FULLY CONVOLUTIONAL NEURAL NETWORKS FOR VOLUMETRIC MEDICAL IMAGE SEGMENTATION

Names : Mahak Bafna

Riya Ralhan

Mohana Siddhartha Chivukula

AIM OF THE RESEARCH PAPER

- In this work we propose an approach to 3D image segmentation based on a volumetric, fully convolutional neural network.
- This paper is purely based on convolutional neural network.
- We introduce a novel objective function, that we optimize during training, based on Dice coefficient. In this way we can deal with situations where there is a strong imbalance between the number of foreground and background voxels.
- We show in our experimental evaluation that our approach achieves good performances on challenging test data while requiring only a fraction of the processing time needed by other previous methods.

WHAT IS CNN ?

A CONVOLUTIONAL NEURAL NETWORK (CNN) IS A TYPE OF ARTIFICIAL INTELLIGENCE USED IN IMAGE RECOGNITION AND PROCESSING THAT IS SPECIFICALLY DESIGNED TO PROCESS PIXEL DATA.

CNNS ARE POWERFUL IMAGE PROCESSING, ARTIFICIAL INTELLIGENCE THAT USE DEEP LEARNING TO PERFORM BOTH GENERATIVE AND DESCRIPTIVE TASKS, OFTEN USING MACHINE VISION THAT INCLUDES IMAGE AND VIDEO RECOGNITION

Segmentation

Segmentation is a highly relevant task in medical image analysis

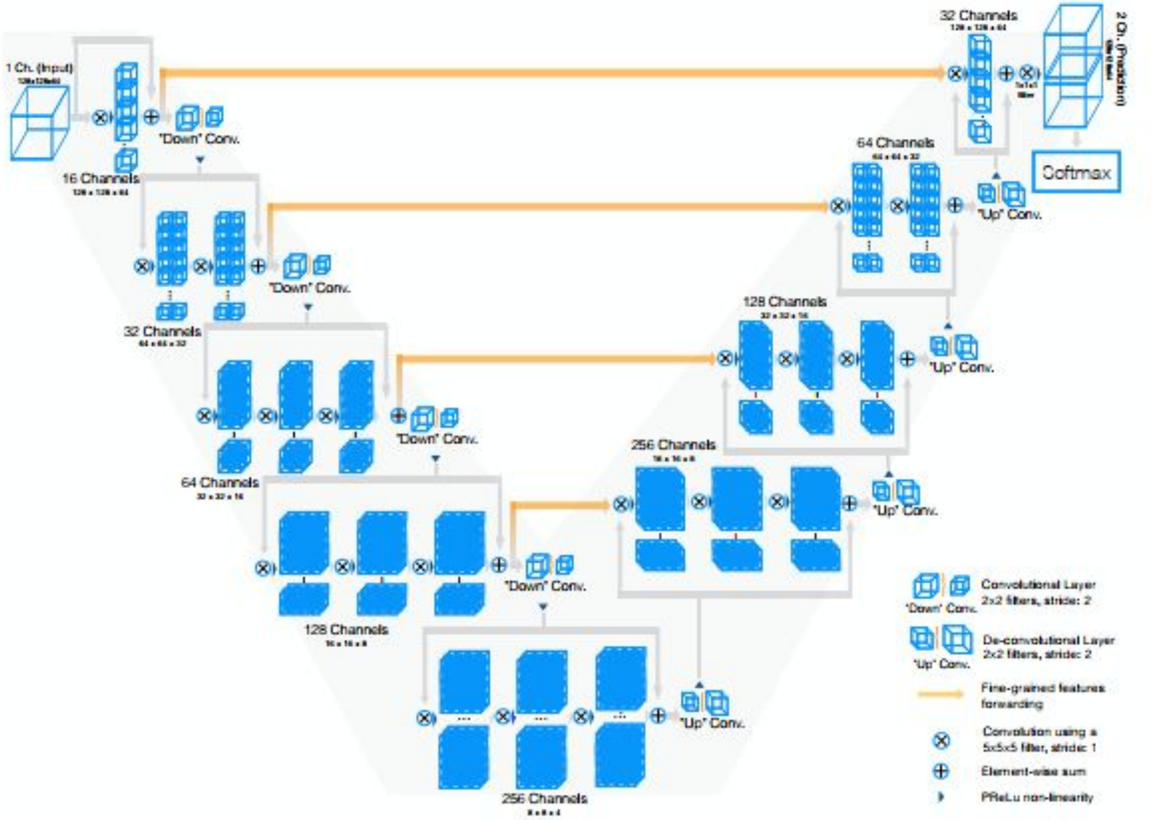
It is often necessary to perform tasks such as visual augmentation, computer assisted diagnosis , interventions and extraction of quantitative indices from images

