http://braintumorsegmentation.org

The **Brain Tumor Segmentation (BraTS) challenge** celebrated its **10th anniversary** in 2021.

The **RSNA-ASNR-MICCAI BraTS 2021** challenge utilizes multi-institutional pre-operative baseline multi-parametric magnetic resonance imaging (mpMRI) scans, and focuses on the evaluation of state-of-the-art methods for (**Task 1**) the **segmentation** of intrinsically heterogeneous brain glioblastoma sub-regions in mpMRI scans.

Furthemore, this BraTS 2021 challenge also focuses on the evaluation of (**Task 2**) **classification** methods to predict the MGMT promoter methylation status.

Previous BraTS Challenges

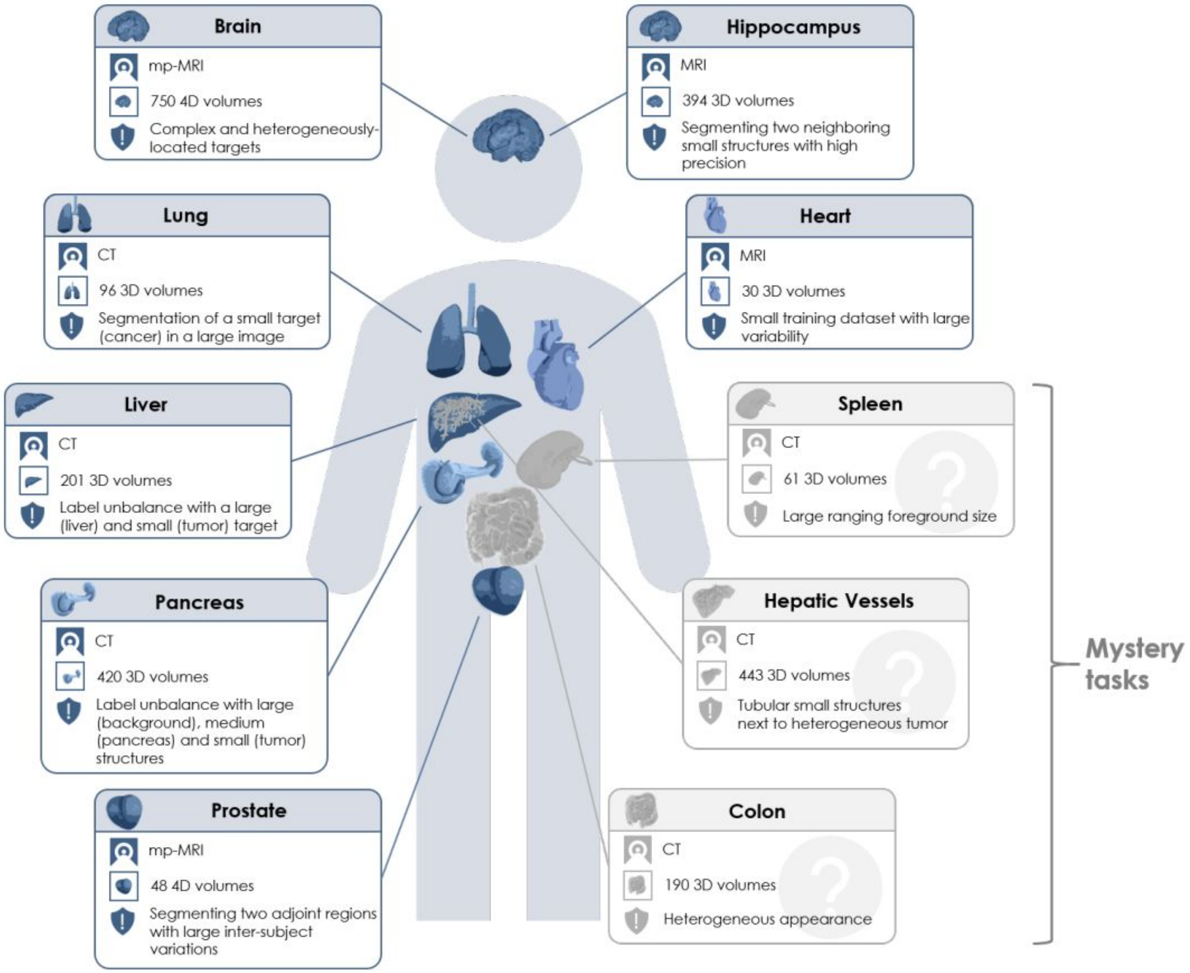
- BraTS 2012 (Nice, France) - [proceedings]
- BraTS 2013 (Nagoya, Japan) - [proceedings]
- BraTS 2014 (Boston, USA) - [proceedings]
- BraTS 2015 (Munich, Germany) - [proceedings]
- BraTS 2016 (Athens, Greece) - [proceedings]
- BraTS 2017 (Quebec City, Canada) - [proceedings]
- BraTS 2018 (Granada, Spain) - [proceedings]
- BraTS 2019 (Shenzhen, China) - [proceedings: vol.1, vol.2]
- BraTS 2020 (virtual) - [proceedings: vol.1, vol.2]

