
END-TO-END LUNG NODULE SEGMENTATION AND VISUALIZATION IN COMPUTED TOMOGRAPHY USING ATTENTION U-NET

A PREPRINT

Nanyan Zhu*
Biotechnology[†]
nz2305@columbia.edu

Chen Liu*
Electrical Engineering[†]
cl3760@columbia.edu

Christine P. Hendon
Associate Professor in Electrical Engineering[†]
cpf2115@columbia.edu

Andrew F. Laine
Professor in Biomedical Engineering and Radiology[†]
al418@columbia.edu

Jia Guo[‡]
Assistant Professor in Neurology[†]
jg3400@columbia.edu

August 24, 2019

ABSTRACT

One of the most important features to examine during lung cancer screening is the presence, quantity, size, location and morphology of lung nodules. To alleviate the human labor, we developed a fully automated pipeline to facilitate the detection of lung nodules in chest computed tomography scans. The pipeline consists of a lung-segmentation module that implements a unsupervised, traditional image processing algorithm and a subsequent supervised, nodule-segmentation module based on deep learning that utilizes the Attention U-Net architecture.

1 Introduction

By far, lung cancer has been the leading cause of cancer death each year for both men and women in the US, contributing to a higher annual death count than colon, breast, and prostate cancers combined. By the end of the year **2019**, there will be **228,150** new lung cancer cases and **142,670** deaths from lung cancer, according to the estimation given by the American Cancer Society [3]. To reduce the mortality rate of lung cancer, a solution proposed and demonstrated effective in recent years is computed tomography (CT) screening [11]. By interpreting the images generated from thoracic (aka., chest) CT scans, radiologists may catch some early signs of lung cancer, and eventually increase the patients' chance of survival by providing follow-up examinations and treatment plans [7] [8] [9] [10].

Lung nodules, or more formally referred to as "pulmonary nodules", are defined as "round opacity, at least moderately well margined and no greater than **3 cm** in maximum diameter" [13]. They are an important feature to which radiologists pay special attention when interpreting the thoracic CT images, as their quantity, size, location and morphology may greatly affect the lung cancer diagnosis. In current clinical practices, human experts are the only entities to view and draw inferences from the CT scan images, and this implies a time-consuming and laborious process of carefully examining beyond 100 slices of images per patient [12].

*These authors contributed equally.

[†]These authors are all affiliated to Columbia University, New York, NY.

[‡]Corresponding author.

In the aim to assist the radiologists and alleviate human labor, we developed a computer-aided diagnosis (CAD) pipeline to automate the process. The goal of the pipeline is to process any qualified thoracic CT scan, segment out the lungs and lung nodules, and provide a visualization of the scan with the volumes corresponding to background-stripped lungs and lung nodules color-coded in two distinct colors.

2 Relevant Work

The need for lung nodule segmentation has received unprecedented attention recently, and there has been many relevant work that address issues similar to what we propose. The attempts mostly come from three disciplines: traditional image processing, classic machine learning, and deep learning.

Traditional image processing approaches typically seek the most effective combination of methods to perform nodule segmentation. The most popular segmentation methods to be chosen from include global or adaptive thresholding, edge-based segmentation, filter-based segmentation such as Laplacian of Gaussian, as well as slightly more advanced ones including watershed, region-growing, and active contour (aka., snakes). Some researchers have also utilized more sophisticated, typically probabilistic techniques. For instance, Badura et al. [18] segmented various types of lung nodules with fuzzy connectedness analysis, and Gonçalves’ team [19] proposed a comprehensive Hessian-based strategy that utilizes the concepts of central medialness adaptive principle, shape index and curvedness.

With the recent advancements in Machine Learning, techniques from this field have also been applied to solve the lung nodule segmentation task. While classic machine learning models such as support vector machine (SVM) has been used to solve this problem, with Keshani’s group being an example [20], Convolutional Neural Network (CNN), an iconic breakthrough in the field of deep learning, is by far the most frequently adopted category of models in nodule segmentation as well as in most other image-related fields. In the LUNA16 grand challenge [21] and the Data Science Bowl 2017 [22], two major events focusing on lung nodule analysis and lung cancer detection, numerous participants demonstrated dozens of different CNN models and multiple techniques to improve model performances. Follow-up studies are still being conducted even years after the competitions ended. For example, Dou’s team [23] proposed a three-dimensional (3D) CNN network that extracts and combines textural features at three different magnification levels; Zhu’s team [4] utilized two consecutive 3D dual path networks to perform nodule segmentation and cancer classification respectively, and in the most recent work we found, a conditional generative adversarial network (cGAN) was used to artificially generate realistic nodule counterfeits to balance the various nodule types that were unevenly represented in the original data [17].