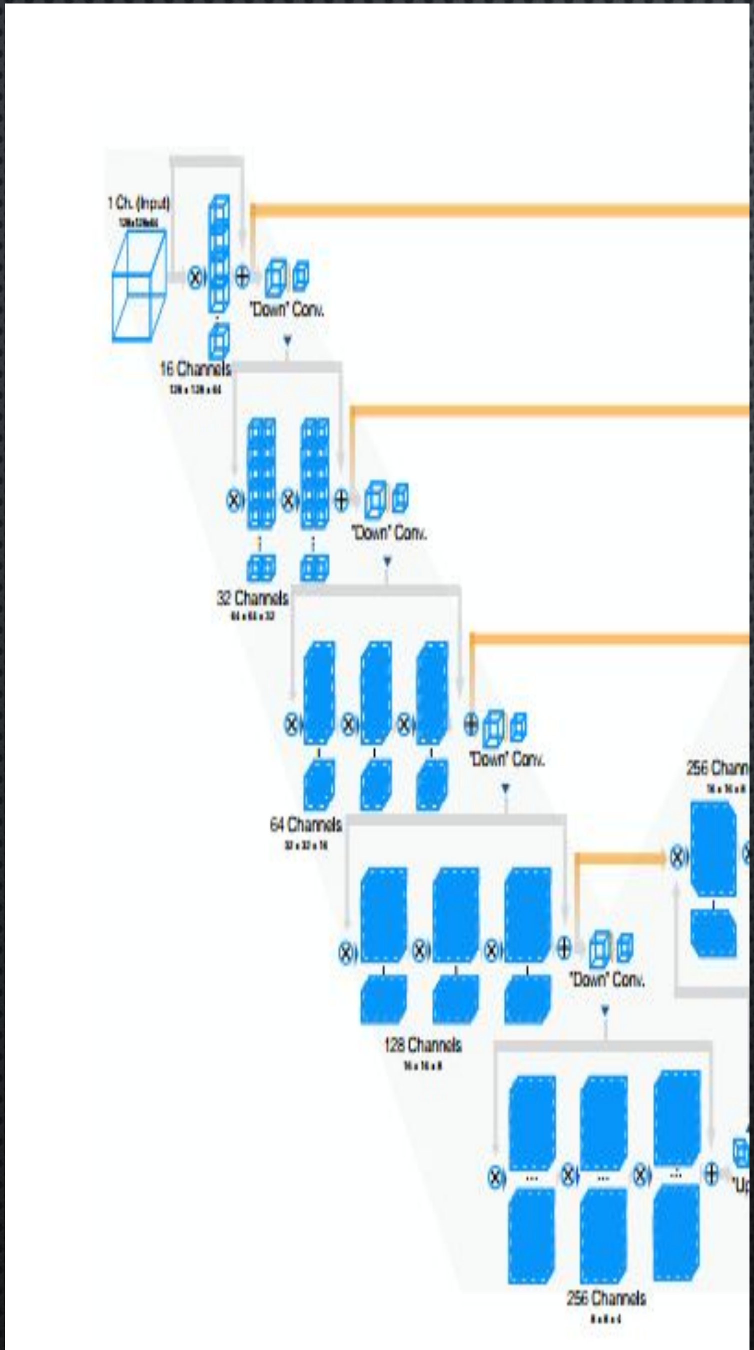
In particular, since diagnostic and interventional imagery often consists of 3D images, being able to perform volumetric segmentations by taking into account the whole volume content at once, has a particular relevance