Data from BraTS challenges (used in this Lab) have been incorporated in the:

International challenges have become the de facto standard for comparative assessment of image analysis algorithms given a specific task. Segmentation is so far the most widely investigated medical image processing task, but the various segmentation challenges have typically been organized in isolation, such that algorithm development was driven by the need to tackle a single specific clinical problem. **We hypothesized that a method capable of performing well on multiple tasks will generalize well to a previously unseen task and potentially outperform a custom-designed solution.**

http://medicaldecathlon.com

MSD



Overview of the ten different tasks of the Medical Segmentation Decathlon (MSD) http://medicaldecathlon.com

The challenge comprised different target regions, modalities and challenging characteristics and was separated into seven known tasks (blue; the **development phase**)

and three mystery tasks probing **generalization** to unseen segmentation tasks.

mp-MRI= multiparametric-magnetic resonance imaging

Development Phase - **Brain** (Brain Tumor Segmentation (**BraTS**) challenges)

- The data set consists of 750 multiparametric magnetic resonance images (mp-MRI) from patients diagnosed with either glioblastoma or lower-grade glioma.

- The sequences used were native T1-weighted (**T1**), post-Gadolinium (Gd) contrast T1-weighted (**T1-Gd**), native T2-weighted (**T2**), and T2 Fluid-Attenuated Inversion Recovery (**FLAIR**).

- The corresponding target ROIs were the three tumor sub-regions, namely **edema**, **enhancing tumor**, and **non-enhancing tumor**.

- This data set was selected due to the challenge of locating these complex and heterogeneously-located targets.

- The data was acquired from 19 different institutions and contained a subset of the data used in the 2016 and 2017 Brain Tumor Segmentation (BraTS) challenges [12, 13, 14].

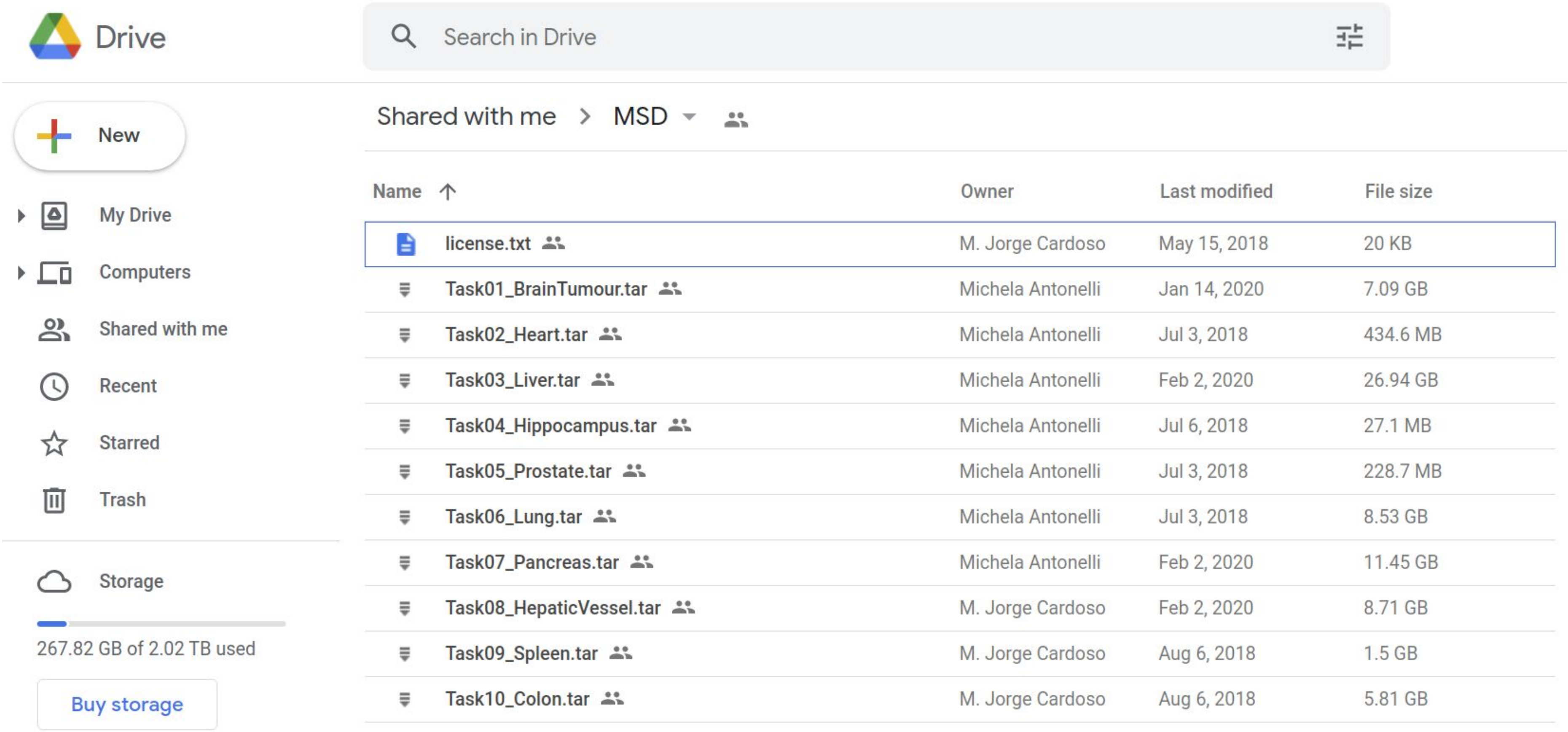
Summary of the ten data sets of the Medical Segmentation Decathlon