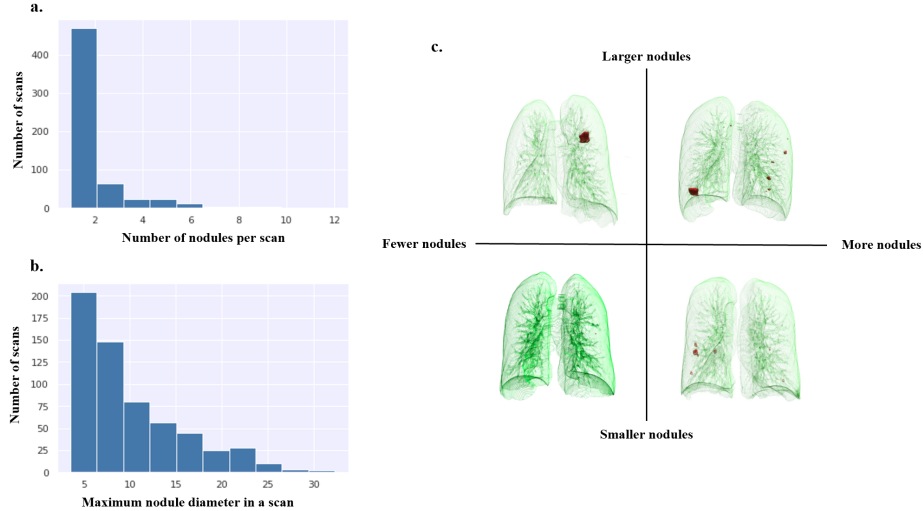


Figure 1: **Overview of nodule quantity and size in the 888-scan dataset.** **a.** Nodule quantity distribution across all scans. **b.** Nodule size distribution across all scans. **c.** Some example scans representing different positions along the quantity-size spectrum. These scans are visualized using our proposed visualization as mentioned in the Abstract.

3 Dataset

Our **3D** thoracic CT scan images came from the LUNA16 dataset [21], which is a selected subset of the Lung Image Database Consortium image collection (LIDC-IDRI) [24] such that scans with axial slice thicknesses (along the superior-inferior, aka., the head-to-foot direction) greater than **2.5 mm** are excluded. A total of **888** unique full-lung CT scans are included. The in-plane dimension of those slices within the axial plane (perpendicular to the superior-inferior direction) are **512×512** pixels, but the total number of axial slices, as well as the voxel spacing along the three Cartesian axes vary among scans. The quantity and size of lung nodules within every scan also vary from patient to patient, and such information is summarized in Figure 1.

Along with the CT scan files, LUNA16 provided a set of coarse nodule annotations using a location-with-approximate-radii (x, y, z, r) format. These annotations assumed the nodule shapes to be spherical, which were clearly sub-optimal if we would like to use them as the segmentation ground truth. Fortunately, voxel-level binary nodule labels on the parent dataset of LUNA16 were available on the LIDC-IDRI website. From there the voxel-level annotations, provided by **4** experienced radiologists according to the standards specified in Amato et al [25], were extracted and used as the segmentation ground truth after some modification.

4 Method

We divided the end-to-end lung nodule segmentation objective into two sequential tasks, lung segmentation and nodule segmentation, and designed a separate module for each task. Lung segmentation, due to the absence of ground truth annotation, was completed using unsupervised traditional image processing. The nodule segmentation, with ground truth annotation available, was tackled with a supervised deep learning network. The proposed pipeline is shown in Figure 2. Some design alternatives, such as the exact choice of nodule segmentation network architecture, are evaluated and justified. Certain design choices, specifically **1)** why we used a **2D** network instead of its **3D** counterpart, and **2)** why we provide the nodule segmentation network with cropped patches rather than entire slices, are explained in the Discussion section.

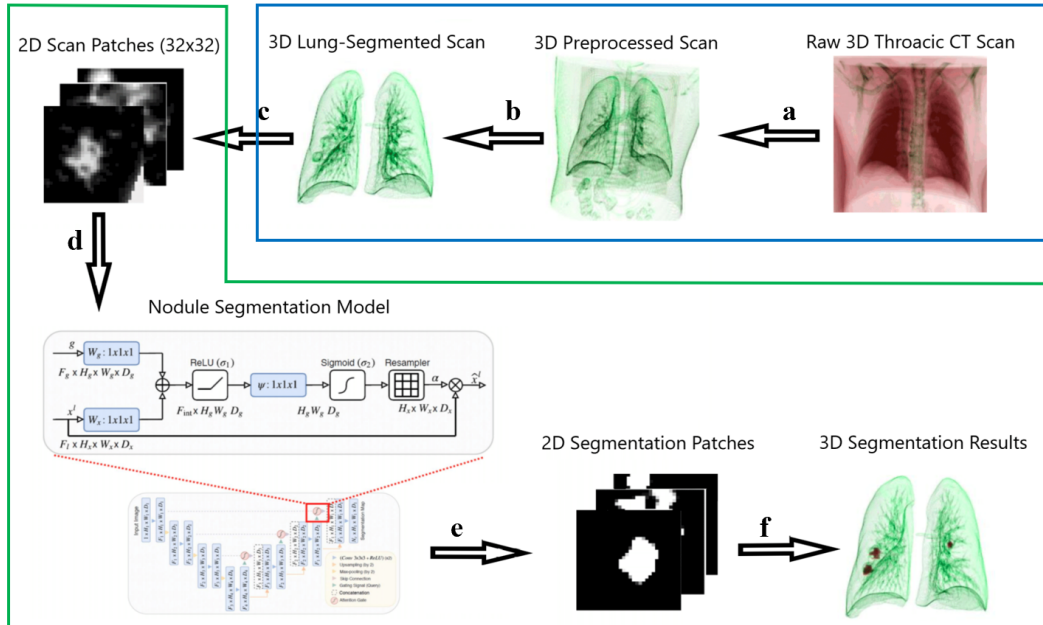


Figure 2: **Proposed end-to-end nodule segmentation pipeline with a lung segmentation module (blue) and a nodule segmentation module (green).** The operations represented as successive hollow arrows in the diagram are: **a.** preprocessing, **b.** lung segmentation, **c.** patch generation and data augmentation, **d.& e.** deep learning for nodule segmentation, and **f.** segmentation reconstruction and visualization.