- In this work, we aim to segment prostate MRI volumes.
- This is a challenging task due to the wide range of appearance the prostate can assume in different scans due to deformations and variations of the intensity distribution.
- Moreover, MRI volumes are often affected by artefacts and distortions due to field inhomogeneity

METHOD

- Here, we perform convolutions aiming to both extract features from the data and at the end of each stage reduce its resolution by using appropriate stride.


1 Ch. (Input)
16 Channels
"Down" Conv.
32 Channels
"Down" Conv.
64 Channels
"Down" Conv.
128 Channels
"Down" Conv.
256 Channels
"Up" Conv.
256 Channels
"Up" Conv.
128 Channels
"Up" Conv.
64 Channels
"Up" Conv.
32 Channels
2 Ch. (Prediction)
Softmax
Convolutional Layer
2x2 filters, stride: 2
"Down" Conv.
De-convolutional Layer
2x2 filters, stride: 2
"Up" Conv.
Fine-grained features forwarding
Convolution using a 5x5x5 filter, stride: 1
Element-wise sum
PReLu non-linearity



<u>Left side of the network :</u>

The left side of the network architecture represents the compression path

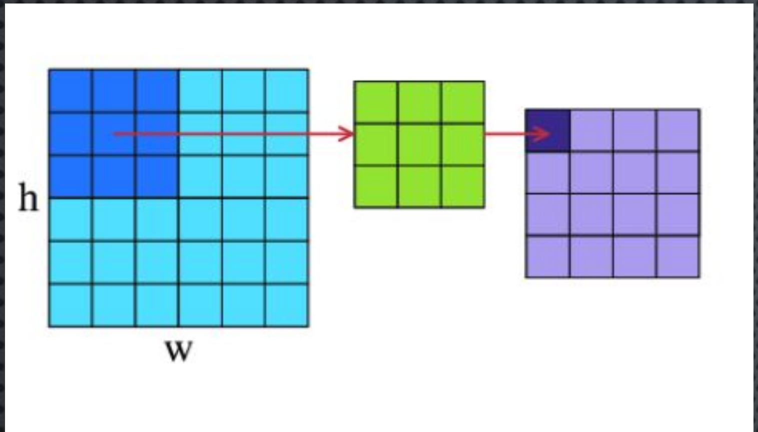
The network is divided into different stages and each stage operates at different resolution.

Each stage undergoes one to three convolution layers depending on the resolution of the sample.

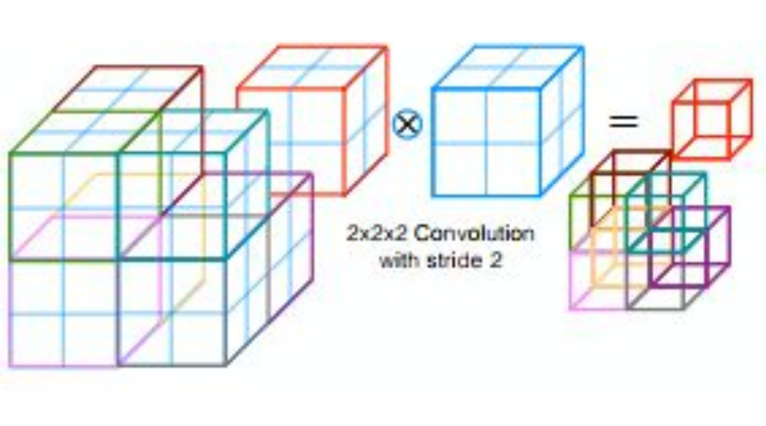
At the end of Each stage, we formulate it such that it learns a <u>residual function</u>

- RESIDUAL FUNCTION: THE INPUT OF EACH STAGE IS
- (A) USED IN THE CONVOLUTIONAL LAYERS AND PROCESSED THROUGH THE NON-LINEARITIES AND
- (B) ADDED TO THE OUTPUT OF THE LAST CONVOLUTIONAL LAYER OF THAT STAGE IN ORDER TO ENABLE LEARNING A RESIDUAL FUNCTION.

