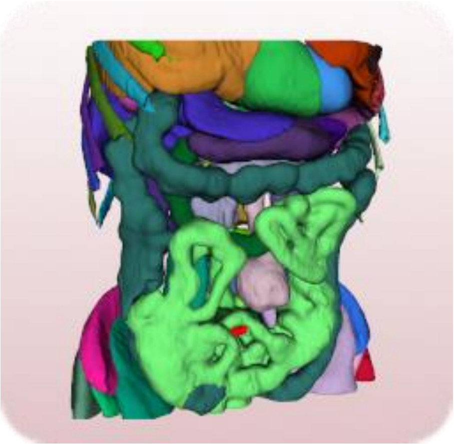
- 3D Slicer is a wrapper of externally-trained models

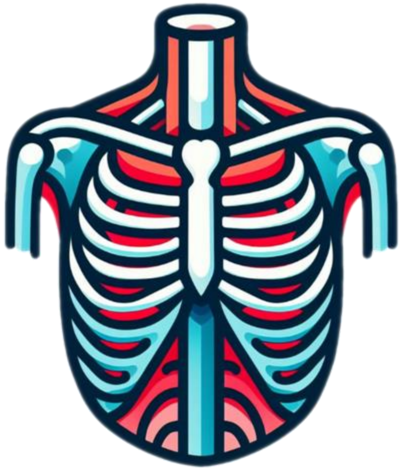


AI Segmentation in 3D Slicer

- Extensions







AI Segmentation in 3D Slicer

- Extensions

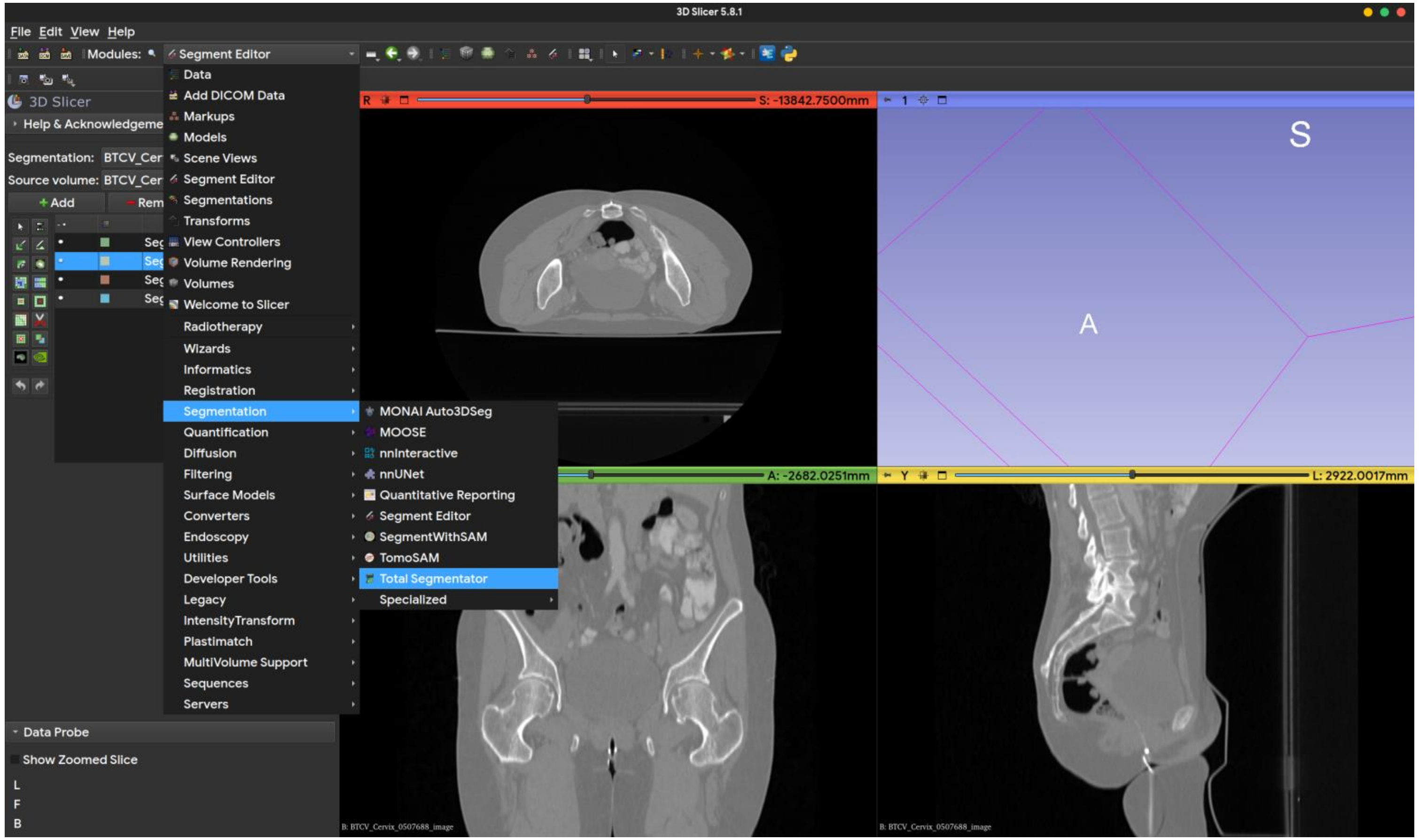
AI Segmentation in 3D Slicer

- Disclaimer:
  - When you install everything, click 'yes' to installing all libraries
  - Some extensions might break others because they use different library versions
  - They are unstable, so expect crashes (especially if you use CPU and no GPU)