4.1 Lung Segmentation Module

Our lung segmentation module performs the following procedure. Strictly speaking, **2–4** are preprocessing steps, while the lung segmentation takes place in steps **5–7**. Figure 3 demonstrates our lung segmentation procedure summarized above.

1. Take in the **3D** CT scans in MetaImage format (*.mhd and *.raw) [26] with intensity values in Hounsfield units.
2. Re-sample the **3D** data to a consistent isotropic spacing of **1×1×1 mm** among adjacent voxels.
3. Apply a global threshold at an experimentally-determined level at **-800** Hounsfield units to merge the regions outside the human body.
4. Perform an intensity transform to match the standard for lung CT images found on Radiopaedia [27] using a window center at **-600** and a window width of **1500**.
5. Segment the volumes that belong to the lungs using "Active Contour followed by Bayesian Corrections" introduced in [1].
6. Remove everything except for the two largest **3D** connected components in the scan. Also remove the second largest component if the volumes of the two largest components differ by more than **5** times or if the second largest component is smaller than **0.2** Liters. The ratio and volume thresholds were determined based on our basic knowledge about human lungs.
7. Pad all regions other than the lungs using a value of **45** which corresponds to the Hounsfield unit intensity of healthy muscle tissues, and
8. Store the **3D** preprocessed (PP) and lung-segmented (LS) scans as DICOM (*.dcm) [28] files for easy access in the following steps.

Our lung segmentation module utilized some simple preprocessing techniques, the idea of Active Contour followed by Bayesian corrections in Chung et al. [1], and lastly some post-cleaning based on our amateur knowledge about the human lung geometry. The main reason we used the "Active Contour with Bayesian Correction" approach is that Chung et al. compared their method with some other existing ones and demonstrated its great performance on segmenting the juxtapleural nodules that reside on the lung boundaries as part of the lungs rather than as tissues outside the lungs. The post-cleaning that set volume criteria on the **3D**-connected components aimed to distinguish the lungs from the isolated segmentation false positives that floated outside the lungs. Since the segmented left and right lungs were touching each other in some but not all scans, we made an educated judgement to differentiate these two cases and preserve either the two-lung continuum or the two segregated lungs as appropriate (step **6**).

4.2 Nodule Ground Truth (GT) Generation

The MATLAB toolbox shared by TesterTi [29] helped us generate the expert-annotated voxel-level binary labels where 1's stand for nodule-related and 0's stand for non-nodule in every **2D** slice where one or more radiologists believed at least one nodule exists. Then we allocated the annotated slices back to where they belong in the **3D** scans and created the ground truth for lung nodule segmentation. The same isotropic re-sampling was applied to match the LS scans. This process resulted in 4 nodule segmentation ground truth scans for every CT scan, each contributed by one radiologist.

For consistency, we further integrated these **4** sets of ground truth scans into **1** set using a majority-voting rule, such that for every voxel in a scan, its final binary value was set to **1** if it was annotated by at least **2** out of the **4** radiologists and **0** otherwise. The ground truths mentioned after this point were all referring to this set of majority-voting nodule ground truth scans we generated.

4.3 Nodule Segmentation Module

Under the help of the skeleton code provided by LeeJunHyun [30], we implemented several variants of U-Net for nodule segmentation. We split the LS scans into train, validation and test sets at a **8:1:1** ratio, and prepared the inputs to the nodule segmentation deep learning network in appropriate forms. Randomization on the scan level instead of at the slice level was ensured in the hope to prevent data leakage (i.e., the worry that deep learning model, when asked to make predictions on the test set, could simply recall the ground truth segmentation results for similar slices of the same patient from the training set).