- THIS ARCHITECTURE ENSURES CONVERGENCE IN A FRACTION OF THE TIME REQUIRED BY A SIMILAR NETWORK THAT DOES NOT LEARN RESIDUAL FUNCTIONS.



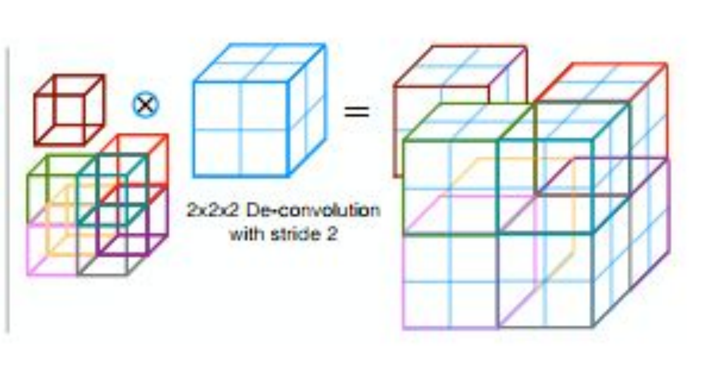
Example of a 2-D convolution



Example of a 3-D convolution

The convolutions performed at each stage use volumetric kernels of size 5x5x5 voxels.
As the stages go on and on the convolution is done with 2x2x2 voxels.
The strategy here used is like pooling layers (usually max pooling) but it is not used in this case as at the end of each stage of convolution, as the feature maps double up and this results in having a smaller memory footprint during the decompression phase which is going to be done in the right side of the network diagram.
Down sampling allows us to reduce the size of the signal presented as input and to increase the receptive field of the features being computed in subsequent network layers. Each of the stages of the left part of the network, computes a number of features which is two times higher than the one of the previous layer.

- THE RIGHT PART OF THE NETWORK DECOMPRESSES THE SIGNAL UNTIL ITS ORIGINAL SIZE IS REACHED
- THE RIGHT PORTION ACTUALLY EXTRACT FEATURES AND EXPANDS THE SPATIAL SUPPORT OF THESE LOWER RESOLUTION FEATURE MAPS TO GATHER NECESSARY INFORMATION FOR VOLUMETRIC SEGMENTATION
- AFTER EACH STAGE OF THE RIGHT PORTION OF THE CNN, A DE-CONVOLUTION OPERATION IS EMPLOYED IN ORDER INCREASE THE SIZE OF THE INPUTS FOLLOWED BY ONE TO THREE CONVOLUTIONAL LAYERS INVOLVING HALF THE NUMBER OF $5 \times 5 \times 5$ KERNELS EMPLOYED IN THE PREVIOUS LAYER.



Example of De-convolution

- During the decompression stages in the right side, there might be an issue of loss of quality or loss if detail.
- To prevent that , the features extracted from the left side of the cnn are used to improve the final contour prediction.
- we forward the features extracted from early stages of the left part of the CNN to the right part.. In this way we gather fine grained detail that would be otherwise lost in the compression path and we improve the quality of the final contour prediction. We also observed that when these connections improve the convergence time of the model.

We report in Table 1 the receptive fields of each network layer, showing the fact that the innermost portion of our CNN already captures the content of the whole input volume.