AI Segmentation in 3D Slicer

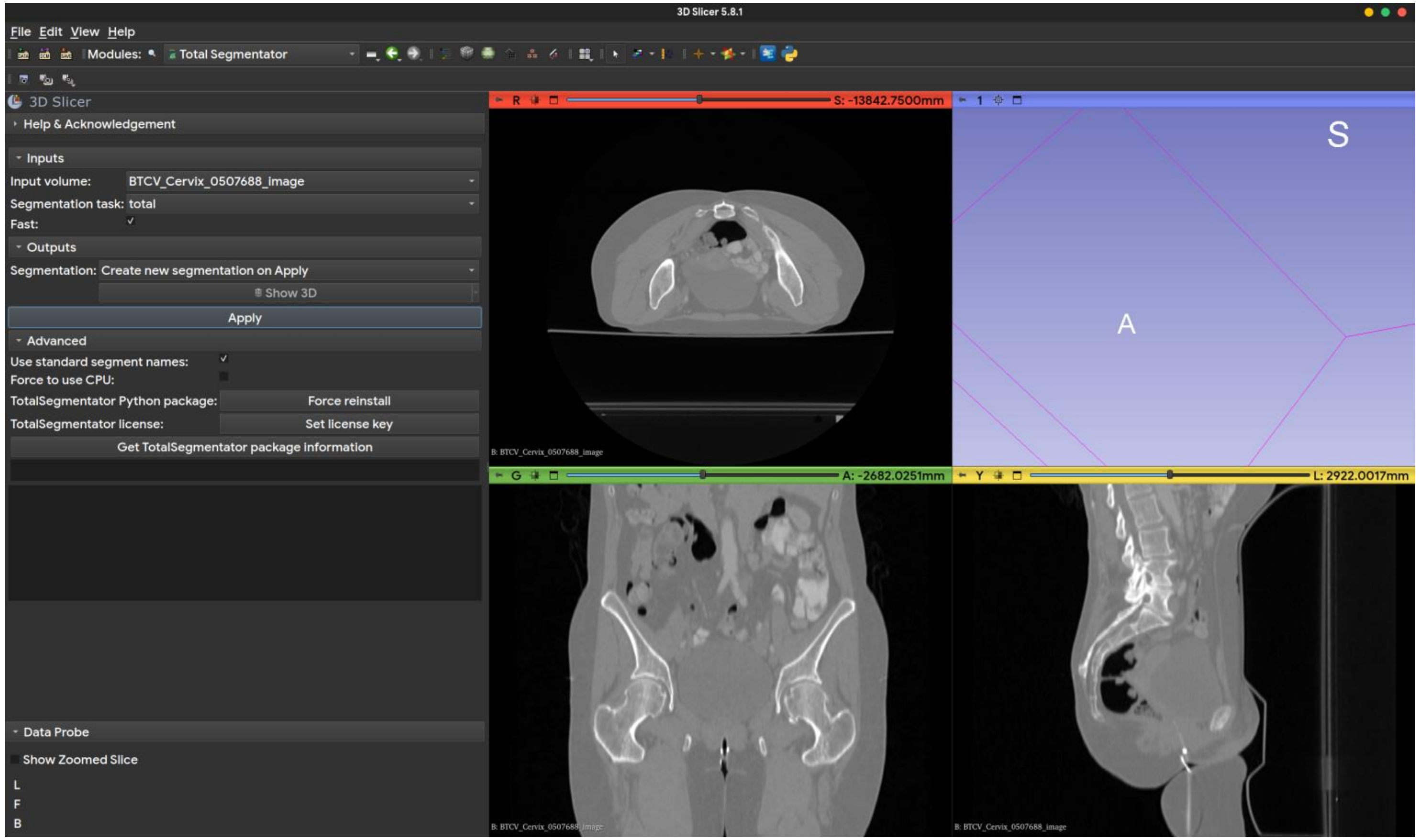
3D Slicer SOFA

- TotalSegmentator
  - By far the most popular and stable AI extension



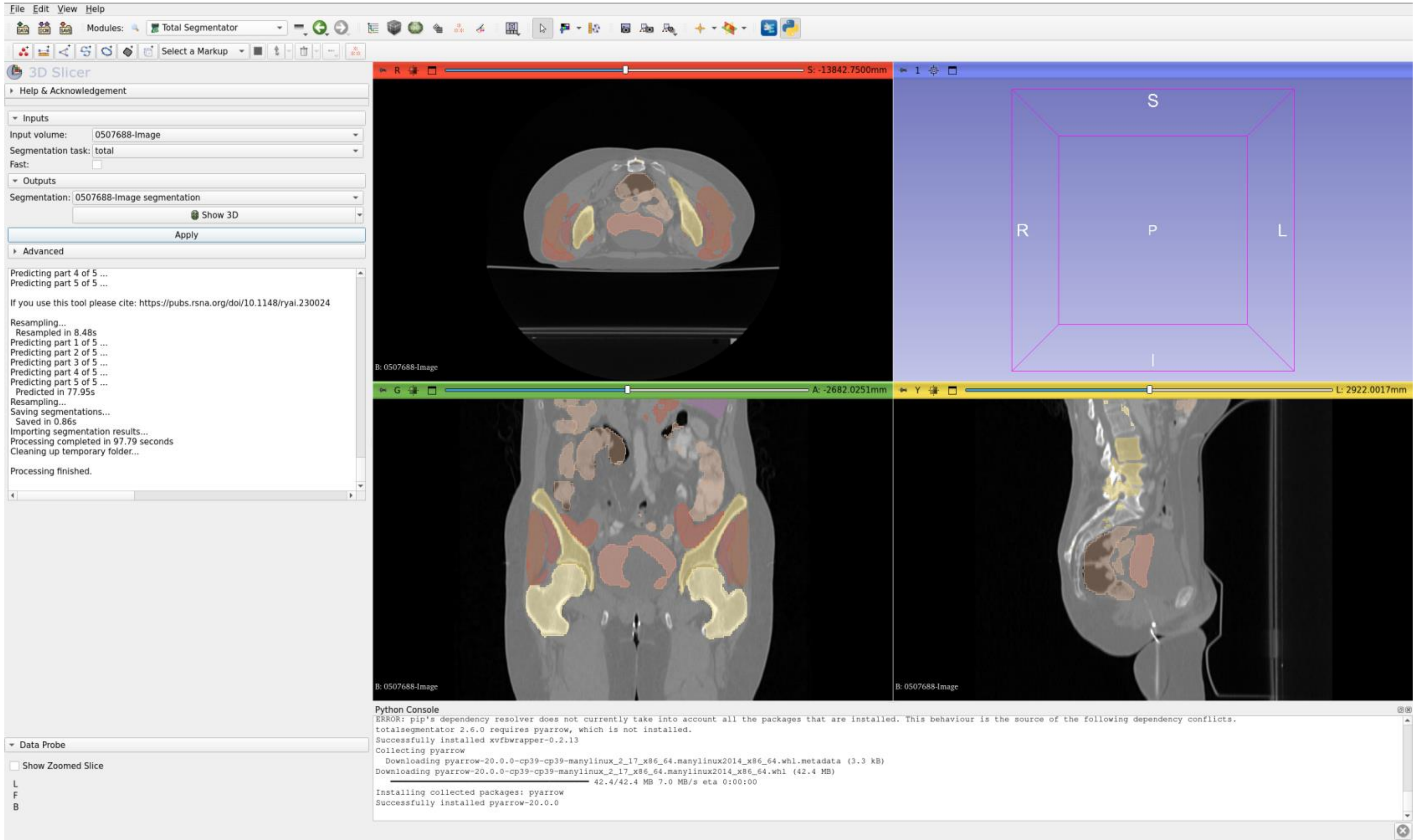