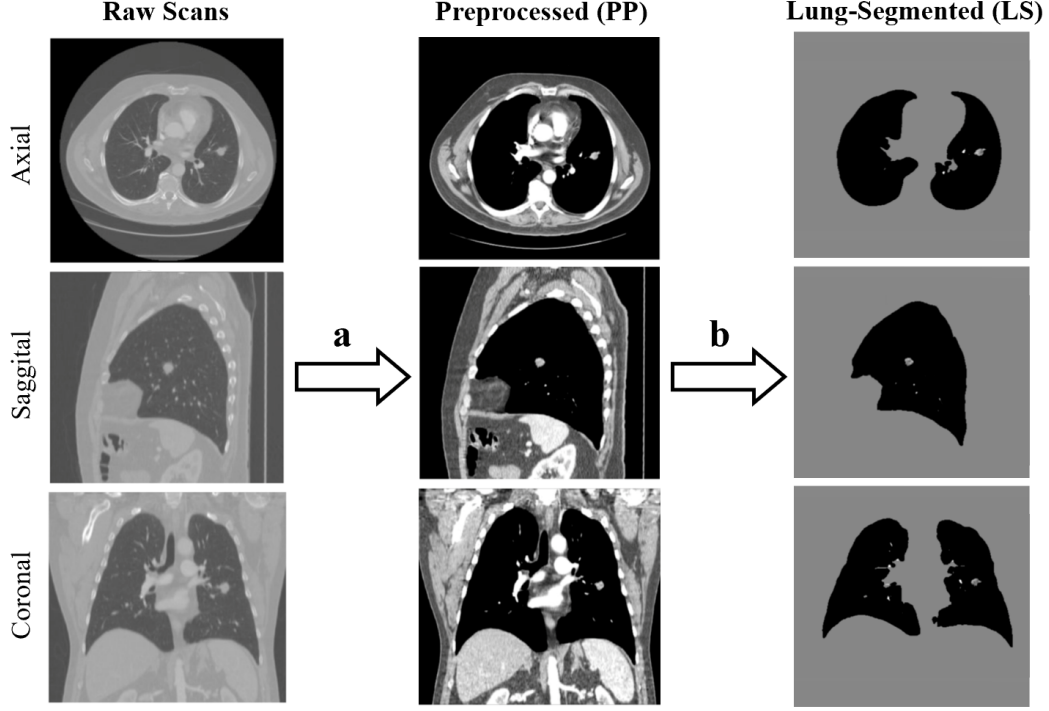


Figure 3: **Demonstration of the lung segmentation module.** One sample scan is used to visualize the effects of the lung segmentation procedure. The operations represented as successive hollow arrows in the diagram are **a.** preprocessing and **b.** lung segmentation, respectively. The resulting scans from these two operations are named as preprocessed scans (**PP**) and lung-segmented scans (**LS**). Note that the voxel values within the lungs have not changed throughout the process. The seemingly different intensities across the three scans and some blobs/fibers that seem to be missing in the PP and LS lungs are neither real: they are just display artifacts due to the fixed intensity range for image display.

4.3.1 Patch Generation and Data Augmentation

Within any scan, the total count of nodule-related voxels is minimal. The percentage of nodule-related pixels is still overwhelmingly small even when we discarded the non-nodule-containing slices and only used axial slices with nodules. Patch-based training was our solution to prevent under-representation of nodules in the training data. We extracted regions of size 32×32 around each nodule, where the nodule locations were defined by the (x, y, z, r) annotation from LUNA16.

To ensure sufficient variance of nodules were covered in the training data in the hope to prepare the deep learning model for most unseen cases, data augmentation techniques was adopted. For each nodule, five patches rather than one were generated, and each patch was randomly shifted in the x - and y - directions for any random integer value between -16 and 16 . Since the orientation and voxel spacing are fixed among these scans and such information has medical implications, we didn't consider rotation, re-scaling, or flipping as appropriate data augmentation methods. Given the nodule size distribution as shown in Figure 1b, by choosing such patch dimension and amount of shifting as mentioned, it is now possible for the fraction of nodule-related pixels in a patch to fall anywhere between **0** percent and **100** percent.

On the reverse side, we also balanced the training data with negative (i.e., absence of nodule) cases to prevent heavy over-representation of nodules. We randomly sampled **1** patch for every **4** patches generated using the method discussed above. Due to the natural low presence of nodules, these random samples are likely to cover either nodule-free areas of the lungs, or areas outside the lungs. At a very low probability they may contain nodules as well. With such patch-based dataset preparation and data augmentation techniques, we had a reason to believe we reduced the bias in the dataset, and this was also experimentally verified. The final LS patch-based dataset contains **23,647** patches in the train set, **3,737** in the validation set and **3,002** in the test set. Note that this ratio slightly deviated from **8:1:1** because different scans do not necessarily contain the same number of nodules.

Since the current patch-based model only works with image patches of 32×32 pixels, we would need a way to apply the patch-based model on 32×32 regions over all axial slices in a scan to be analyzed. The traditional way is to use a sliding window of dimension 32×32 and a step size of 1 that iterate over the entire scan, and since each voxel was assigned a prediction score $32 \times 32 = 1024$ times, the final segmentation result would be given by averaging these 1024 values and rounding that into a binary score. Instead of using this approach, we recommend a more computationally favorable approach called nodule proposal network that can be found in the Discussion section. Before that is fully implemented, we would only test the patch-based nodule segmentation on 32×32 regions rather than over entire scans.

4.3.2 Loss Function Candidates

The choice of loss function significantly impacts the segmentation performances. Even after the patch-based dataset preparation and data augmentation, the total quantity nodule-related pixels were still fewer than non-nodule pixels. Certain loss functions may exacerbate such slight imbalance of positive and negative voxel values and encourage the network to predict all- 0 's.

$$\begin{aligned} \text{i.} \quad & \text{Dice coefficient loss with smooth term} = - \frac{|Prediction| \cap |GT| + smooth}{|Prediction| + |GT| + smooth} \\ \text{ii.} \quad & \text{R\&R dice coefficient loss} = - \frac{1}{1+\lambda} \left(\frac{|Prediction| \cap |GT|}{|Prediction| + |GT|} \right) + \lambda \frac{|Prediction| \cap |GT|}{|Prediction| + |GT|} \end{aligned}$$

Figure 4: **Loss functions candidates for the nodule segmentation network.** The overbars indicate logical inversions while the absolute value signs indicate the cardinality of the sets (i.e., the number of elements in each set). They are both global loss functions such that one image yields one value. **i.** dice coefficient loss with a smooth term is a popular loss functions for binary image segmentation. **ii.** R&R dice coefficient loss is our proposed loss function to better handle the imbalance of 1 's and 0 's in binary segmentation, where λ is called the inverse ratio.