Phase	Task	Modality	Protocol	Target	# Cases (Train/Test)
Development phase	Brain	mp-MRI	FLAIR, T1w, T1 \w Gd, T2w	Edema, enhancing and non-enhancing tumor	750 4D volumes (484/266)
	Heart	MRI	—	Left atrium	30 3D volumes (20/10)
	Hippocampus	MRI	T1w	Anterior and posterior of hippocampus	394 3D volumes (263/131)
	Liver	CT	Portal venous phase	Liver and liver tumor	210 3D volumes (131/70)
	Lung	CT	—	Lung and lung cancer	96 3D volumes (64/32)
	Pancreas	CT	Portal venous phase	Pancreas and pancreatic tumor mass	420 3D volumes (282/139)
	Prostate	mp-MRI	T2, ADC	Prostate PZ and TZ	48 4D volumes (32/16)
Mystery phase	Colon	CT	Portal venous phase	Colon cancer primaries	190 3D volumes (126/64)
	Hepatic Vessels	CT	Portal venous phase	Hepatic vessels and hepatic tumor	443 3D volumes (303/140)
	Spleen	CT	Portal venous phase	spleen	61 3D volumes (41/20)

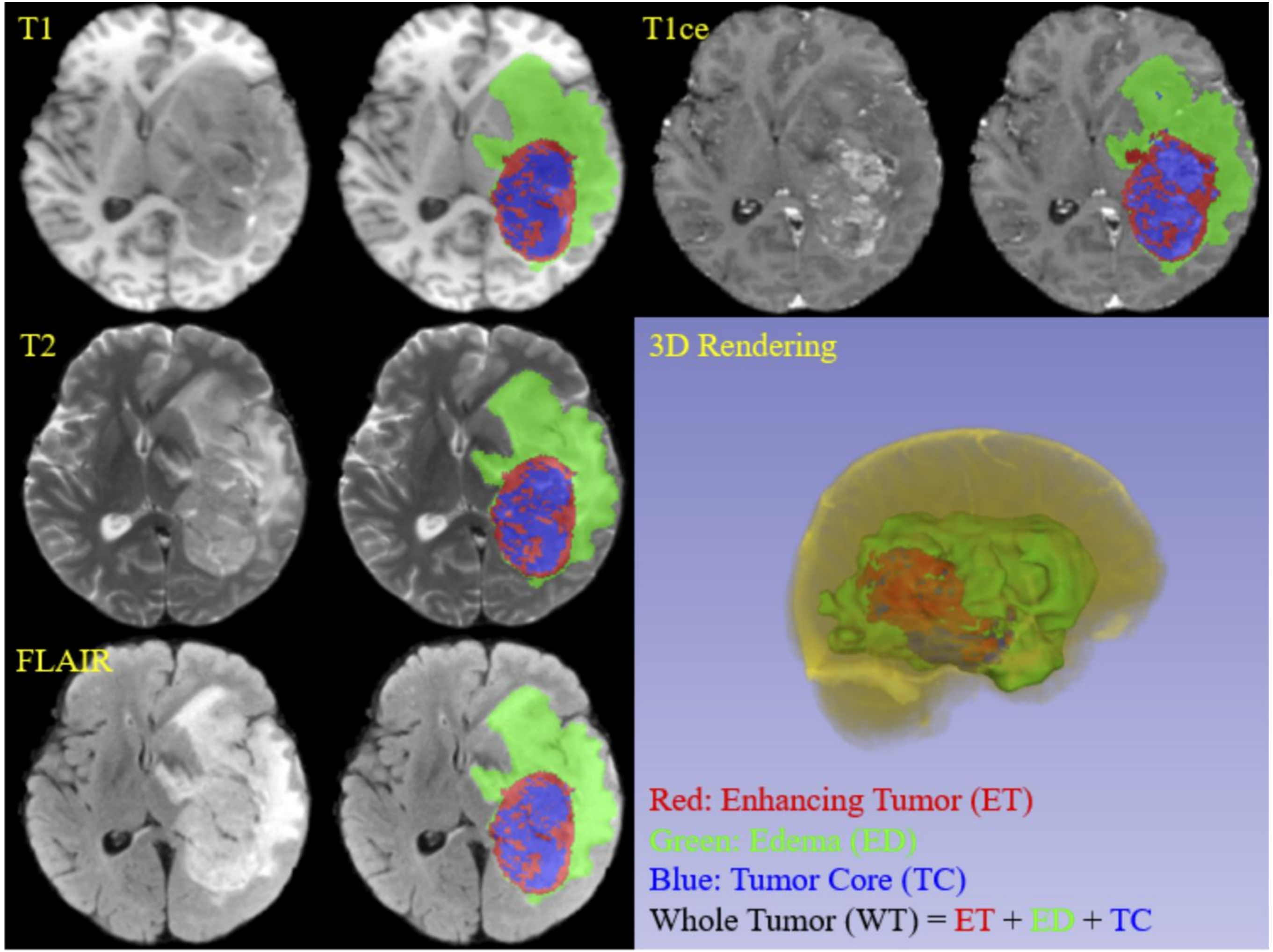
(Prostate: PZ - peripheral zone, TZ - transition zone)