AI Segmentation in 3D Slicer

- TotalSegmentator
  - Choose 'fast' unless you have a desktop RTX-20XX GPU



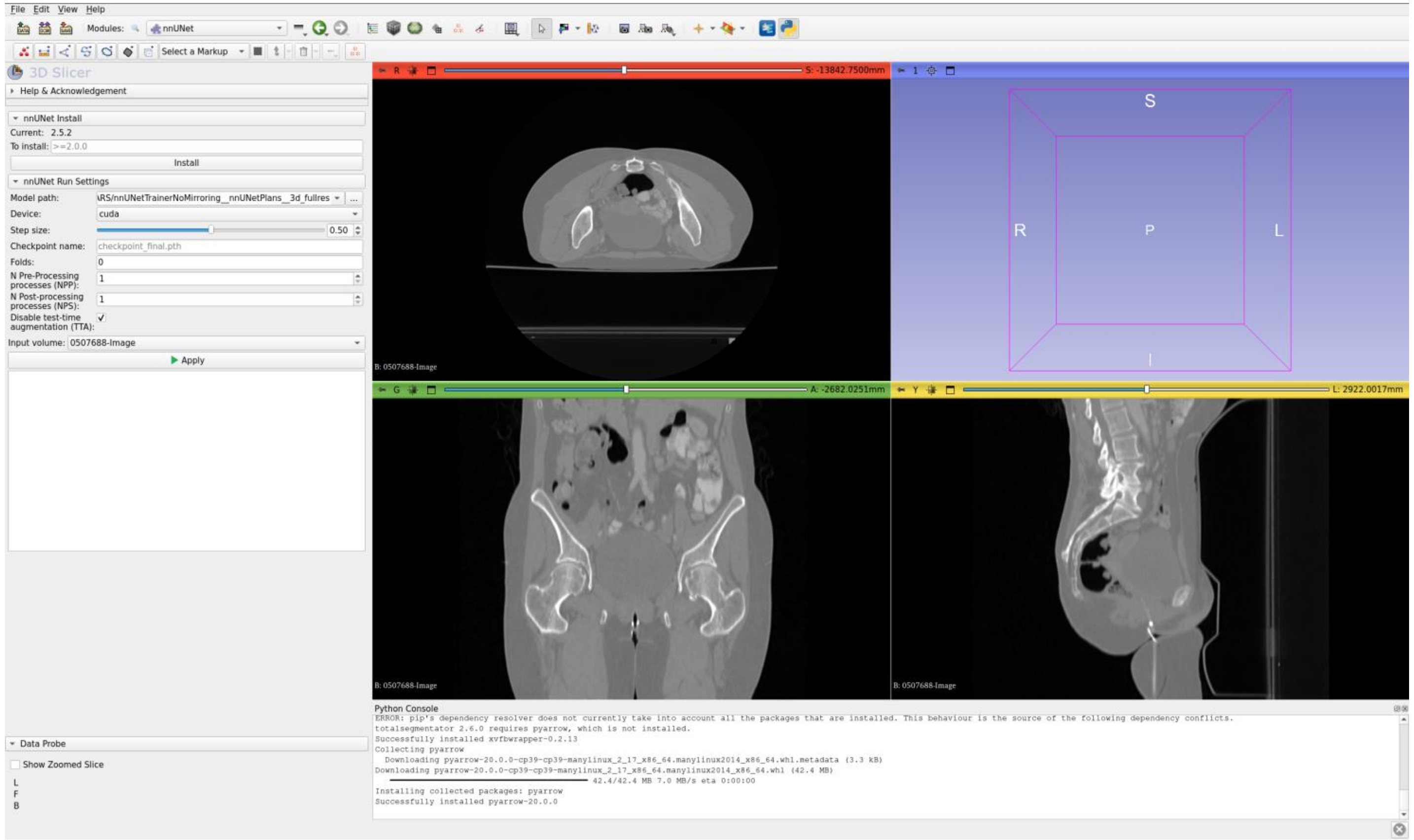
AI Segmentation in 3D Slicer

- TotalSegmentator
  - TotalSegmentator gives all the organs