To promote the model's sensitivity to nodules, we introduced the dice coefficient loss with a smooth term (Figure 4 i) as a loss function candidate. Since the Prediction and GT are both matrices (or vectors after flattening has been applied) of the same dimension, the dice coefficient loss is similar to computing the intersection over union between the predicted image and the ground truth. The smooth term, which was intended not only to prevent division by zero but also to decrease the loss magnitude and encourage quicker convergence, was set to 1 in our cases.

While the dice coefficient loss did encourage the model to detect the nodules and generate predictions of 1 's when the ground truth contains nodules, it is still not ideal. Practically speaking, the quantitative scores given by the dice coefficient did not fully reflect our subjective judgement on the segmentation results, and from the theoretical standpoint, this loss function does not penalize the model from making arbitrary predictions on a slice where the ground truth is almost all- 0 's. Therefore we also modified this loss function so that it would put restrictions on the model to behave reasonably both for nodule-present patches and nodule-absent patches.

Our proposed loss function, called the R&R (i.e., Rosalie and Raphael, preferred names of the two student authors) dice coefficient loss, is given in Figure 4 ii. By fine-tuning the inverse ratio λ , we could balance the strength of its regulation on 1 -valued predictions and 0 -valued predictions.

4.3.3 Assessment of Network Architectures

We assessed four variants within the U-Net family, namely U-Net [5], Attention U-Net (AttU-Net) [2], Recurrent Residual U-Net (R2U-Net) [6], and Attention Recurrent Residual U-Net (AttR2U-Net), with the last being the combination of AttU-Net and R2U-Net.

To compare the performances of the 4 network architectures in a meaningful manner, we kept all of them at their default configurations as delineated in [30], with the exception that we changed the input tensor from 3 channels to 1 channel as our image patches were all in gray-scale instead of RGB-colored, resulting in a input shape of $(1, 1, 32, 32)$, corresponding to batch size of 1, single channel, and 32×32 in-plane dimension.

4.3.4 Explaining the convolutional kernels via Kernel Activation Maps

In our attempt to explore what the deep learning network was learning and to make sense of its decisions, we adopted an approach where we demonstrated the behaviors of the convolutional kernels (aka., filters) by artificially generating the input images that would maximize their responses. These images, sometimes called kernel activation maps, were generated through a process called gradient ascend, where images were initialized with random noise and their pixel-level intensity values were gradually adjusted to invoke the highest level of response on the corresponding convolutional

kernel. By observing these images we could get a sense of what patterns and textures each convolutional kernel is the most sensible to. That was not the most straightforward representation of the knowledge acquired by the model, but at least it could provide some insight.