Layer	Input Size	Receptive Field	Layer	Input Size	Receptive Field
L-Stage 1	128	$5 \times 5 \times 5$	R-Stage 4	16	$476 \times 476 \times 476$
L-Stage 2	64	$22 \times 22 \times 22$	R-Stage 3	32	$528 \times 528 \times 528$
L-Stage 3	32	$72 \times 72 \times 72$	R-Stage 2	64	$546 \times 546 \times 546$
L-Stage 4	16	$172 \times 172 \times 172$	R-Stage 1	128	$551 \times 551 \times 551$
L-Stage 5	8	$372 \times 372 \times 372$	Output	128	$551 \times 551 \times 551$

Dice Loss Layer

In this work we propose a novel objective function based on dice coefficient, which is a quantity ranging between 0 and 1 which we aim to maximise. The dice coefficient D between two binary volumes can be written as

$$D = \frac{2\sum_i^N p_i g_i}{\sum_i^N p_i^2 + \sum_i^N g_i^2}$$

$$\frac{\partial D}{\partial p_j} = 2\left[\frac{g_j\left(\sum_i^N p_i^2 + \sum_i^N g_i^2\right) - 2p_j\left(\sum_i^N p_i g_i\right)}{\left(\sum_i^N p_i^2 + \sum_i^N g_i^2\right)^2}\right]$$

we obtain results that we experimentally observed are much better than the ones computed through the same network trained optimising a multinomial logistic loss with sample re-weighting

Training

The CNN is trained end-to-end on a dataset of prostate scans in MRI.

In the process of training the dataset, the following steps are followed

upload dataset, convolutional layer, pooling layer(which is not used in our case), …..

the datasets of the prostate scans in MRI, all the volumes have size 128x128x64 voxels

during every iteration, different deformed versions of the training images are used.

moreover, the intensity is being adapted by histogram matching

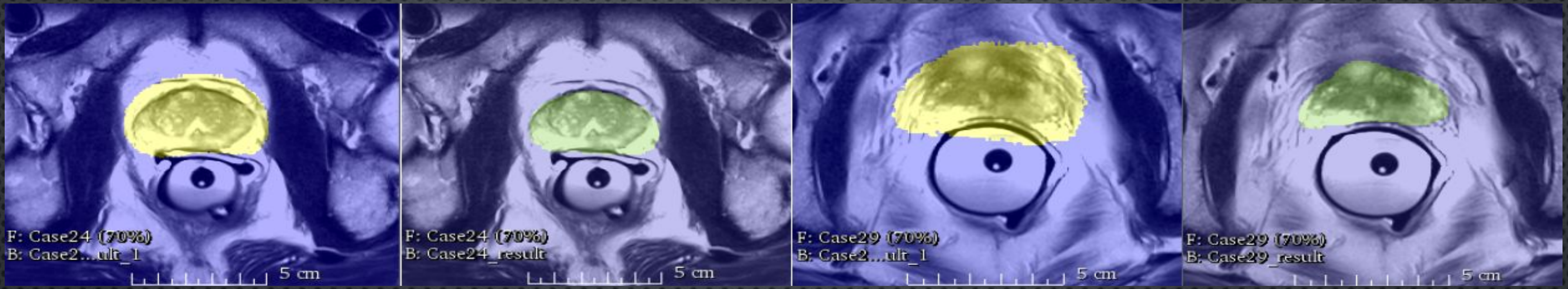
Testing

A new unseen MRI volume is then processed through the network

the output of the last convolutional layer after soft-max consists of the probability map.

the voxels having probability >0.5 are considered a part of the anatomy

RESULT AND CONCLUSION



We presented and approach based on a volumetric convolutional neural network that performs segmentation of MRI prostate volumes in a fast and accurate manner.

We introduced a novel objective function that we optimise during training based on the Dice overlap coefficient between the predicted segmentation and the ground truth annotation.

Our Dice loss layer does not need sample re-weighting when the amount of background and foreground pixels is strongly unbalanced and is indicated for binary segmentation tasks