http://medicaldecathlon.com

Get the data from: https://drive.google.com/drive/folders/1HqEgzS8BV2c7xYNrZdEAnrHk7osJJ--2



Or, download Decathlon Dataset with Monai APIs: https://github.com/NVIDIA/clara-train-examples/blob/master/PyTorch/NoteBooks/Data/DownloadDecathlonDataSet.ipynb



Visualized examples in BraTS Challenge

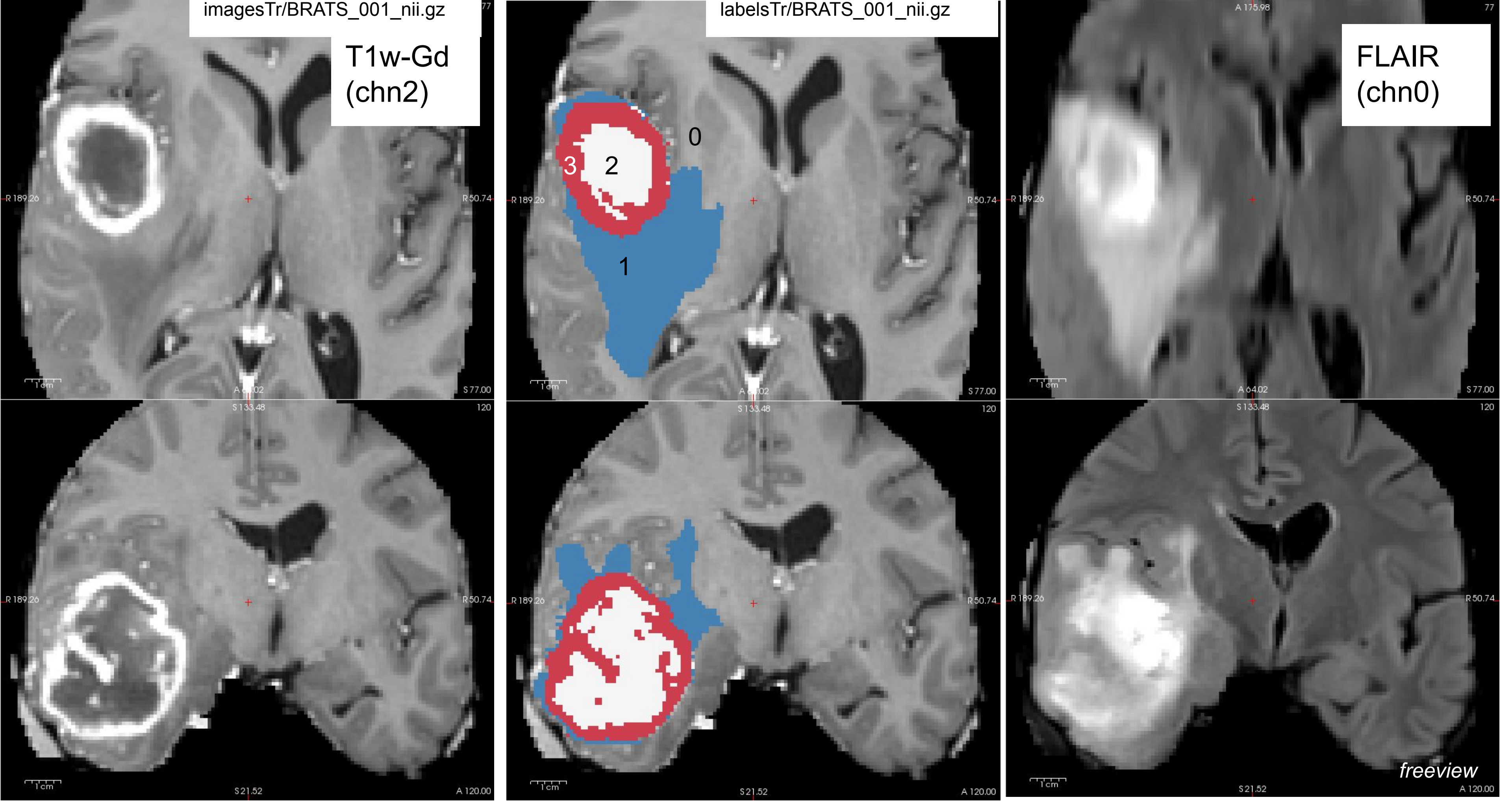
Ground truth is shown in 2D projected onto the multi-sequence MR images and in 3D together with a volume rendering of the raw data