5 Results

5.1 Performances of Different Network Architectures

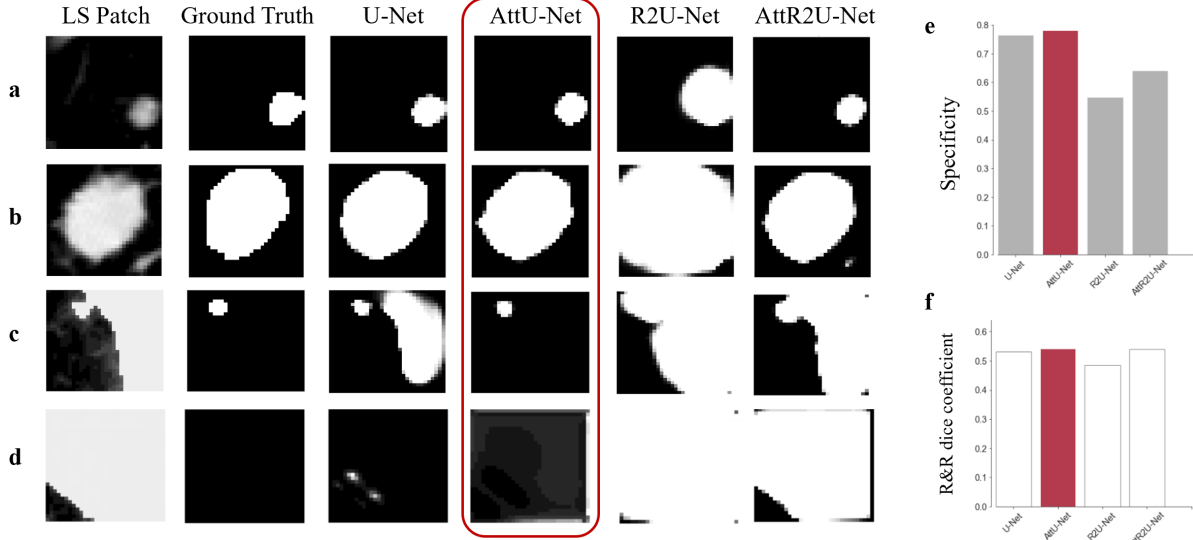


Figure 5: **Qualitative and quantitative evaluation of the four network architectures.** **a-d.** Qualitative inspection of the segmentation predictions given by the four network architectures under four representative scenarios. The best-performing network is highlighted in red. **a.** A small nodule inside a lung. **b.** A big nodule inside a lung. **c.** A small juxtapleural nodule on the edge of a lung. **d.** A region on the side of a lung without nodules. In **c** and **d**, the uniform white background is the result of constant padding in lung segmentation step 7. **e-f.** Quantitative metrics, namely the specificity and R&R dice coefficient.

The 4 candidate network architectures were trained using the regular dice coefficient with smooth term, and all other settings such as number of kernels in each convolutional layer were kept the same. Several common quantitative metrics, such as accuracy, precision, sensitivity, specificity and jaccard similarity were calculated. Dice coefficient with smooth term and R&R dice coefficient were both included as evaluation metrics as well. Based on the fact that the false positive rates are generally higher than false negative rates for all our models, specificity and R&R dice coefficient shall be the most descriptive metrics for our task, since they have the power to punish false positives in the segmentation predictions.

Segmentation examples in Figure 5 **a-d** covered a variety of scenarios and demonstrated why Attention U-Net was favored over the other three variants. It can be inferred that for our specific task, networks with the Recurrent Residual (R2) blocks tend to produce false positives while networks with the Attention (Att) blocks tend to suppress false positives. The quantitative metrics in Figure 5 **e-f** also implied that AttU-Net was the best-performing among the four variants.

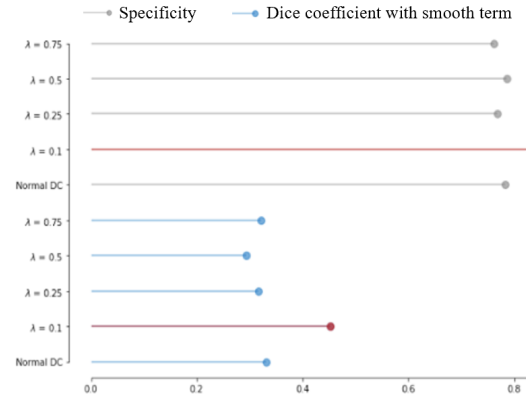


Figure 6: **Quantitative evaluation of the five loss function candidates.** Specificity and regular dice coefficient with smooth term were used for evaluation. The candidate with the highest score in each metric were highlighted in red.

5.1.1 Effects of Different Loss Functions

Based on the quantitative and qualitative analysis results, AttU-Net seemed to be the best network architecture. Further studies on loss functions were conducted exclusively on AttU-Net. Copies of the AttU-Net models were trained using the following loss function candidates: the regular dice coefficient loss with smooth term and our proposed R&R dice coefficient loss at various levels of inverse ratio λ , and their performances were recorded. Since R&R dice coefficient was optimized as it was used as the loss function, the regular dice coefficient with smooth term was instead used as one of the evaluation metrics.

Based on the quantitative metrics shown in Figure 6, we would conclude that $\lambda = 0.1$ yielded the best performance. Qualitative visual inspections, not included here to avoid overflow of figures, also confirmed this finding.

5.1.2 Selected segmentation Examples using the Best Network and Loss Function

Here we include some segmentation results of our best behaving architecture paired with the best performing loss function to showcase its performance (Figure 7). The network indeed generated sensible segmentation and in some cases the segmentation is more reasonable than the ground truth annotations given by the radiologists.

