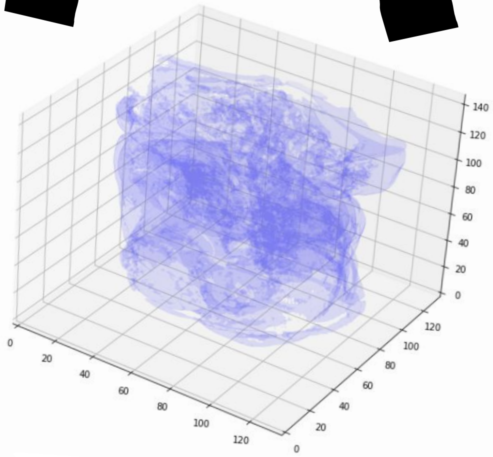
Detecting stomach and Bowels (intestines) in MRI scans of cancer patients



Collection of MRI's over time

Vikram (B19CSE098)

Under:

Dr. Binod Kumar

Assistant Professor,
Electrical Engineering,
IIT Jodhpur

Motivation

- 5 million people were diagnosed with a cancer of the Gastro-intestinal tract worldwide

- 50% are eligible for radiation therapy,
  - High X-ray delivered 10-15 minutes/day for 1-6 weeks.

- Manually outline the position of the stomach and intestines
  - prolong treatments from 15 minutes a day to an hour a day
  - difficult for patients to tolerate for patient

Unless there is some technique to automatically detect stomach and intestines.

Solution and Future Use

- Solution to this problem is to create a deep learning model to automate segmentation process.

- It will make cancer patient treatment faster
- get more effective treatment with less side effects
- better long-term cancer control

Dataset

UW-Madison GI Tract Image Segmentation (Kaggle)

CSV

contain rows with id, class and predicted columns

Total no. of unique images: 101739

	id	class	segmentation
194	case123_day20_slice_0065	stomach	28094 3 28358 7 28623 9 28889 9 29155 9 29421 ...
197	case123_day20_slice_0066	stomach	27561 8 27825 11 28090 13 28355 14 28620 15 28...
200	case123_day20_slice_0067	stomach	15323 4 15587 8 15852 10 16117 11 16383 12 166...
203	case123_day20_slice_0068	stomach	14792 5 15056 9 15321 11 15587 11 15852 13 161...
206	case123_day20_slice_0069	stomach	14526 6 14789 12 15054 14 15319 16 15584 17 15...

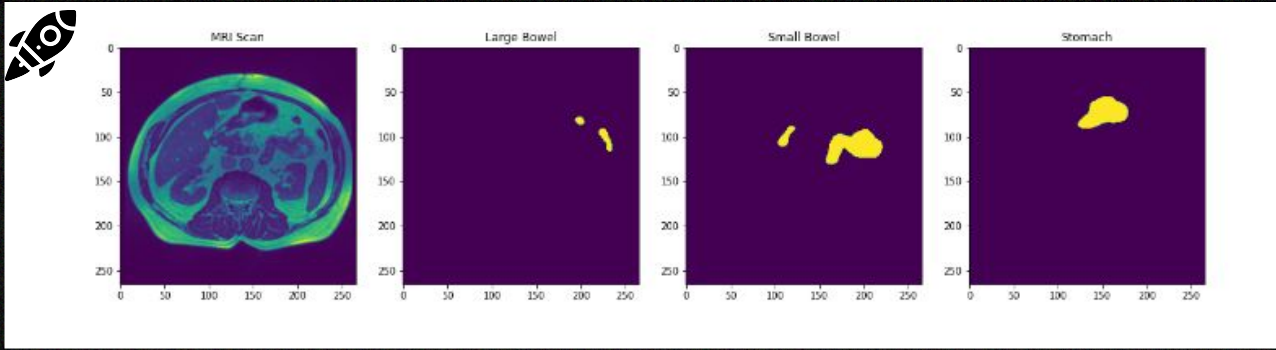
Dataset

Image Data

IMG

Along with CSV MRI scans with respective masks are provided for each scan

MRI scan and mask corresponding to different parts



Preprocessing:

Strategy 1

Tried to generate all three masks for all of the images

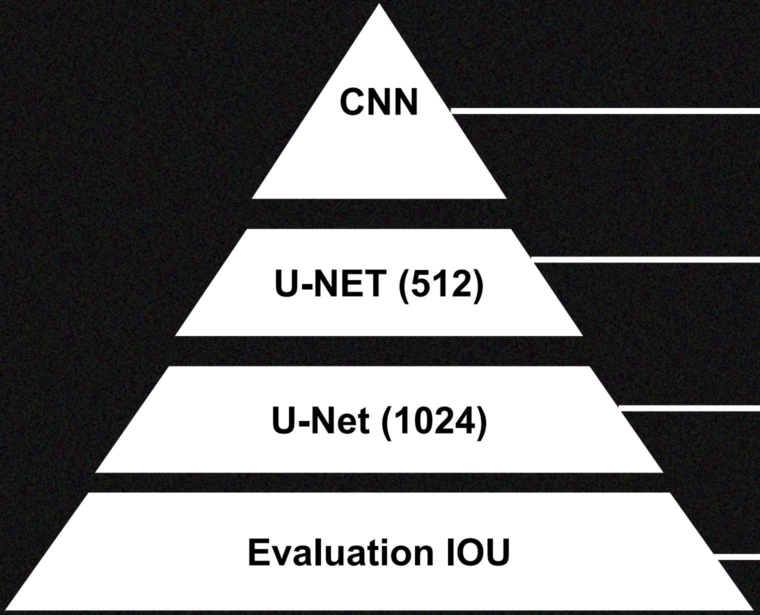
GPU memory was getting exceeded

Strategy 2

Generated masks during training for a batch of 32 images at the time

Created a csv file containing all the RLE encoded values of the masks and generated masks while training.

Model



01 Single CNN

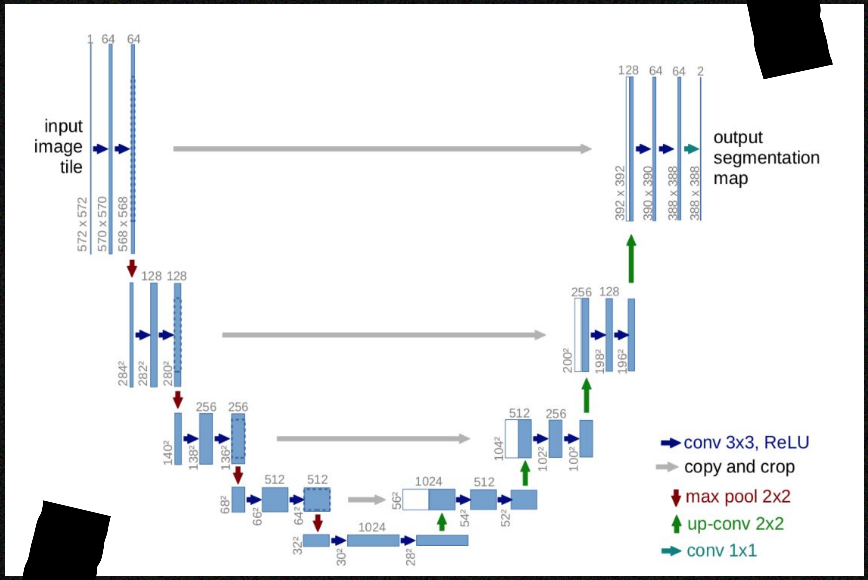
02 U-net with 512 filters

03 U-net with 1024 filters

04 For evaluation IOU is used

U-net

- Contraction Phase: Reduce spatial dimension, but increases the “what.”
- Expansion Phase :Recovers object details and the dimensions, which is the “where.”
- Concatenating feature maps from the Contraction phase helps the Expansion phase with recovering the “where” information.



Training :

- Due to low available GPU memory I generated mask images for every batch (32 images) during training

- Due to very long training time, I saved checkpoints after every 5 epochs during training

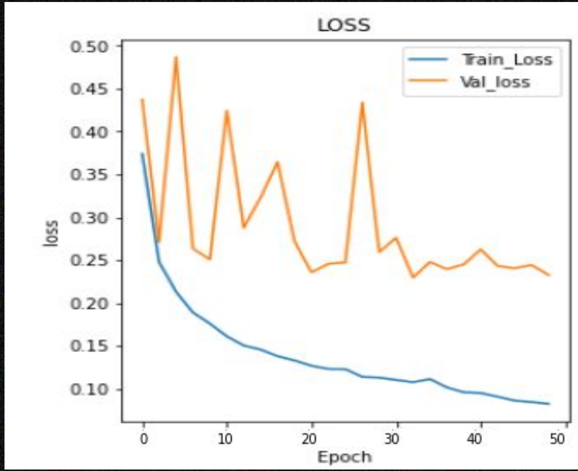
```
==============================================================
Total params: 31,042,369
Trainable params: 31,042,369
Non-trainable params: 0
--------------------------------------------------------------
Input size (MB): 0.25
Forward/backward pass size (MB): 808.00
Params size (MB): 118.42
Estimated Total Size (MB): 926.67
--------------------------------------------------------------
```

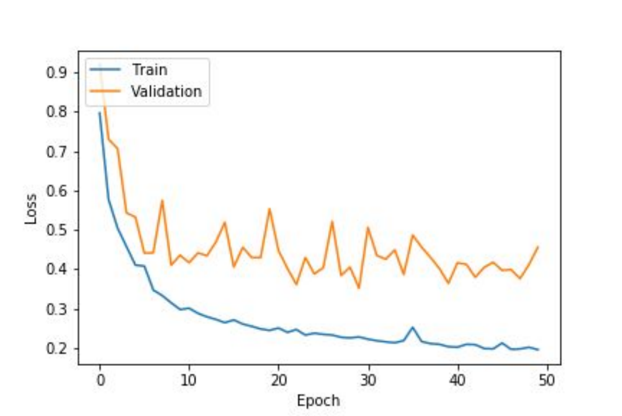
Training Loss:

Epoches	Small Bowel	Large Bowel	Stomach
Epoch 5	0.1012	0.0732	0.4400
Epoch 20	0.0301	0.0131	0.2003
Epoch 30	0.0072	0.0062	0.1031
Epoch 40	0.0063	0.0045	0.0329
Epoch 50	0.0055	0.0037	0.0146

Training Loss vs Epochs

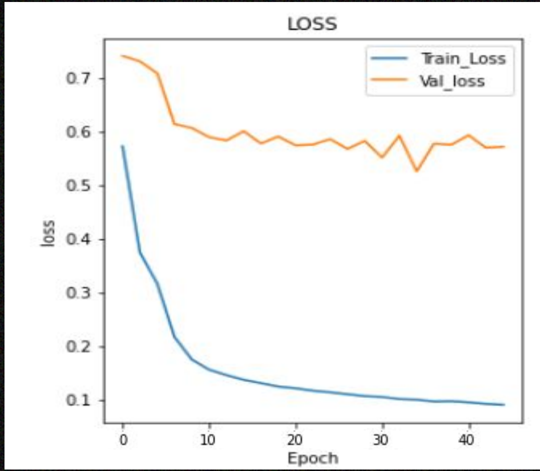
Small Bowel



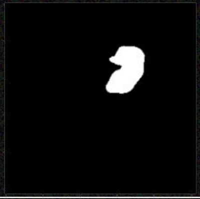
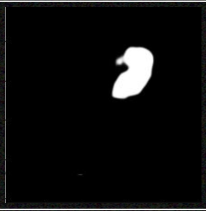
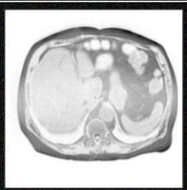
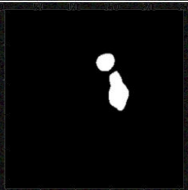
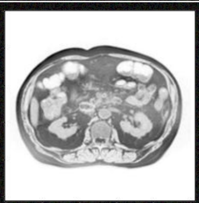
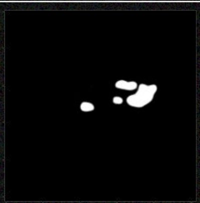


Stomach

Large Bowel

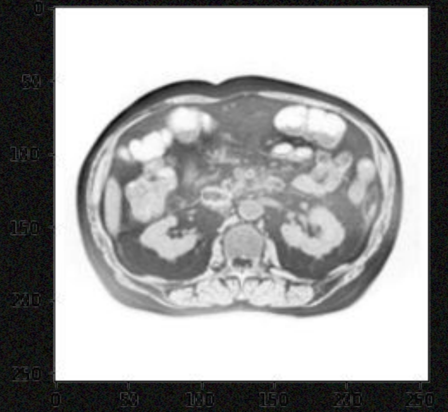


Predictions

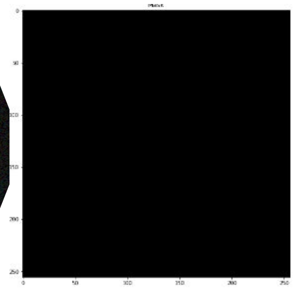
Class	MRI SCAN	Mask	Predicted mask
Stomach			
Large Bowel			
Small Bowel			

Improvement in Results

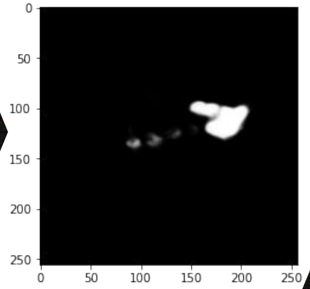
Small Bowel



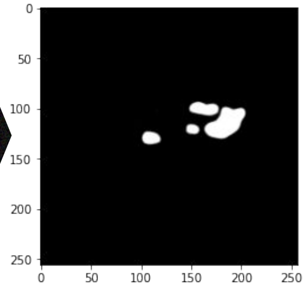
CNN predicted

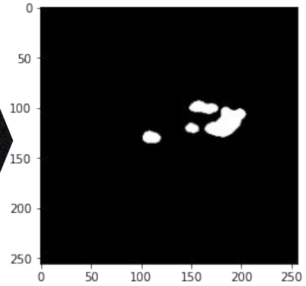


U-net 20 epoch



U-net 50 epoch

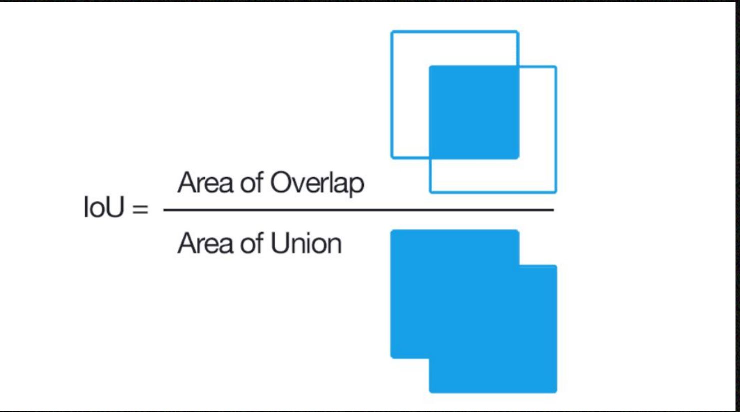




Real

Evaluation: IOU Intersection over union

Intersection over Union is an evaluation metric used to measure the accuracy of an object detector on a particular dataset.



Testing Results

Rank	Class	IOU	Loss (after 50 epochs)
1.	Stomach	0.480353	0.0046
2.	Large Bowel	0.578355	0.0037
3.	Small Bowel	0.613420	0.0055

Conclusion

- Learned and implemented UNet architecture in pytorch
- Performed image segmentation
- Results obtained after 50 epochs were decent
- With more computational resources we can converge the models even better

Thanks

Vikram (B19CSE098)