nnU-Net backbone All the top methods used UNet like architectures in the ten 3D segmentation challenges.

Remarkably, **nnU-Net** was used by the top teams in nine out of ten challenges, because it is open-sourced, powerful, flexible, and out-of-the-box. Participants can easily integrate their new methods into nnU-Net.

Check: https://youtu.be/C6tpnJRpt90

https://arxiv.org/pdf/2101.00232.pdf



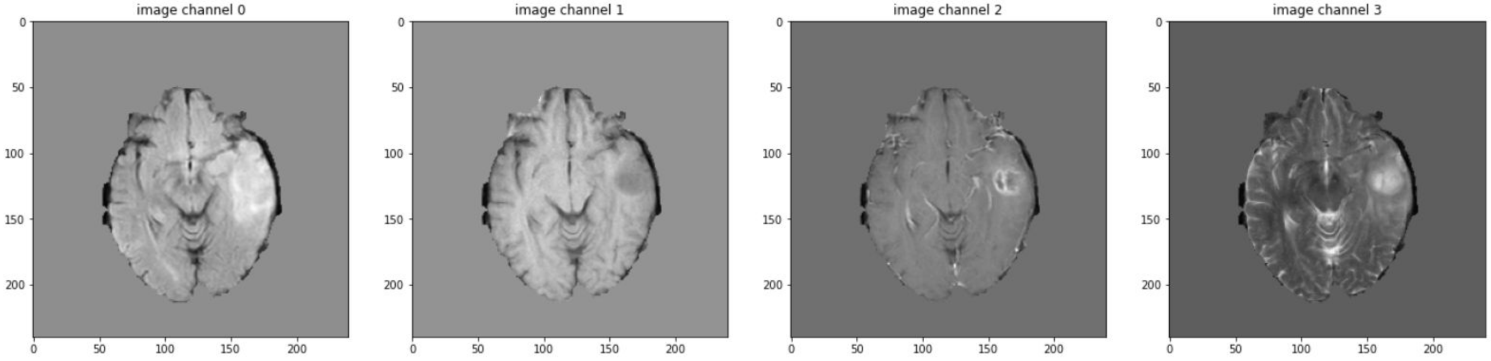
Tumor core (**TC**): label 2 + label 3 ¦ Whole tumor (**WT**): label 1 + label 2 + label 3 ¦ Enhancing tumor (**ET**): label 3 ¦ chn1: T1w, chn3: T2w

Check data shape and visualize

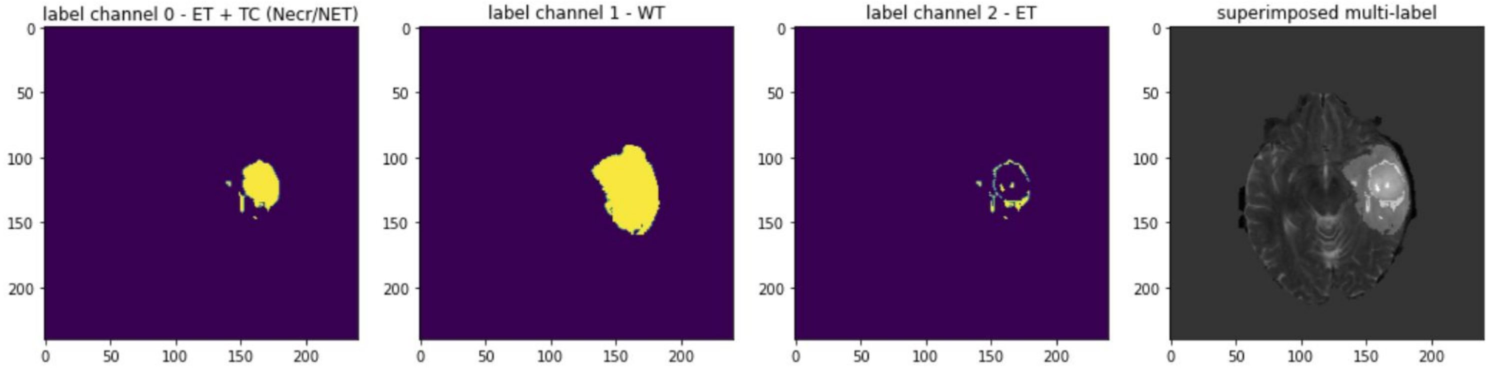
```
# pick one image from DecathlonDataset to visualize and check the 4 channels

nshape = val_ds[2]["image"][:, :, :, 60].detach().cpu().numpy().shape
```

```
image shape: torch.Size([4, 240, 240, 155])
```




```
label shape: torch.Size([3, 240, 240, 155])
```



