

Data augmentation techniques for the task of Lung Segmentation

Andjela Todorovic *

* Institut Polytechnique de Paris

Abstract: This report gives an overview on possible directions in working with limited size datasets for lung segmentation. Using common data enriching methods and novel GAN-based approaches, we are able to generate new synthetic data for improving model accuracy and robustness. We demonstrate the approaches on the small lung segmentation dataset, using Attention UNet as a backbone model.

Keywords: data augmentation, image segmentation, medical imaging

1. INTRODUCTION

Lung segmentation is one of the most useful tasks of machine learning applications in medical imaging. Lung CT image segmentation is an initial step necessary for lung image analysis, and a preliminary step to provide accurate lung CT image analysis such as lung cancer prediction. However, due to the characteristic of biomedical engineering domain, where complexity and the high cost of experiments restrain the number of available samples, segmentation is often performed on relatively small datasets.

On the other hand, prediction accuracy of the neural networks is largely reliant on the amount and the diversity of data available during training. Data augmentation and dataset enrichment techniques come handy in increasing the diversity of the training data, and the amount of data in the training set. In this report, we will propose couple of simple, yet efficient techniques for data augmentation before and during the training. We test results on UNet and its advanced version, Attention UNet and draw meaningful insights towards the future work.

2. THEORETICAL BACKGROUND

2.1 Attention U-Net

Oktay et al. [2018] propose an efficient modification to the UNet model that is particularly effective for medical image segmentation tasks, especially for tissue/organ localization and segmentation of small organs (such as pancreas, as proposed in the paper). The key concept introduced by the authors are *attention gates* that eliminate the necessity of applying an external object localisation model.

The attention gate takes two inputs, vectors x and g , where g is taken from the lower layers of the network (which results in better feature representation). Vectors x and g are then parsed respectively to the strided convolution and 1x1 convolution respectively, and then are summed element-wise. This process results in aligned weights becoming larger while unaligned weights become

small enough to get scaled to 0. The weights are then parsed to the sigmoid layer that scales the vectors in a $[0, 1]$ range. Attention coefficients are upsampled to the original dimensions of the x vector using trilinear interpolation and multiplied element-wise to the original x vector. This process briefly explains the idea of giving the skip connections present in the UNet a suggestion on which region to focus on while segmenting the given object.

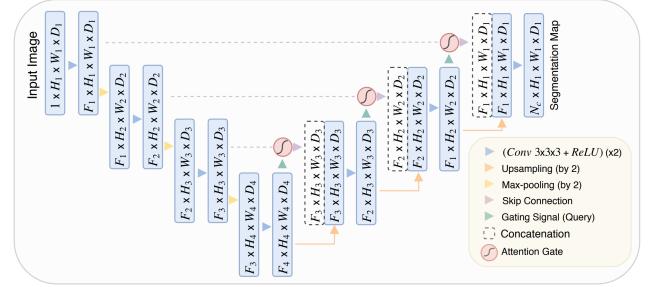


Fig. 1. Attention UNet architecture, with attention Gates displayed in red

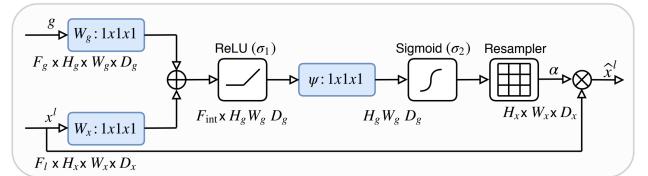


Fig. 2. A detailed view on the attention gate mechanism

3. DATA AUGMENTATION METHODS FOR PRE-TRAINING DATASET ENRICHMENT

3.1 Using generative models as an augmentation technique

GANs (Generative Adversarial Networks) have proven to be superior on all tasks related to the generation of artificial imagery. They consist of two neural networks

which work against each other: the generator and the discriminator. The generator creates fake samples with the aim of fooling the discriminator, and the discriminator learns to differentiate between real and fake samples.

[Radford et al. \[2015\]](#) have proposed and evaluated a set of constraints on the architectural topology of Convolutional GANs that make them stable to train in most settings and create photorealistic imagery. Their architecture, named *DCGAN*, replaces pooling layers in discriminator and generator by strided convolutions and fractional-strided convolutions and removes fully connected hidden layers in order to improve image generation.

DCGAN Generator takes a noise vector as the input in order to diversify the potential outputs. This vector is then reshaped into a 4-dimensional tensor and used as the start of the convolution stack. A succession of convolutional layers will reduce the depth and form a pattern in the other dimensions until we are able to obtain the output, for a colorized image. For the discriminator, the last convolution layer is flattened and then fed into a single sigmoid output.

In the experiments we conduct, we train a DCGAN on the separate parts of a dataset that all belong to the same patient (and thus have similarly looking lung masks). The back idea is to create artificial noise to the existing images while keeping the same lung mask and thus enable the model to be more robust in segmentation.

