Kidney Tumor Segmentation Using Deep Learning

Prashant Jadiya

Marwadi Education Foundation's Group of Institutions, Department of Information Technology, Gouridad-Rajkot, Gujarat, India
jadiyap[at]ymail.com

Abstract: ***Proper segmentation of Kidney Tumors can assist doctors to detect and diagnose diseases. When it comes to segmentation, U-Net is arguably the most successful segmentation architecture in the medical domain. In this paper, we address the challenge of simultaneous semantic segmentation of kidney tumor and proposed multidimensional 3D U-Net with an attempt to improve it with V-Net and Auto-Encoder architecture. Due to marginally higher dice scores, it was very difficult to choose architecture that will be accurate for segmentation. We did experiment for training of 190 cases that is provided by Kidney Tumor Segmentation Challenge database.***

Keywords: Medical Segmentation, Kidney Tumor Nephrometry Segmentation, Multidimensional images, Neural Network, Auto-Encoder

1. Introduction

With more than 400,00 kidney cancer cases in 2018 resulting in more than 175,000 deaths [1], for 208,000 diagnoses and 102,000 deaths in 2002[2]. Proper segmentation of kidney tumor in CT scan (Computed Tomography) images is cumbersome and challenging task due to randomized irregular dimensions, similar structure and diverse shapes. Many doctors take years of experience to correctly identify the tumor. Kidney cancers are not uncommon in the contemporary era. Nephrectomy, partial nephrectomy and radical nephrectomy are the most common treatment for Kidney Tumors. Although detecting in the late stages are considerably easy through other tests, the objective of diagnosis is mainly upon detecting the presence of tumor at quite primary stages from just CT scan results.

By fortune of publicly available databases, semantic segmentation is the most popular research topic in the medical image computing domain. Despite the broad availability of abdominal CT scan data, some of which also includes the kidneys are segmentation targets, no public dataset with kidney tumor labels has been available until now. This results in a relatively low number of segmentation algorithms that especially designed for segmenting kidney tumors. This challenge KiTS19 aims at tackling this efficiency by providing 210 high quality annotated CT scans for training set and 90 CT scans for testing set [3].

Various factors such as the region of cancer, the potential growth and the current stage of the cancer determines whether surgery might be done to remove either the tumor with adjoining kidney cells with potential spreading or the entire kidney itself. A major part of the analysis of the cancer is its localization and determination of the size of the dice (representing the tumor). Traditional methods of identification and localization of the cancer included well trained and experienced doctors evaluating the image of the Kidney generated by CT or MRI scans along with various other test results. Visualization techniques like MRI (Magnetic Resonance Imaging) and CT (Computed Tomography) can contain enough information to identify positive tumors. But this highly depends on the experience and competency of the consulting doctor. This outcome can be made more precise by applying Deep Learning technology to automatically identify and segment the tumor present in the kidney. With the success rate of U-Net as in 3D image segmentation in medical domain is based on 3D variants of U-Net [4,5]. While U-Net is thereby commonly augmented using residual [6] and dense [7] connections, recent work has achieved excellent results using just a plain U-Net architecture [7], questioning the inevitability of extensive architecture research in the medical domain.

This research paper picks up the experiments that we have done towards segmenting the tumorous area of kidney from CT scan images using Deep Learning Algorithms. All the experiments are done in context of the KiTS19 challenge including collecting and understanding the data-set, preprocessing and methodology used for research and final results have been elaborated below.

2. Related Work

In the previous years, there have been various models to segment cancer areas from CT scan images of kidneys, few of them being the use of Crossbar-Net [3] or Hybrid Level Set Method [4].

A method proposed to segment images and detect edges by author named Urvey in 2014. That method was used to detect damaged tissue and removing a noise by filters. It converts grey-scale images from MRIs, or CT scans through the image segmentation process. Then in 2017, Malathi proposes a method of segmentation of CT scan images of kidney tumor from patients' data. In that research, RUSBoost and decision trees were used. That enables the solution to reginal classification problems.