https://nbviewer.org/github/MMIV-ML/ELMED219-2022/blob/main/Lab4-BRATS/ELMED219-2022-Lab4-BRATS-segmentation-3D-MONAI.ipynb

```python
# standard PyTorch program style: create SegResNet, DiceLoss and Adam optimizer
device = torch.device("cuda:0")
model = SegResNet(
    blocks_down=[1, 2, 2, 4],
    blocks_up=[1, 1, 1],
    init_filters=16,
    in_channels=4,
    out_channels=3,
    dropout_prob=0.2,
).to(device)

loss_function = DiceLoss(smooth_nr=0, smooth_dr=1e-5, squared_pred=True, to_onehot_y=False, sigmoid=True)

optimizer = torch.optim.Adam(model.parameters(), 1e-4, weight_decay=1e-5)

lr_scheduler = torch.optim.lr_scheduler.CosineAnnealingLR(optimizer, T_max=max_epochs)

dice_metric = DiceMetric(include_background=True, reduction="mean")
dice_metric_batch = DiceMetric(include_background=True, reduction="mean_batch")

post_trans = Compose(
    [EnsureType(), Activations(sigmoid=True), AsDiscrete(threshold=0.5)]
)
```

```python
VAL_AMP = True # Automatic Mixed Precision
```

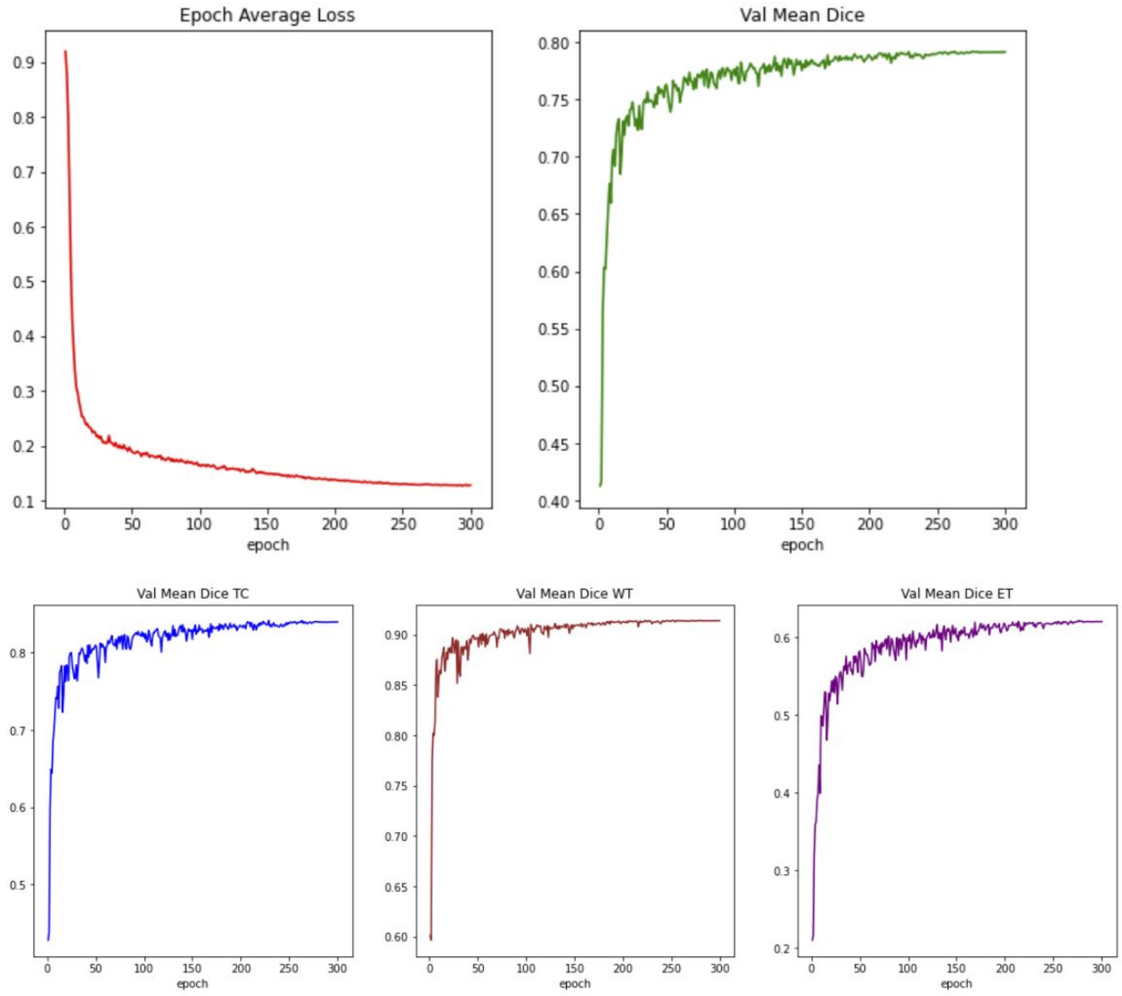
https://pytorch.org/docs/stable/notes/amp_examples.html