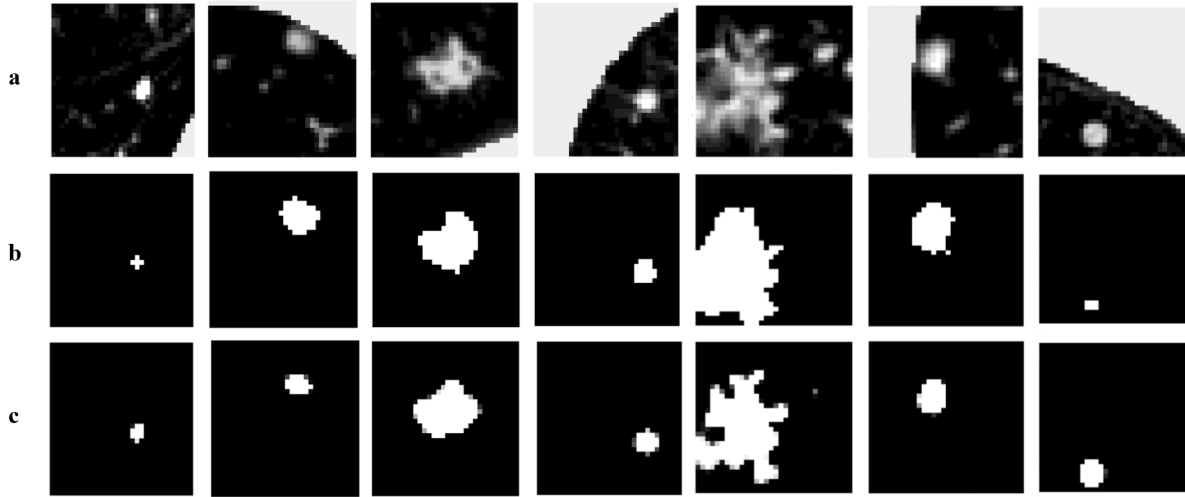


Figure 7: **Demonstration of the performance using the final nodule segmentation module.** **a.** Sample lung-segmented patches of size 32×32 pixels. **b.** Ground truth segmentation (majority-voting result of the annotations given by 4 radiologists). **c.** Segmentation predictions generated by our final nodule segmentation deep learning model (AttU-Net + R&R dice coefficient loss with inverse ratio $\lambda = 0.1$).

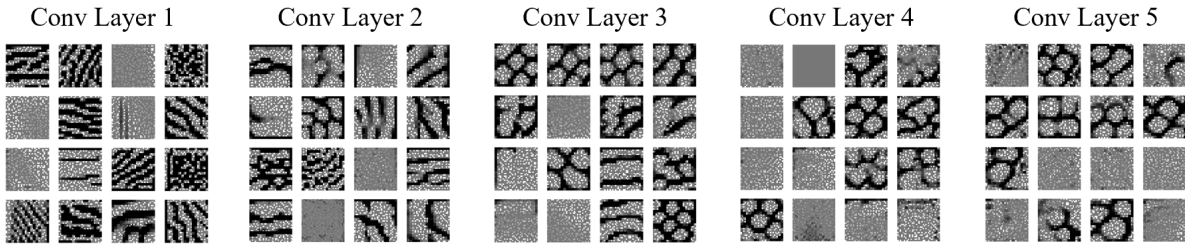


Figure 8: **Kernel activation maps in the five convolutional layers.** Sixteen (out of 64) kernels were selected at random and their kernel activation maps were computed for each of the 5 convolutional layers.

5.1.3 Visualization of the Convolutional Layers through Kernel Activation Maps

Figure 8 shows the image patterns that invoked the strongest responses to 16 randomly selected convolutional kernels in all the five convolutional blocks in our nodule segmentation deep learning model. (In fact, they are the first 16 out of 64 kernels, but since the convolutional kernels are not ordered by any means, the first 16 such kernels are mathematically

equivalent to 16 random kernels.) As we could see, the activation maps of the kernels in shallower convolutional layers possessed simpler patterns and textures, such as alternating intensities along the horizontal, vertical and diagonal directions. The activation maps deeper down the network, however, had more sophisticated patterns. For our specific application, since the deep learning network was trained to identify the lung nodules, which are mostly circular in shape, it made perfect sense that the deeper convolutional kernels were sensitive to circular patterns that resembled the size and shape of the lung nodules.

6 Discussion

We successfully implemented the end-to-end lung nodule segmentation pipeline that could produce visualizations of lung and nodule segmentation. A **3D** demonstration where the volumes corresponding to the lungs and lung nodules were stained in distinct colors can be viewed on our GitHub repository https://github.com/RosalieZhu/DL-Lung_Nodule_Segmentation. We experimentally determined that Attention U-Net yielded the best performance among the U-Net variants assessed. Finally we confirmed that our proposed R&R dice coefficient loss function did improve the segmentation performance and the optimal inverse ratio λ was **0.1** compared to other values assessed.

6.1 Rationales of design choices

6.1.1 2D versus 3D

We are aware that **3D** CNN models outperform their **2D** counterparts with other conditions being equal, because they better incorporate equally-valuable information in all three Cartesian dimensions. However, implementation of a **3D** model is likely to generate worse segmentation performances in our specific case, unless significant effort is spent in cleaning up the ground truth annotations.

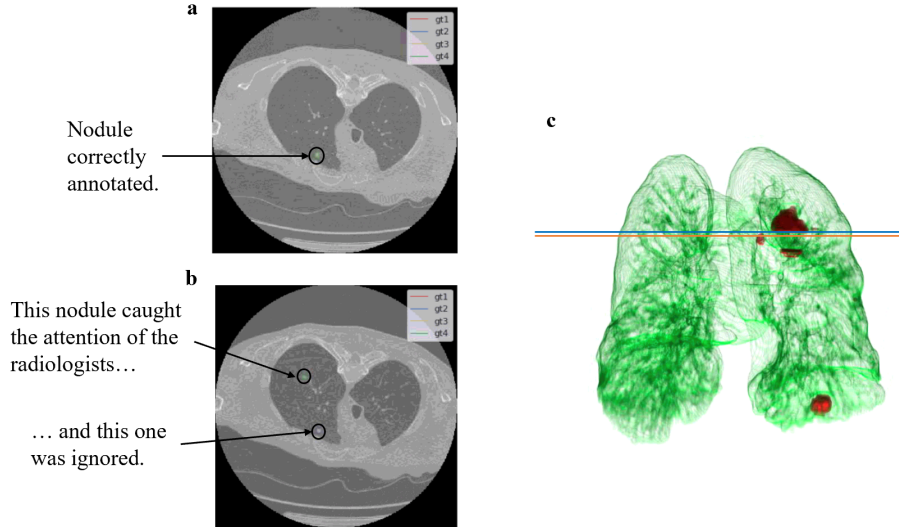


Figure 9: **Example of imperfect nodule annotation from radiologists.** **a** and **b** are two adjacent axial slices from the same CT scan. The four radiologists annotated the same and only one nodule in the superior slice, but in the inferior slice where two nodules exist, they all ignored the previously-annotated yet still existing nodule and only annotated the newly-emerged one. The **3D** rendering of the majority-voted radiologist annotation is shown in **c**, where the **blue** and **orange** lines indicate the axial slice that **a** and **b** respectively corresponds to.