'Automatic detection and segmentation of kidney using Random Forest and Template Deformation from 3D CT scans', by Rémi Cuingnet of Philips Research Medisys, France, gives deep insight on the basics of kidney segmentation from the CT scans. Coarse-to-fine strategy is utilized so that kidney can be separated with random forests. These are processed with a cascade of classic regression forests, such that the primary positions are identified with global contextual information. Then probabilistic segmentation of kidneys is obtained using classification forest. Ultimate segmentation is done using template 'deformation-algorithm' run using these kidney's probability-mappings.

3. Preliminary Experiments

With a data-set accounting to around 80GB, it was a challenge to pre-process the data as a single set and feed to the model named V-Net, U-Net and Auto-Encoder. First approach was V-Net, but due to some loss function issue we migrated to U-Net and Auto-Encoder, this both played a major role in this research.

A. Gathering Data

The data-set for training and testing was obtained from the KiTS19 Challenge, which released CT scan images of 210 unique kidney cancer patients who underwent partial or radical nephrectomy at the University of Minnesota Medical Center between 2010 and 2018. The data acquired was from the GitHub account managed by KITS19 Challenge. The images and masks were obtained both in 'nii.gz' extension i.e. NIFTI - GNU Zipped Neuroimaging Informatics Technology Initiative-1 Format.

B. Data Pre-Processing

The images obtained were processed using the 'nibabel' library. The images obtained were of coordinates like (slice spacing, width spacing, height spacing). For train and test split of dataset we tried 70% for training and 30% for testing data.

There are four major steps of Data pre-processing in this process.

Resampling The coordinates of each image would vary and was not a constant value. This was then reshaped using the 'reshape' method to a size of 512x512. A sample of the image and mask of one of the data samples have been obtained from taking one slice form whole slice spacing. And another approach was to use 'med2image' library that will extract slices of CT scan images, to desired format of image that can be jpg, bmp or png.

Normalization In order to include more target pixels and take a greater number of inputs in batches we linearly changed raw data to the [0,255]. This can make image contrast relatively higher, making easier to distinguish between target and non-target organs.

Mask We extracted mask as in each slice. The mask is used for GT ground truth to calculate loss and dice score.

Augmentation For avoiding overfitting as much as possible we adopted simple operations like mirroring, scaling and tilting to increase diversity of data-set.

4. Methodology

We have used U-net model with default loss functions and Auto-encoder model with custom loss and metric functions.

A. U-Net Model

During the slicing of data, it should be in compressed format otherwise there will be space complexity as well in feeding the model so Preprocessing of data is important task of the whole research.

This U-net architecture consists of 40 lac parameters and around 40 layers in that. And by binary cross entropy loss function we had got better result and also tried for categorical cross entropy.

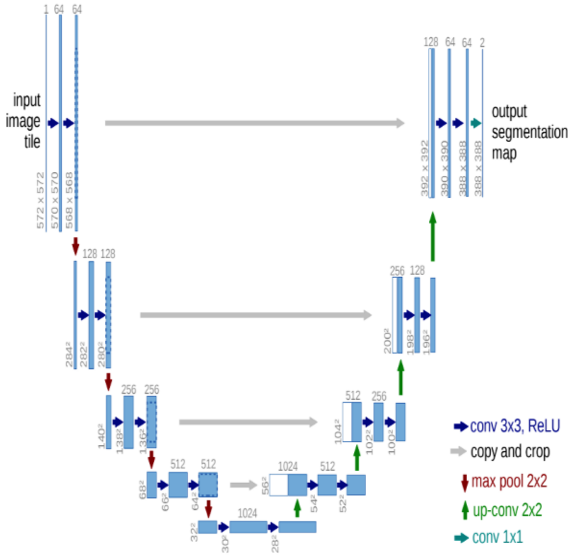

Figure 1: Architecture of U-Net

There is a large consent that it requires very much sized data that needs annotated images as training examples as the data layers. So, that needs data augmentation techniques should be used there. It will give more accurate results for all. The main use of CNN layers is to classify tasks, where the output labels will be in a unique class label. But, in various tasks, usually in biomedical scan results, the model gives the required output such as localization and there could be pixel by pixel generation of output. Moreover, many of the scan results are usually beyond a single label classification in the biomedical tasks. [4, 7, 10].