The NVIDIA RTX 6000 GPU workstation where the `best_metric_model_5820_rtx6000_300epochs_20220115_1048.pth` model was trained:

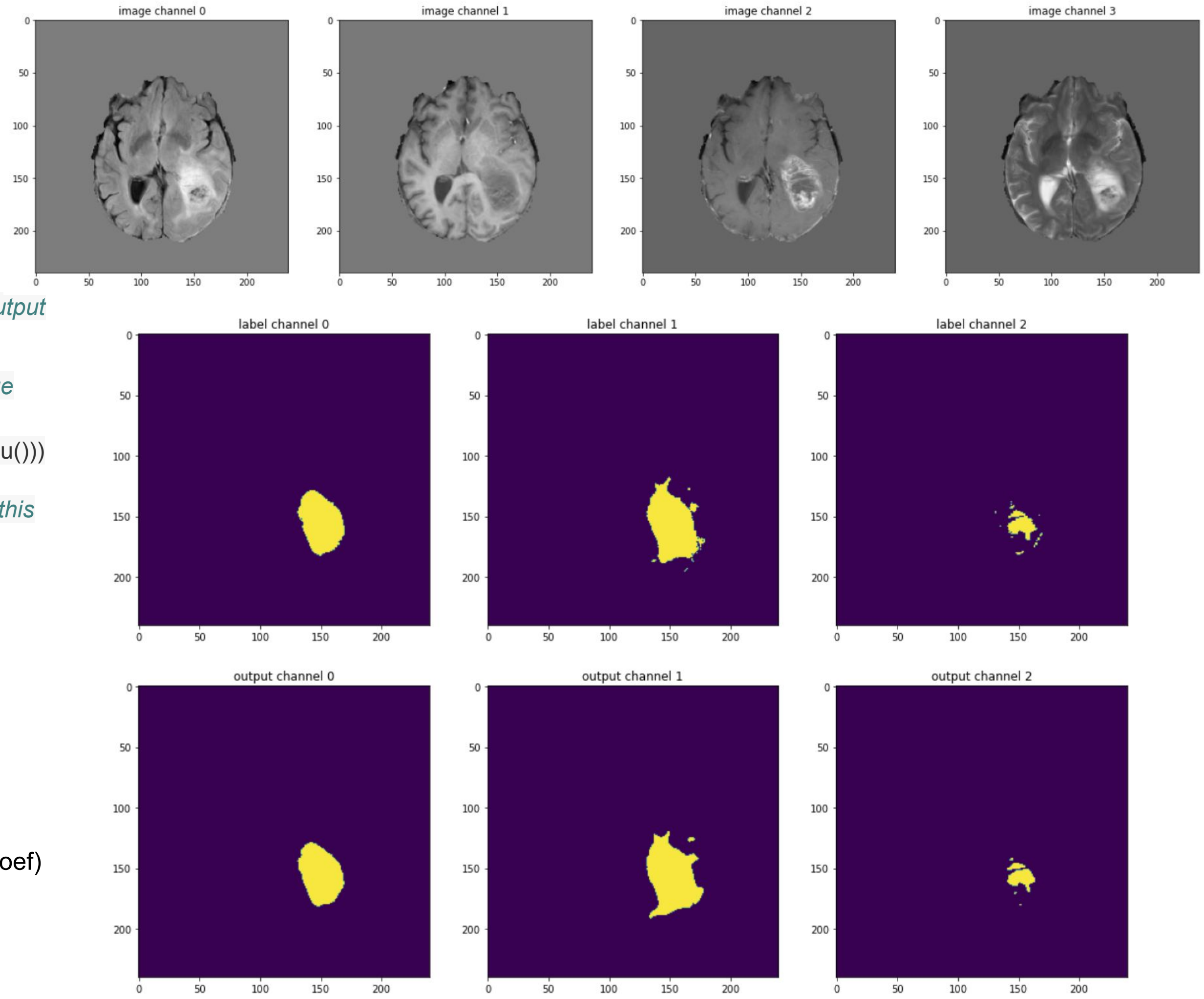
```
Fri Jan 14 13:40:10 2022
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 465.19.01    Driver Version: 465.19.01    CUDA Version: 11.3     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA Quadro R...  On   | 00000000:65:00.0  On |                  Off |
| 33%   29C    P8    19W / 260W |    524MiB / 24197MiB |      9%      Default |
|                               |                      |                  N/A |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A      1357      G   /usr/lib/xorg/Xorg                 77MiB |
|    0   N/A  N/A      1852      G   /usr/lib/xorg/Xorg                207MiB |
|    0   N/A  N/A      1984      G   /usr/bin/gnome-shell               81MiB |
|    0   N/A  N/A      2914      G   ...AAAAAAAAA= --shared-files       30MiB |
|    0   N/A  N/A      7139      G   ...AAAAAAAAA= --shared-files       54MiB |
|    0   N/A  N/A     10037      G   /usr/lib/rstudio/bin/rstudio       57MiB |
|    0   N/A  N/A     15250      G   gnome-control-center                3MiB |
+-----------------------------------------------------------------------------+
```