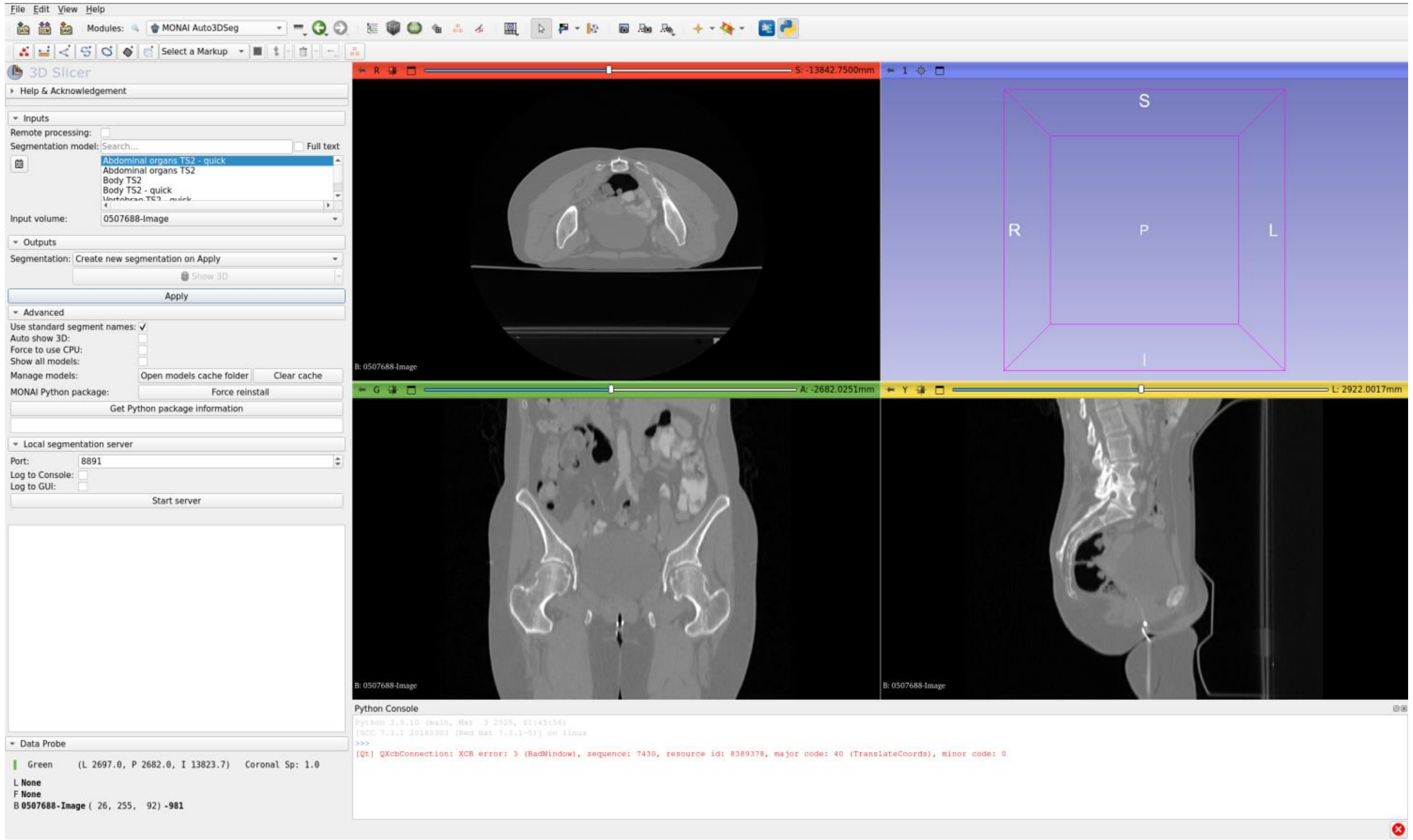
AI Segmentation in 3D Slicer

- SlicerNNUNet
  - Lets you natively run your nnUNet models in 3D Slicer
  - It expects nnUNet folder structure



AI Segmentation in 3D Slicer

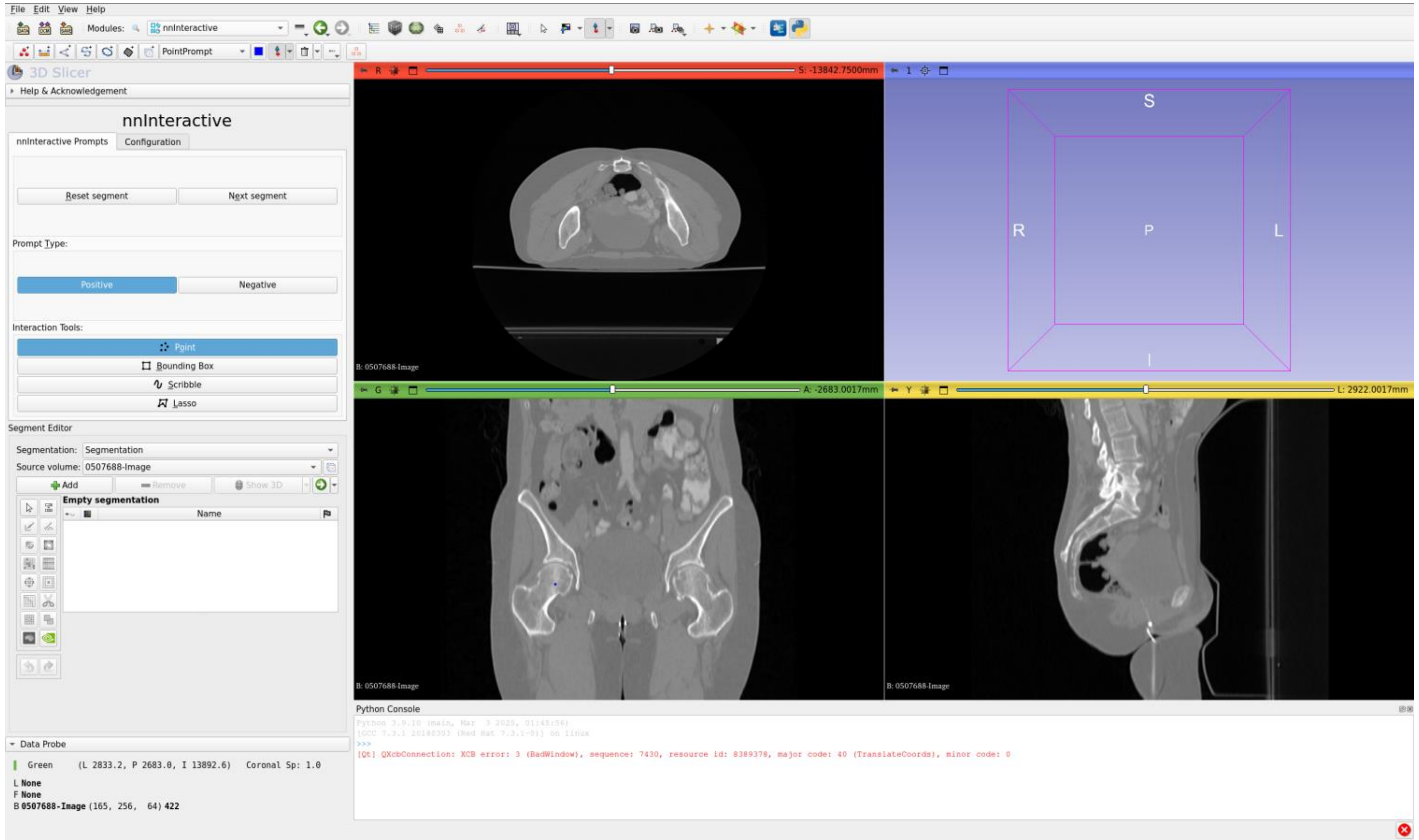
- MONAI Auto3DSeg
  - MONAI is the biggest medical DL library in Python
  - MONAI Auto3DSeg is the most popular alternative to TotalSegmentator



AI Segmentation in 3D Slicer

3D Slicer + SOFA

- nnInteractive
  - Prompt-based nnUNet module
  - Currently DKFZ (German Cancer Research Center) most potent model



AI Segmentation in 3D Slicer

- nnInteractive
  - Run the Docker (recommended) commands in the GitHub repo (https://github.com/coendevente/SlicerNNInteractive)

Option 1: Using Docker

```
docker pull coendevente/nninteractive-slicer-server:latest
docker run --gpus all --rm -it -p 1527:1527 coendevente/nninteractive-slicer-server:latest
```

This will make the server available under port `1527` on your machine. If you would like to use a different port, say `1627`, replace `-p 1527:1527` with `-p 1627:1527`.

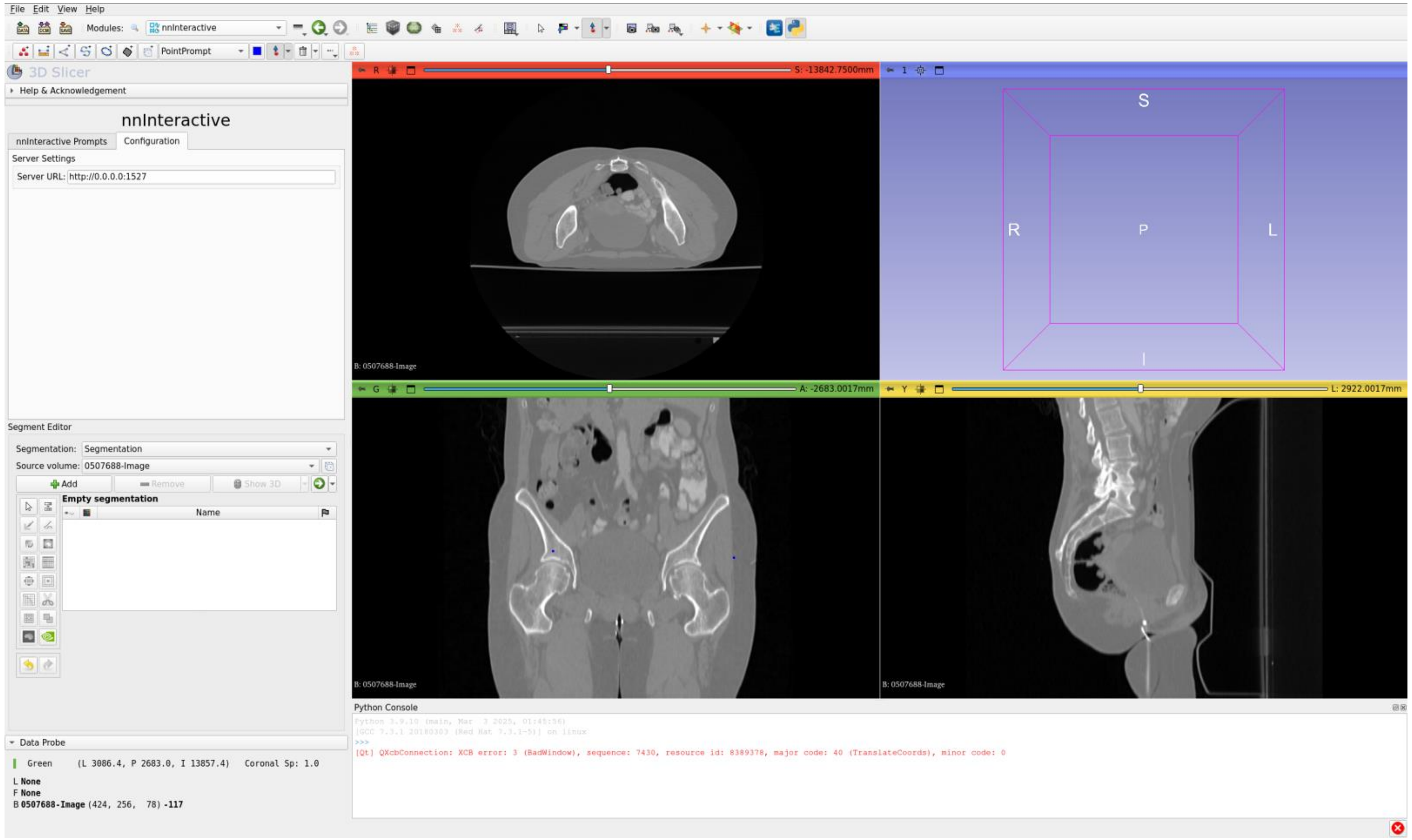
Option 2: Using `pip`

```
pip install nninteractive-slicer-server
nninteractive-slicer-server --host 0.0.0.0 --port 1527
```

```
nnUNet_raw is not defined and nnU-Net can only be used on data for which preprocessed files are already present on your system. nnU-Net cannot be used for experiment planning and preprocessing like this. If this is not intended,
please read documentation/setting_up_paths.md for information on how to set this up properly.
nnUNet_preprocessed is not defined and nnU-Net can not be used for preprocessing or training. If this is not intended, please read documentation/setting_up_paths.md for information on how to set this up.
nnUNet_results is not defined and nnU-Net cannot be used for training or inference. If this is not intended behavior, please read documentation/setting_up_paths.md for information on how to set this up.
plans.json: 100%|                    | 6.29k/6.29k [00:00<00:00, 31.5MB/s]
dataset.json: 100%|                  | 245/245 [00:00<00:00, 2.68MB/s]
inference_session_class.json: 100%|  | 121/121 [00:00<00:00, 997kB/s]
checkpoint_final.pth: 100%|          | 411M/411M [00:35<00:00, 11.5MB/s]
Fetching 4 files: 100%|              | 4/4 [00:36<00:00,  9.05s/it]
INFO:     Started server process [1]
INFO:     Waiting for application startup.
INFO:     Application startup complete.
INFO:     Uvicorn running on http://0.0.0.0:1527 (Press CTRL+C to quit)
```