This U-net architecture is built on the FCN and modified as better segmentation in way that it needs better results in medical imaging. It uses excessive data augmentation techniques will be there for deformations of training images. This network will make us to learn in-variance to such without the trained transformations to the annotated images.

U-net architecture has two parts referring to figure 1:

- Encoder or compression part (left-side)
- Decoder or expansion part (right-side).

Compression part consist of a filter of size (3x3) convolutions and then by the o/p function, ReLU. After a max pooling operation of (2x2) with stride of 2 is done to down-sample. At all down-sampled steps, we have doubled the number of feature channels. It captures the contexts via this compact feature map. That expanded path consists of feature map and a convolution layer of (2x2) and then it will take the half no of features from the cropped feature map by the ReLU. The up-sampling of the feature dimension is done to meet the same size as the block to be concatenated on the left.

While this process the cropping will be vital since the loss with the pixels' borders are more. In the final layer, a convolution layer that is of (1x1) is used to select all 64-components features vectors to desired number of classes. Data augmentation is done by creating more deformations on each training-images that will allow our model to learn where a variance to this deformation is formed, without a requirement of seeing the transformations in the labeled image-corpus. This is a serious task in biomedical image segmentation where irregularities are the most common variation in any organ of the human body, in this case, being the Kidney.

B. Auto-encoder Model

An Auto-encoder is a deep learning network that is designed to train a model in such a way that it gives a reconstructed form of input as the prediction of the model. Theoretically it has 3 stages: i/p stage, encoded hidden stages and decoding stage.

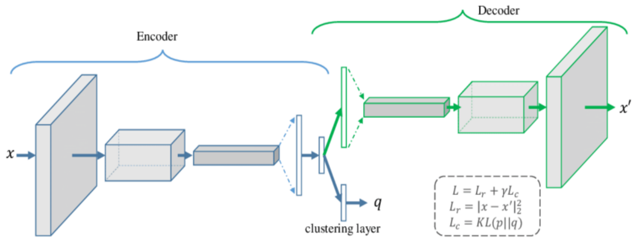

Figure 2: An Auto-Encoder

Basically, it composes of 2 identical Deep Belief Networks that has 4-5 deep layers representing with the encoding with halves the network and the other set of 4/5 layers that makes the decoded half. Internally, it will take a hidden-layer that will be describing a code meant for representing the input shapes as an encoder function lies in h equals to f(x) and then a decoder function will produce a reconstructed g(h). The auto-encoders has not been designed as not to learn copy the features in perfect way. Usually it will resist all the ways and let allow them to copy only approximate values of the i/p and copy only input that resembling training-data. Since the network will be forced to prioritize in such aspects of the input that can be specifically relatable, thus it can learn useful properties.

The Convolutional Auto-encoder model takes a single channel input of dimension (1x512x512x1). The Structure of the model is listed as below referring to figure 2:

Encoder: It has 3 blocks of convolution; each block has a convolution layer followed a batch normalization layer. And max-pooling layer is used both after the first and second convolution blocks.

- Convolution block 1: 32 filters of size (3x3), which is later down-sampled using max-pooling layer,
- Convolution block 2: 64 filters of size (3x3), with another down-sampling layer at the final stage,
- The o/p of encoder stage will have a total of 128 filters each of (3x3) size.

Decoder: A total of two convolution blocks, each block with convolution layer that is succeeded by normalization layer. Up sampling layer is used when that data after first and second convolution block.

- Block 1: 128 filters of (3x3) size followed by up- sampling layers,
- The 2nd block has 64-filters of (3x3) size succeeded by up-sampling layer,
- The third final layer of encoder will have 1 filter with size (3x3) which will reconstruct back the input having a single channel.

Max pooling layer will do down-sampling with the input like two times and each time you use it, while the up-sample the input by two times each time it is used.

