An attempt at KiTS19 Challenge

Daniele Russica

December 22, 2022

Abstract

In the Kidney Tumor Segmentation challenge (KiTS19) our task was to create a model capable of segmenting kidney and kidney tumor, given a dataset of annotated CT scans. Since the U-Net is considered to be one of the best segmentation architecture in medical domain, my approach was to create 2 different U-Net models: one to segment the kidneys and the other one to segment kidney tumor. The average score calculated using annotated CT scans for testing was 0.57 for kidneys and 0.21 for tumor.

1 Introduction

The kidney tumor segmentation challenge was a competition held in 2019, where the objective was to create a model capable of correctly segmenting kidneys and kidney tumors given a CT scan. The dataset consisted in 300 high quality abdominal CT scans, 210 of which were annotated and the remaining 90 where used to evaluate contestants' models.

Considering that when it comes to segmentation architectures the U-Net[2] or some of its variants like the 3D U-Net[1] are considered to be the most successful models in the medical domain, in this project I tried to implement 2 U-Net models in order to segment separately kidneys and tumors. Unfortunately, this structure performed modestly for tumor segmentation but failed to segment correctly tumors many times.

2 Preprocessing

The provided dataset had inhomogeneus values. So it was necessary to resample both the imaging and the mask data we were provided. To optimize the trade-off between training time and information loss I decided to resample all cases to a common voxel spacing of 1x1x1 mm and to a shape of 192x256x256.

2.1 Dataset modifications

During the challenge phase of the competition, various teams pointed out faulty cases to the organizers (like with Case Ids 15 and 37)[1] so known errors in the dataset were already amended.

During preprocessing, I noted that Case Id 160 had an unusual shape of Zx512x796, considering that every other patient in the dataset had a shape of Zx512x512, but since it was only one case and its imaging were quite similar to augmented imaging from other cases I decided to keep it.

3 Network training and dataset splitting

Since Case Id 0-209 where annotated CTs I decided to split 80% of the cases (Case Id 0-167) for training and use the remaining 20% for testing. So considering that every case CT scan was sliced in 192 images, the training dataset was made of more than 30,000 images[2].

During training I applied data augmentation, using horizontal flipping, shifting, sheering, zooming and changing the brightness.

---

[1]https://github.com/neheller/kits19/issues/21
[2]https://www.kaggle.com/datasets/danielerussica/kaggle19v2

Both the models for kidney segmentation and kidney tumor segmentation were trained on Google Colab and built using Keras, with a batch size of 16 and 12 epochs. The training of a single network took about 8 hours.
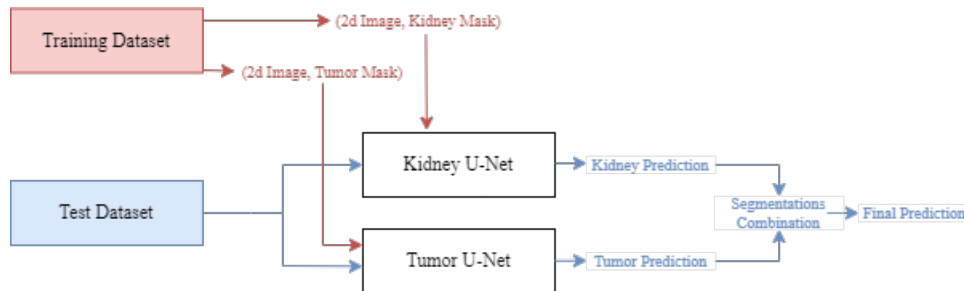

Figure 1: Schematic diagram of the U-Net model

4   Results

After the same amount of training, the kidney model performed much better compared to the tumor model. This is probably because the kidneys are bigger, easier to segment, and appeared more frequently compared to the tumors in our training dataset.

Using Case Id 168-209 as test cases, and using the evaluation function provided in the GitHub page[3] of the competition, the average score calculated for kidneys was 0.57 and the average score for tumor was 0.21.
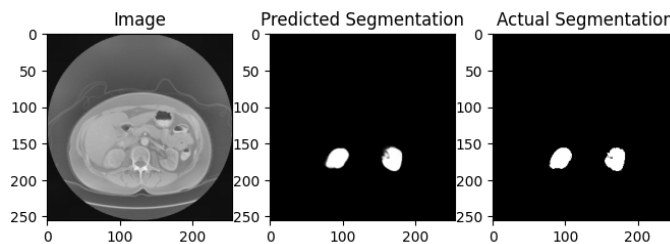

Figure 2: Visualizing the results. How different predicted kidneys are from actual ones
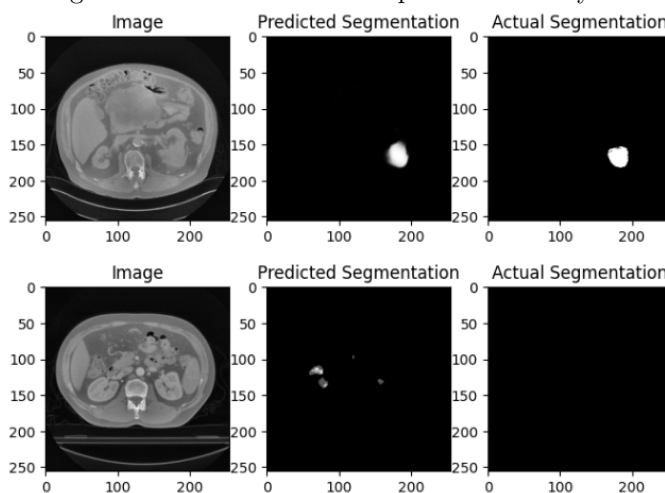

Figure 3: Visualizing the results. How different predicted tumor is from the actual one. Frequently the predictions were off by a lot.

---

[3]https://github.com/neheller/kits19/blob/master/starter_code/evaluation.py

5 Discussion

Considering that I had no experience in computer vision, I am already satisfied with the result I was able to get, every step of the process was something new for me from preprocessing to predicting the output. However, I am quite sure that those result are probably not enough and I am even more sure that there are thousands of way to improve my result.

My original idea was to develop more complex U-Net architectures, like a 3D U-Net and 2D multi-class U-Net model, after the first 'naive' method was implemented, in order to compare the results. However, due to lack of skills, knowledge and time I was not able to accomplish that.

Furthermore, when testing the model, cross validation was not used, therefore the real effectiveness of the model might be slightly different compared to the one that was calculated.

References

[1] Özgün Çiçek et al. "3D U-Net: Learning Dense Volumetric Segmentation from Sparse Annotation". In: *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016*. Ed. by Sebastien Ourselin et al. 2016, pp. 424–432. ISBN: 978-3-319-46723-8.

[2] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation". In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.