Liver Segmentation from abdominal CT Scans

Arshdeep Singh
IIT Delhi
cs5160625@iitd.ac.in

Aditya Jain
IIT Delhi
cs1160335@iitd.ac.in

1. Introduction

The extraction of liver tissue is very important in various types of disease detection. Computed tomography is used for these purpose due to high signal to noise ratio and to get better spatial resolution. However, it is tedious and time consuming to get liver regions by manual delineation from several thousand slices. Therefore, automatic methods for liver segmentation are used to solve this problem. So, given a ct scan image for patient the segmentation algorithm should be able to output for each slice which pixels belong to the liver and which belong to the background. Doing this requires semantic understanding of the image slice.

2. Related Work

Segmentation in medical images has been a famous problem and has contributed to a lot of medical applications Traditionally region based approaches were given to solve the problem in which a region is defined as collection of pixels and boundary is made from differences between the two regions. Deep learning models have achieved state of the art results but they have not demonstrated sufficiently accurate results due to the inherent properties of medical images.

One popular approach for image segmentation model is to follow an encoder-decoder architecture where we first downsample the spatial resolution of the input slice, hence developing lower resolution features which are learned to be highly efficient at discriminating between the classes and then upsample to obtain the full resolution segmentation map.

There are several ways in which we can upsample the resolution of feature map. Whereas pooling operations downsample the resolution by replacing a local area with a single value, there are several unpooling operations that distribute a single value into a higher resolution. Transposed convolution or Deconvolution is the most popular approach as they develop a learned upsampling.

U-Net [2] is one such earlier deep learning model that was proposed for this task. A plain U-Net is an encoder-decoder model with dimension of final output same as the whole image in which segmentation has to be done. Along with these several other ideas have been proposed such as using dilated convolution [4] which results in multi-scale content aggregation. Dilated convolutions also support exponential expansion of receptive field without loss of resolution. Pooling and Strided are similar concepts but they reduce the resolution. Skip connections are also commonly used in many networks used for solving this task. Skip connections from earlier layers provide the network necessary details in order to reconstruct the correct shapes for the image segmentation.

3. Proposed Model

We propose a network as a combination of an autoencoder and a detector. The major key ideas of the our model are as follow :

- **U-Net Based architecture** : U-Net have been hugely successful in medical images segmentation. They allow the model to learn the underneath represantation of the image. Since they are an encoder-decoder model it is easy to use them as an auto-encoder for the further training of our model.
- **Dilation Convolution**:Dilated Convolution allows a larger receptive field with less parameters and help the network to get a larger context of the image. This increase in receptive field helps in segmentation as segmentation depends much on the context .
- **Inception Module and Residual Connections**: Inception module [3] helps in getting more information about the image and also by using convolution of kernel size (1,1) it is doing a cross-channel correlation ignoring the spatial dimension. Residual connections help in better passing of the gradient . These techniques help learn the model better.
- **Neighbouring Context** : A CT scan image is a collection of many slices. All the slices are heavily correlated to its neighbouring slices as they form a 3D volume if combined. To include this context we take a