5. Results

We got the training accuracy obtained for the whole model was found to be 98.25%. The validation accuracy of the model was estimated as 97.92%, for U-net. But the model was not able to predict isolated tumors rather they skewed over false predictions.

Auto-Encoder performed with a lower accuracy of 75% but predication was way better than U-net because of the custom loss function and Dice Coefficient Metric function.

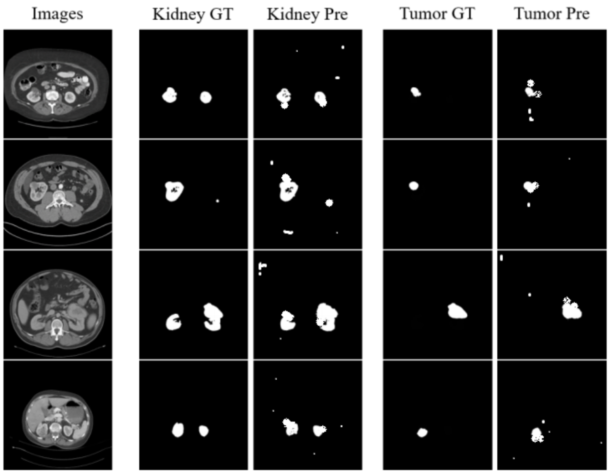

Figure 3: Results

6. Conclusion and Future Work

Although the U-net architecture has been identified to give very good performance on various different biomedical classification problems with good accuracy. Auto-Encoder is predicting better problem-oriented results with proper detection of tumor. Data-augmentation using elastic deformations helps in reducing the limited data problem, since it only needs few labeled images to result in a very reasonable training time, but it is not continuously predicting kidney and tumor both but instead it is predicting the areas where tumor is not present. We are sure that better architectures can be implemented in the coming future with more databases so that the prediction accuracy can be efficient and effective.

References

[1] Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R.L., Torre, L.A., Jemal, A.: Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. CA: a cancer journal for clinicians 68(6), 394–424(2018)

[2] Parkin, D.M., Bray, F., Ferlay, J., Pisani, P.: Global cancer statistics, 2002. CA: a cancer journal for clinicians 55(2), 74–108 (2005)

[3] Heller, N., Sathianathen, N., Kalapara, A., Walczak, E., Moore, K., Kaluzniak H., Rosenberg, J., Blake, P., Rengel, Z., Oestreich, M., et al.: The kits19 challenge

[4] data: 300 kidney tumor cases with clinical context, ct semantic segmentations, and surgical outcomes. arXiv preprint arXiv:1904.00445 (2019)

[5] Ronneberger, O., Fischer, P., Brox, T.: U-net: Convolutional networks for biomedical image segmentation. In: MICCAI. pp. 234–241. Springer (2015)

[6] Paul Yushkevich, Joseph Piven Heather Cody Hazlett, Rachel Gimpel Smith James C. Gee and Guido Gerig. User guided 3D active segmentation of anatomical structure: Significantly improved efficiency and reliability. Neuroimage. 2006 Jul 1.

[7] Ronnebrger, O., Fischer, P., & Brx, T. (2015 October). U-net: Convolutional networks for biomedical image segmentation.

[8] Li, X., Chen, H., Qi, X., Dou, Q., Fu, C.W., Heng, P.A.: H-denseunet: Hybrid densely connected unet for liver and tumor segmentation from ct volumes. IEEE TMI 37(12), 2663–2674 (2018)

[9] Isensee, F., Petersen, J., Kohl, S.A., J¨ager, P.F., Maier-Hein, K.H.: nnu-net: Breaking the spell on successful medical image segmentation. arXiv preprint arXiv:1904.08128(2019)

[10] C¸ i¸cek, O., Abdulkadir, A., Lienkamp, S.S., Brox, T., Ronneberger, O.: 3d u-net: learning dense volumetric segmentation from sparse annotation. In: International conference on medical image computing and computer-assisted intervention. pp.424–432. Springer (2016)

Author Profile

Prashant Jadiya is pursuing Bachelor of Engineering in Information Technology at Marwadi Education Foundation’s Group of Institutions (Rajkot, Gujarat, India).