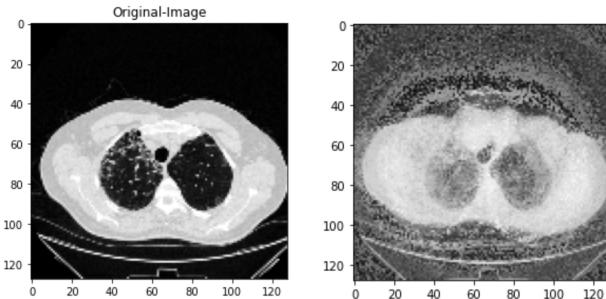


Fig. 3. Comparison of original image versus DCGAN generated image, both for the same patient. Despite the changes in the shape of the ribs, shapes of the lungs stay similar, thus ensuring that the same mask can be applied to the GAN-generated image

3.2 CT-GAN inspired data augmentation

A group of authors [Mirsky et al. \[2019\]](#) have recently constructed a framework, called *CT-GAN*, that they use for tampering medical images.

CT-GAN uses two conditional GANs (cGAN) to perform in-painting (image completion) on 3D imagery. The pipeline is as follows: first, it locates where the evidence should be inject/removed, and proceeds by cropping a random size cuboid. They would proceed by modifying only the cuboid part by cGANS, rescaling it and pasting it into the original scan.

For the sake of the experiment only, we have trained a CT-GAN on 3D voxel data generated from the LIDC-IDRI dataset. LIDC-IDRI (Lung Image Database Consortium

image collection) consists of diagnostic and lung cancer screening thoracic computed tomography (CT) scans with marked-up annotated lesions, and is one of the most commonly known lung cancer detection and diagnosis database. The injection results look promising, but this dataset has no lung segmentation masks, and the network seems not to be reproducible on the dataset that we posses. Our future work will try to implement the network that would take the (2D) image data and a set of coordinates of region of interest, and performs inpainting in a manner of CT-GAN.

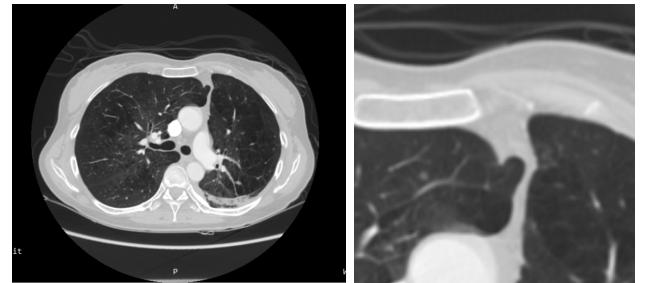


Fig. 4. One example of an image from LIDC-IDRI dataset (before voxel preprocessing): insertion is performed nearby the lung edge. Full image - left, inpainted part where injection of lung tissue (gray) has been performed - right.

4. DATA AUGMENTATION METHODS IN THE TRAINING GENERATOR

4.1 Classical augmentation

For data augmentation, making simple alterations on visual data is popular. We perform operations such as random rotation, perspective change, affine scaling and translation etc. in order to get a simple, yet reliable way to allow the model to learn by forming new and different examples to the training set. Important thing to notice is that these images are created in the training generator, which means that the new samples are generated for each training run.

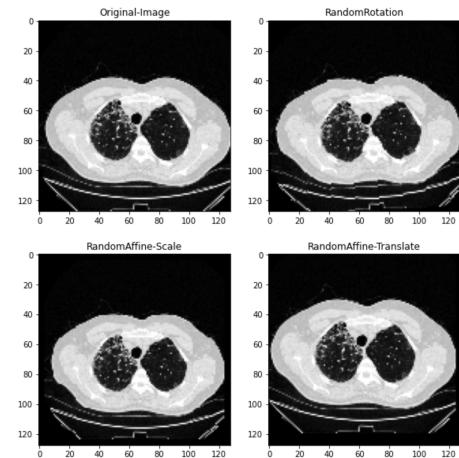


Fig. 5. Classical data augmentation techniques - displayed are only a few techniques

4.2 Polygon generation

What has shown itself as the most promising approach in the data augmentation during the training is generating arbitrary convex polygons near the very edge of a lung. Upon retrieving the edgepoints of the segmentation masks, we choose an arbitrary point on one of the lungs as a center of the polygon to be generated.

The idea of generating polygons is as follows: Given a center, we perform a walk around the circle taking a random angular step each time, and at each step put a point at a random radius. The angles of each point relative to the circle are sampled from the Uniform distribution as

$$\delta\theta_i = U\left(\frac{2\pi}{n} - \epsilon, \frac{2\pi}{n} + \epsilon\right) \quad (1)$$

$$\theta_i = \theta_{i-1} + \frac{1}{k}\delta\theta_i, k = \sum \delta\frac{\theta_i}{2\pi} \quad (2)$$

where ϵ is an *irregularity* parameter that controls whether or not the points are uniformly space angularly around the circle.