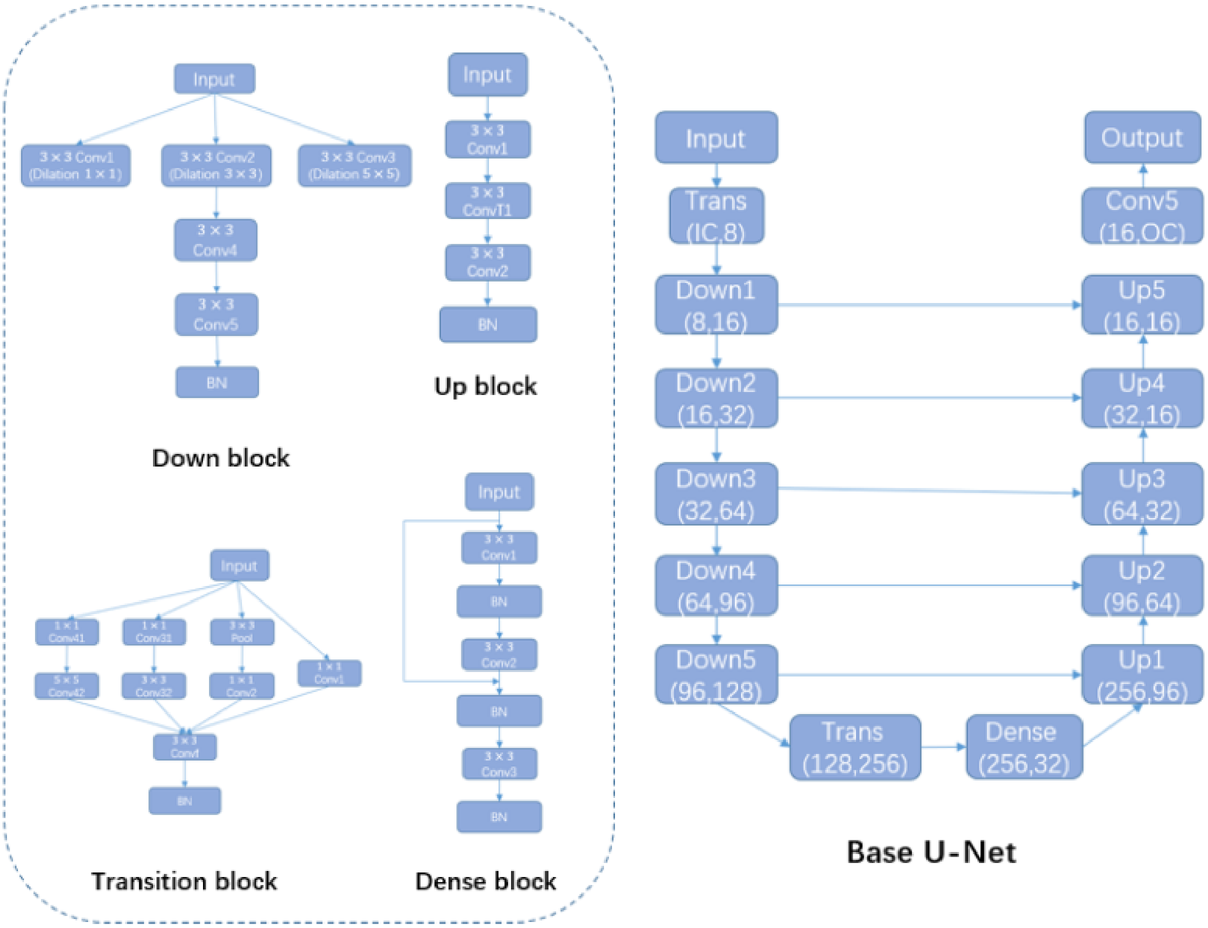

Figure 1. BaseUNet model used

combined input of some consecutive slices before and after the target slice and use that as the input of the model instead of taking only one image.

- **Bottleneck Feature Vectors** : We supervise the training of the bottleneck feature vectors on those of an already trained auto-encoder [1]. The approach of supervising the bottleneck feature vector comes from the following fact: given a pair of intensity image and label map, the bottleneck feature vector of perfectly-trained encoding U-Net and segmentation U-Net should be the same.

Using the above mentioned key idea, we create a base Unet model as shown in figure 1. Our auto-encoder is the same network as the baseUnet without skip connections. We train it on label maps for label maps and also give the bottleneck feature vectors as output. The other network which is the detector is a baseUnet model and its loss function include the binary cross entropy loss for the final segmentation and an euclidean distance between the two network's bottleneck feature vectors.

3.1. Training

First we train the auto-encoder on the label maps. Then we train the detector on the desired output along with the bottleneck feature vectors obtained from the well trained auto-encoder for each input.

4. Experiments

We experimented with two models : one with the bottleneck supervision and one without it. To conduct experiments we divided the 130 CT Scan from **LiTS** dataset, into 20 and 110 and used the first 20 ct scans as our test set to evaluate our models. We trained our models to minimize dice loss but the model did not perform well on the dice score on the test set, while the model trained to minimize the binary cross entropy loss performed better.

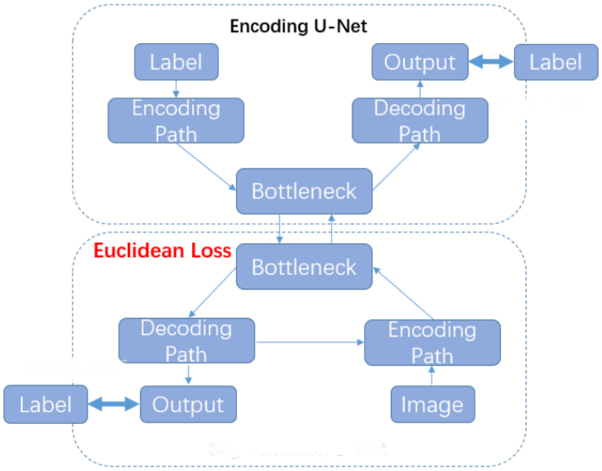

Figure 2. Training of our model

4.1. Loss functions

- Loss functions using Binary Cross-Entropy

$$L_{bce} = \sum_i y_i \log o_i + (1 - y_i) \log(1 - o_i)$$

- Dice Loss

$$L_{dice} = -\frac{2 \sum_i o_i y_i}{\sum_i o_i + \sum_i y_i}$$

4.2. Pre-processing

We pre-processed the input ct scans which contains Hounsfield Units (HU), and transformed into pixel intensities before feeding the network. The pre-processing procedure that was followed is as mentioned below,

- Setting HU values that are larger than 250 to 250 and lower than -200 to -200.

- Normalizing the values from the previous step, to 0-255.

4.3. Results

The dice score was used as the main metric for the evaluation. It was seen that the bottleneck feature loss has a correlation with the dice score since higher dice score often came with less feature vector loss. The metric results are shown in the Table 1. The model with bottleneck supervision and context outperformed the other model which showed that those designing techniques were useful for segmentation.

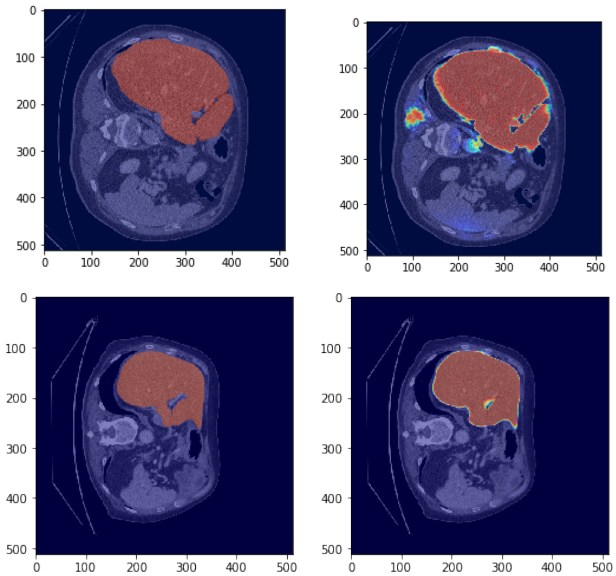

Figure 3. Predictions on the test set with model 2

No.	Model	DICE Score
1	Without Bottleneck Supervision	0.84
2	With Bottleneck Supervision and Context	**0.86**

Table 1. DICE Score versus models

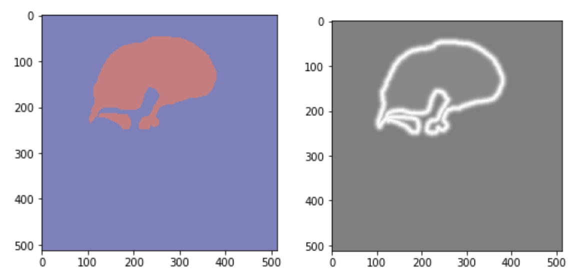

Figure 4. Weight map corresponding to a slice

5. Future Works

It has been seen that segmentation around the border are often not proper so to remove this problem we can incorporate a pixel wise weight map and use it to increase the miss classified pixels at the border more than the other pixels. We can compute the weight matrix using the following formula:

$$W = \exp \frac{-D}{2\sigma}$$

Here D is the distance map with the distance of each pixel with the closest border.

6. Conclusion

This work demonstrates how deep learning techniques can have a huge impact in the medical field. We saw how we can benefit from autoencoders by using their bottleneck feature vectors for the training for other models. Dilated convolutions , inception modules and residual networks are helpful in creating a more robust and better model. Contexts play a important role in case of sequential data and thus using consecutive images can help in getting better results.The results obtained from experimentation showed that these practices help in segmentation. In the end we also saw how using a weighted map we can improve on the cases where borders were not being predicted correctly.

References

[1] S. Li and G. K. F. Tso. Bottleneck supervised u-net for pixel-wise liver and tumor segmentation. *CoRR*, abs/1810.10331, 2018. 2

[2] O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015. 1

[3] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. *CoRR*, abs/1409.4842, 2014. 1

[4] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016. 1