train completed, best_metric: 0.7918 at epoch: 276, total time: 80824.40817975998.



Check best model output with the input image and label




```
model.eval()
with torch.no_grad():
    # select one image to evaluate and visualize the model output
    val_input = val_ds[6]["image"].unsqueeze(0).to(device)

  # visualize the 3 channels label corresponding to this image

  plt.imshow(np.rot90(val_ds[6]["label"][i, :, :, 70].detach().cpu()))

   # visualize the 3 channels model output corresponding to this
   image

   val_output = inference(val_input)
   val_output = post_trans(val_output[0])

   plt.imshow(np.rot90(val_output[i, :, :, 70].detach().cpu()))
```

From the best_metric_model.pth (after 22h 27min training over 300 epochs on a RTX 6000 GPU):

Metric on original image spacing: 0.791255 (DiceCoef)

metric_tc: 0.8398 (tumor core)
metric_wt: 0.9139 (whole tumor)
metric_et: 0.6201 (enhancing tumor)

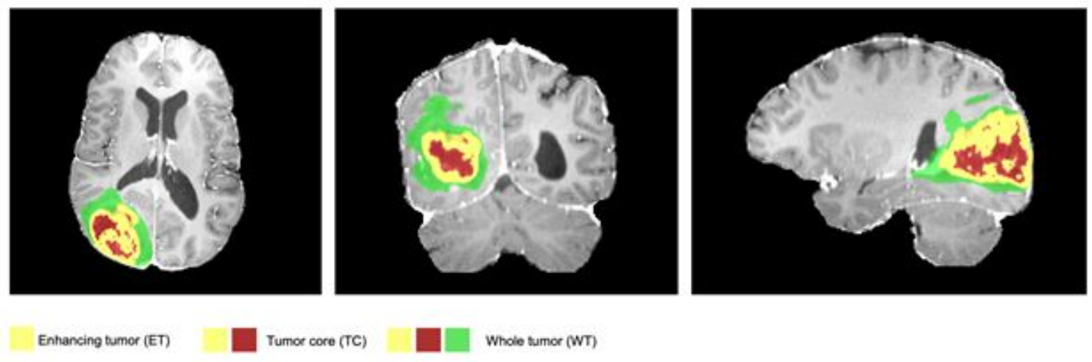
https://nbviewer.org/github/MMIV-ML/ELMED219-2022/blob/main/Lab4-BRATS/ELMED219-2022-Lab4-BRATS-segmentation-3D-MONAI.ipynb

Note on BraTS 2021

- **NVIDIA Data Scientists Take Top Spots in MICCAI 2021 Brain Tumor Segmentation Challenge**:

- Optimized U-Net for Brain Tumor Segmentation – Rank #1 (based on nnU-Net winner of BraTS 2020 by Isense et al.)

- SegResNet: Redundancy Reduction in Semantic Segmentation of 3D Brain MRIs – Rank #2
  The main model is the SegResNet architecture from MONAI, a standard encoder-decoder based convolutional neural network (CNN) similar to U-Net.



A typical segmentation example with predicted labels overlaid overT1c MRI axial, sagittal and coronal slices. The whole tumor (WT) class includes all visible labels (a union of green, yellow and red labels), the tumor core (TC) class is a union of red and yellow, and the enhancing tumor core (ET) class is shown in yellow (a hyperactive tumor part).

https://github.com/NVIDIA/DeepLearningExamples/blob/master/PyTorch/Segmentation/nnUNet/notebooks/BraTS21.ipynb

https://github.com/NVIDIA/clara-train-examples/blob/3df8a31b1c5e37e134ff4588232c41a27192a56b/PyTorch/NoteBooks/GettingStarted/GettingStarted.ipynb