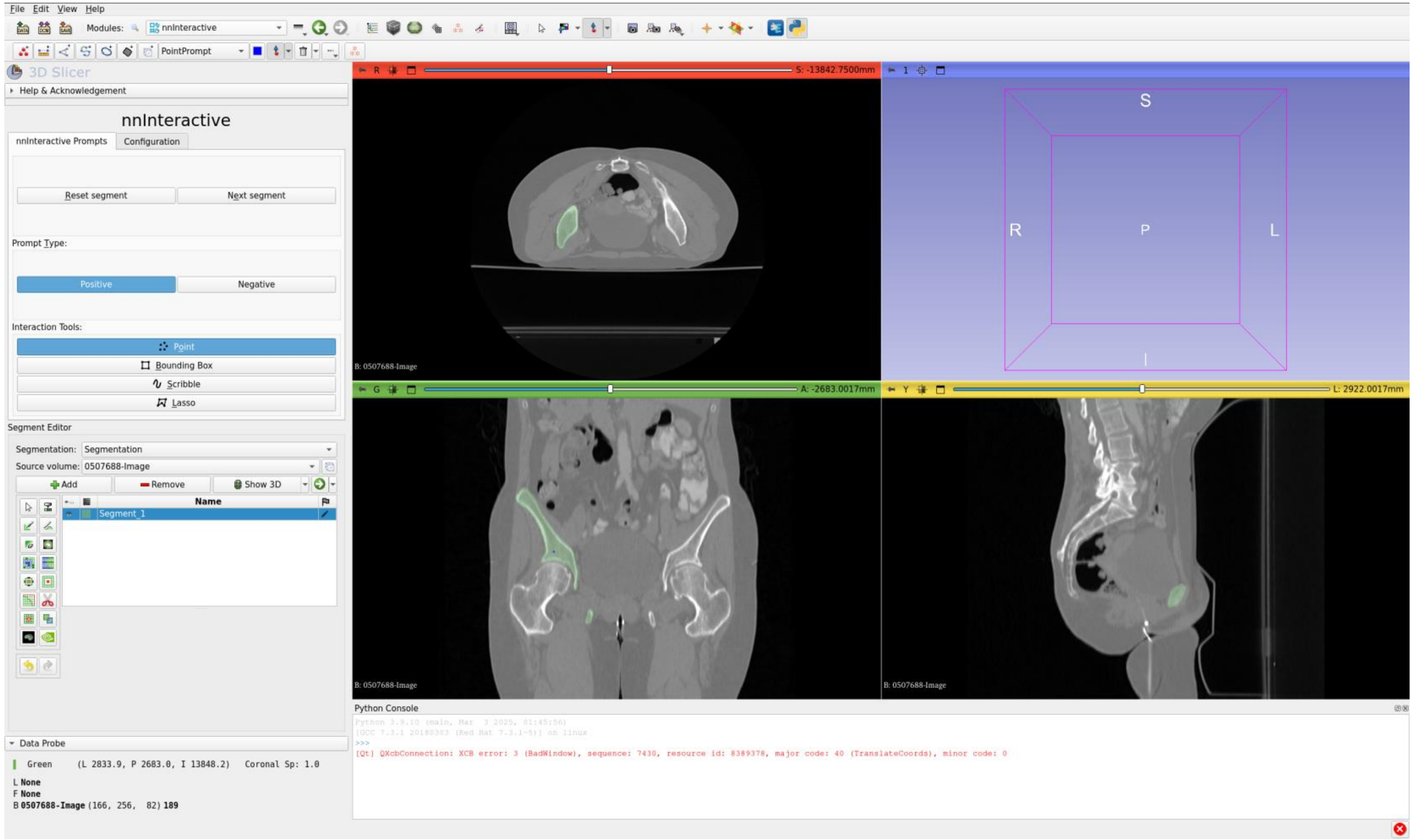
AI Segmentation in 3D Slicer

- nnInteractive
  - Choose the port on the extension



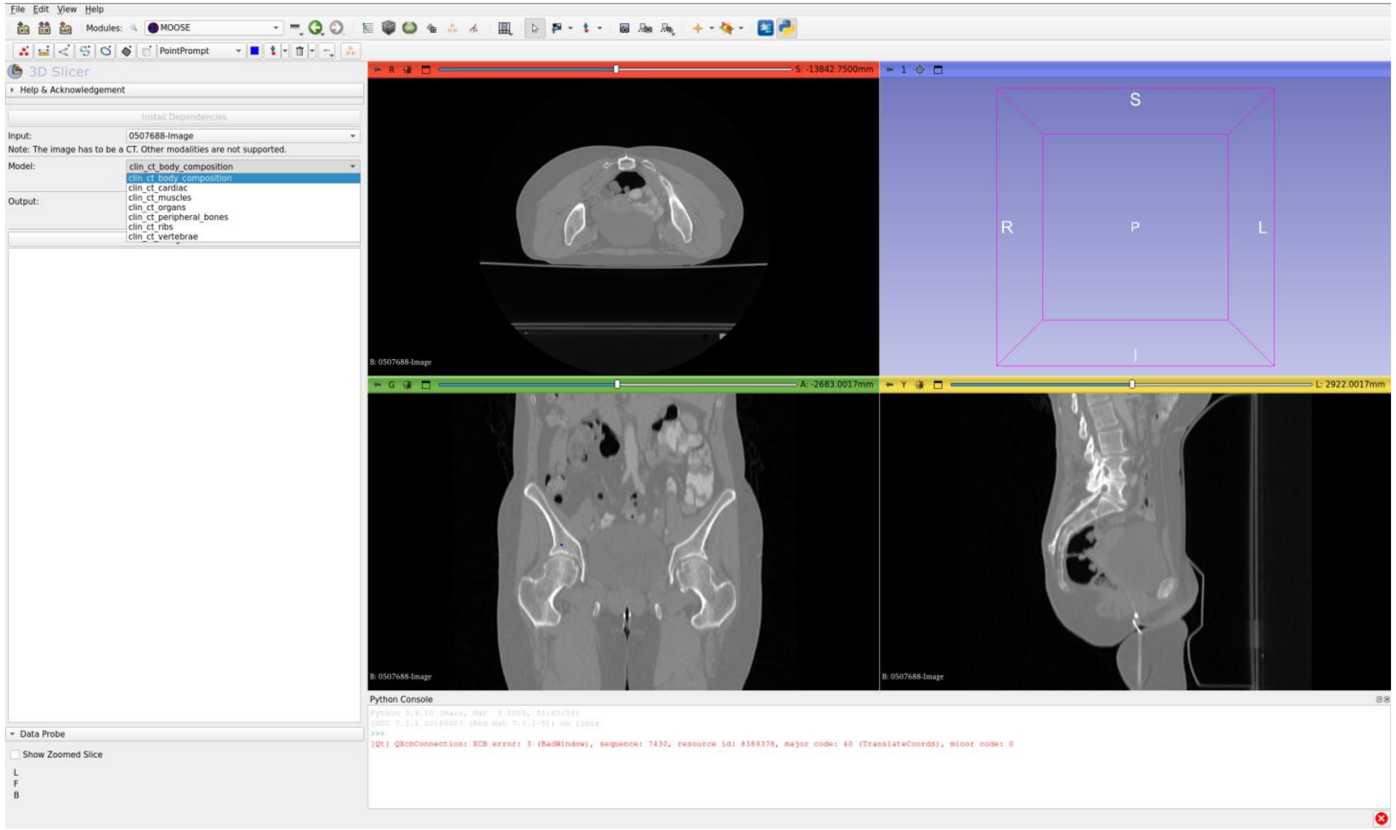
AI Segmentation in 3D Slicer

- nnInteractive
  - Similar to region growing: select area of interest



AI Segmentation in 3D Slicer

- MOOSE
  - Multilabel models similar to TotalSegmentator and Auto3DSeg
  - Extremely fast, based on nnUNet, work only on CT



Segmentation with AI

Lucía Cubero – lcubero@ing.uc3m.es

Sergio Carreras – scarrera@ing.uc3m.es

MINISTERIO DE CIENCIA, INNOVACIÓN Y UNIVERSIDADES

Cofinanciado por la Unión Europea

AGENCIA ESTATAL DE INVESTIGACIÓN

Comunidad de Madrid