The radius is computed randomly from Normal distribution as follows:

$$r_i = \text{thresh}(N(r_{\text{average}}, \sigma), 0, 2r_{\text{average}}) \quad (3)$$

where σ_i controls how much the points can vary from the circle of average radius, and thresh thresholds a number drawn from Normal distribution in a range between 0 and $2r_{\text{average}}$.

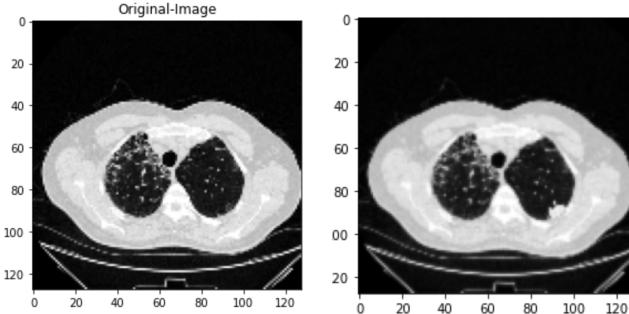


Fig. 6. Comparison of original image versus polygon generated image

5. RESULTS AND CONCLUSIONS

We will observe result obtained by training a classical UNet model, Attention UNet and Attention UNet with data augmentation. Considering that the small dataset can easily lead to the model overfitting in the first couple of epochs, training is done on 10 epochs, with batch size of 32 and learning rate of 10^{-6} with early stopping enabled. We use binary cross-entropy $-(y \log(p) + (1 - y) \log(1 - p))$ as a loss function.

Displayed are the comparisons in average validation accuracy and actual performances.

	UNet	Attention UNet	Attention UNet augm.
Loss	0.45071	0.4480	0.4780
Accuracy	88.24	65.58	68.32

Fig. 7. Loss and accuracy of the UNet, Attention UNet and Attention UNet with the augmented data

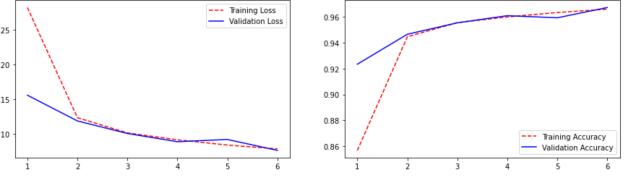


Fig. 8. Loss versus accuracy plot for the UNet model.

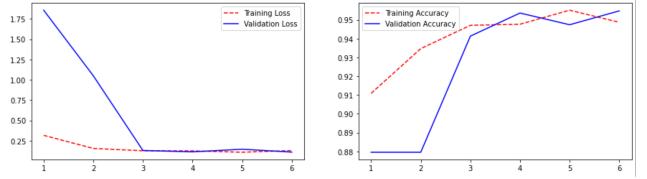


Fig. 9. Loss versus accuracy plot for the Attention UNet model.

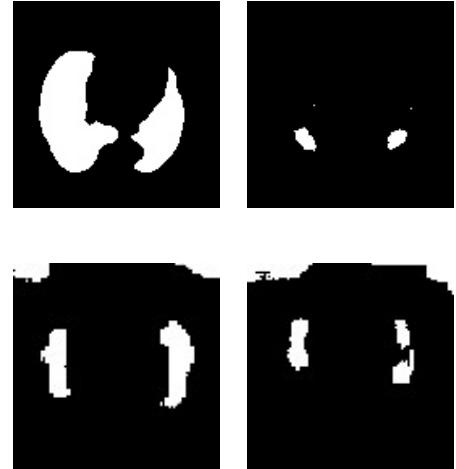


Fig. 10. Prediction on the top-left image as validation mask: UNet (top-right), Attention UNet (bottom-left), Attention UNet with data augmentation (bottom-right)

From the prediction masks, we can draw the conclusion that Attention UNet gives better prediction on the actual lungs, at the rate of giving false-positive predictions in the background, thus giving visually better predictions despite the low model accuracy. This can be also observed looking at the attention maps, displayed below.

In addition to this, our experiments show only slight improvement in terms of model accuracy with a technique of generating polygons, but this, and relevant data augmentation techniques will be further observed and discussed in the future.

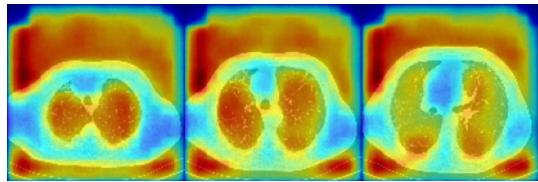


Fig. 11. Attention heatmaps clearly put emphasys on the background of the image, in addition to the actual lungs. This would require further training or better data preprocessing

REFERENCES

- Mirsky, Y., Mahler, T., Shelef, I., and Elovici, Y. (2019). Ct-gan: Malicious tampering of 3d medical imagery using deep learning.
- Oktay, O., Schlemper, J., Folgoc, L.L., Lee, M., Heinrich, M., Misawa, K., Mori, K., McDonagh, S., Hammerla, N.Y., Kainz, B., Glocker, B., and Rueckert, D. (2018). Attention u-net: Learning where to look for the pancreas.
- Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks.