DRINet for Medical Image Segmentation

CS736 Medical Image Computing (Spring'23)

Ananya Rao

Indian Institute of Technology Bombay

May 5, 2024

Table of Contents

Table of Contents

Introduction

- Medical image segmentation is a critical part of medical image analysis, the results of segmentation are used to derive quantitative measurements, for e.g. assessment of CSF(Cerebrospinal fluid) volume, or to identify and diagnose disease like brain tumor.

Introduction

- Medical image segmentation is a critical part of medical image analysis, the results of segmentation are used to derive quantitative measurements, for e.g. assessment of CSF(Cerebrospinal fluid) volume, or to identify and diagnose disease like brain tumor.
- Usage of CNNs in medical image segmentation has revolutionized medical image analysis. Training an ML model like U-net, which is one of the most well known architecture for medical image segmentation, on a large number of images allows us to predict the segmentation with high accuracy, eliminating the need of extensive manual labour.

Table of Contents

Dense-Res-Inception Net

- The U-net architecture comprises of simple convolution, pooling and upsampling layers. Such simple layers do not capture subtle differences, in terms of intensity, location, shape, and size, among different features efficiently.

Dense-Res-Inception Net

- The U-net architecture comprises of simple convolution, pooling and upsampling layers. Such simple layers do not capture subtle differences, in terms of intensity, location, shape, and size, among different features efficiently.
- DRI-Net(Dense-Res-Inception Net) is a novel CNN architecture which addreses these problems using three different blocks, namely a convolution block using dense connections, deconvolution block using residual inception blocks, and an unpooling block.

Table of Contents

Convolution block with dense connections

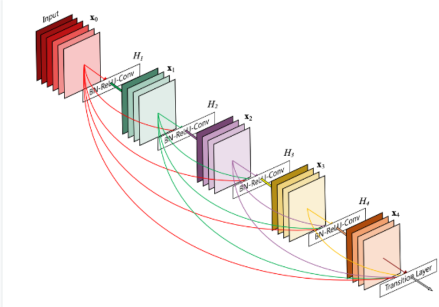

Figure: Dense connection illustration

The convolution blocks comprise the analysis path of our model, where we learn the features in input image.

Convolution block with dense connections

Why use dense connection blocks?
Here we mention two main advantages of using dense connection blocks in our model.

- Vanishing gradient problem
  Generally it is difficult to propagate gradient backward in large and complex models as it tends to vanish in the lower levels. In dense connections, a layer is used in all the subsequent layers therefore it becomes much more convenient to propagate gradient to these layers.

Convolution block with dense connections

Why use dense connection blocks?

Here we mention two main advantages of using dense connection blocks in our model.

- Vanishing gradient problem
  Generally it is difficult to propagate gradient backward in large and complex models as it tends to vanish in the lower levels. In dense connections, a layer is used in all the subsequent layers therefore it becomes much more convenient to propagate gradient to these layers.
- Employing dense connections allows us to reuse features, some of which might have got lost in the previous layers, ensuring maximal data extraction and efficient pattern learning.

Convolution block with dense connections

Structure

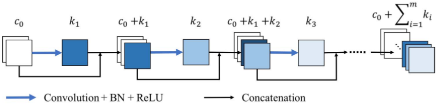

Figure: Dense Connection Convolution Module

Forward propagation in dense block:

$$x_{l+1} = f(x_l) \circ x_l$$

Each convolution block uses three layers of dense connections and in total we have used four convolution blocks, based on the implementation details of the concerned paper.

Deconvolution block with residual inception module

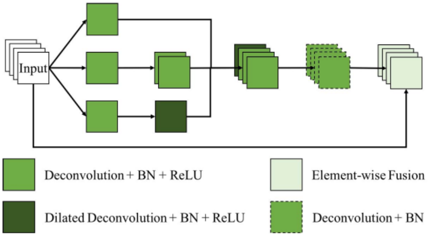

Figure: Inception block

These modules comprise the synthesis path of our DRI-Net model. The main goal of the synthesis path is to aggregate feature maps from different branches.

Deconvolution block with residual inception module

- The main reason behind using Inception modules for the synthesis path is that it approximates local sparse structures from incoming branches using several convolution layers to capture features based on their spatial distributions.

Deconvolution block with residual inception module

- The main reason behind using Inception modules for the synthesis path is that it approximates local sparse structures from incoming branches using several convolution layers to capture features based on their spatial distributions.
- To be precise, smaller filter sizes are used to detect spatially concentrated features whereas larger filter sizes are used to detect spatially spread out features.

Deconvolution block with residual inception module

- The main reason behind using Inception modules for the synthesis path is that it approximates local sparse structures from incoming branches using several convolution layers to capture features based on their spatial distributions.
- To be precise, smaller filter sizes are used to detect spatially concentrated features whereas larger filter sizes are used to detect spatially spread out features.
- One more plausible advantage of inception module is computational cost. The architecture involves concatenating the outputs of convolutions which might result in a blow up withing a few layers. To avoid that, the architecture employ bottleneck convolution before using larger filter sizes in order to preserve the sparse structure as compressed data is hard to process.

Unpooling block

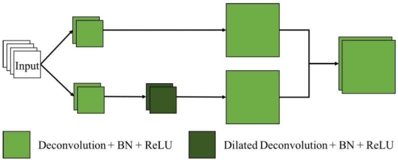

Figure: Unpooling block

An unpooling block is like a mini inception module. It convolves the input with two different filter sizes, to capture a range of spatial distribution, upsamples and then concatenate them which becomes the input for next inception block.

Table of Contents

Implementation

Following are some implementation details which are inspired from the paper being implemented.

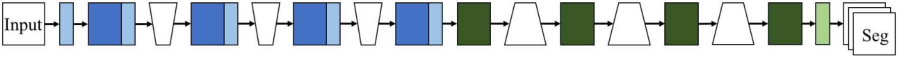

Figure: Visual illustration of DRI-Net model

- Four Dense Connection convolution blocks are used, each followed by a simple convolution layer, each consisting of three dense connection convolution layers, with pooling layers between them.
- Four Residual Inception blocks are used, with unpooling blocks between them, and the last layer is followed by a simple deconvolution layer.
- This implementation utilizes tensorflow platform.

Hyper-parameters

Following are the hyperparameters that are tuned inspired from the paper being implemented.

- Adam optimizer is used with following parameters: $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1\text{e-}8$.
- Learning rate, $\epsilon = 1e - 3$, is used.
- Common number of channels used is 12.
- Size of the training data is 1000 images, 512x512 each. Number of epochs used is 50. The dataset consists of images with meningioma mask, glioma mask and pituitary tumor mask.

Table of Contents

References

- Paper implemented: DRI-Net Paper
- Dataset