Some characteristics of the radiologist annotations that we discovered, such as 1) the radiologists tend to annotate no more than **1** nodule per axial slice, 2) they do not always annotate nodules in their entirety, and 3) when there exist more than multiple nodules in the same axial slice they prefer to annotate the one that have not yet been annotated in previous slices, constituted sub-optimal annotations where multiple nodules cluster in the same scan. Figure 9 showed an great example of the problematic annotations. Under this situation, training a **3D** network with these annotation as ground truths is likely to confuse the deep learning model. Unless these annotation were corrected using sophisticated techniques, we would argue that training a **2D** network where only the validly annotated cases were included for training, validation and testing (just as what we did) was a more reasonable option.

6.1.2 Patches versus slices

We initially attempted to perform nodule segmentation on one entire axial slice at a time, but the preliminary segmentation results were poor, with the vast majority of segmentation predictions for nodule-containing slices being all 0's. We attributed this phenomenon to severe under-representation of nodules in the training data. Since the occurrences of nodules is rare by nature, with no more than a handful of them per 3D scan and they each cover a tiny fraction of any axial 2D slice, thus the ground truth data that could be seen by the deep learning model was heavily biased toward pixel values of 0's that correspond to non-nodules. Under such circumstances, predicting all pixel values to be 0's will yield a very high accuracy, and therefore the model was not motivated to correctly segmenting the nodules at all. Thanks to the patch-based training, along with data augmentation and balancing between nodule-containing and nodule-absent cases, our nodule segmentation model could eventually generate sensible segmentation results.

6.2 Areas of improvements

6.2.1 Refining the Lung segmentation

Despite being highly successful in the examples displayed by Chung et al, the lung segmentation given by the "Active Contour with Bayesian Correction" approach did not meet our expectations. In fact it only properly handled less than a quarter of the juxtapleural nodules, besides having the tendency of incorrectly segmenting tissues away from the lungs. If the lung nodule segmentation pipeline we proposed were to be improved, we would recommend replacing the lung segmentation algorithm with state-of-the-art methods in the future, or eventually find good datasets with lung segmentation ground truths and train a deep learning model for lung segmentation as well.

6.2.2 Nodule Proposal Network

In order to perform whole-scan segmentation with the patch-based model, we recommend deploying an existing deep learning network to generate potentially nodule-containing 32×32 patches for the model to analyze. The Faster R-CNN model mentioned in Zhu's team [4] was designed for almost exactly this purpose, with the only distinction that the patch proposals generated are 3D blocks of dimension $32 \times 32 \times 32$. This dimensional conflict can easily be accommodated by simply applying the nodule segmentation network over each slice of the volume for 32 iterations.

7 Acknowledgement

This study was performed in the Small Animal Imaging Lab (SAIL) at Columbia's Zuckerman Mind Brain Behavior Institute.

References

- [1] Chung, Heewon, Hoon Ko, Se Jeong Jeon, Kwon-Ha Yoon, and Jinseok Lee. 2018. "Automatic Lung Segmentation With Juxta-Pleural Nodule Identification Using Active Contour Model and Bayesian Approach." *IEEE Journal of Translational Engineering in Health and Medicine* 6 (May): 1800513.
- [2] O. Oktay et al. (2018). Attention U-Net: Learning Where to Look for the Pancreas. [online] arXiv.org. Available at: <https://arxiv.org/abs/1804.03999v3> [Accessed 9 May 2019].
- [3] "Key Statistics for Lung Cancer." [Online]. Available: <https://www.cancer.org/cancer/non-small-cell-lung-cancer/about/key-statistics.html>. [Accessed: 07-May-2019].
- [4] W. Zhu, C. Liu, W. Fan, and X. Xie. 2018. "DeepLung: Deep 3D Dual Path Nets for Automated Pulmonary Nodule Detection and Classification." 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). <https://doi.org/10.1109/wacv.2018.00079>.
- [5] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," *Lecture Notes in Computer Science*. pp. 234–241, 2015.
- [6] M. Z. Alom, C. Yakopcic, M. Hasan, T. M. Taha, and V. K. Asari. 2019. "Recurrent Residual U-Net for Medical Image Segmentation." *Journal of Medical Imaging (Bellingham, Wash.)* 6 (1): 014006.
- [7] M. Kaneko et al., "Peripheral lung cancer: screening and detection with low-dose spiral CT versus radiography," *Radiology*, vol. 201, no. 3, pp. 798–802, Dec. 1996.
- [8] P. B. Bach et al., "Benefits and harms of CT screening for lung cancer: a systematic review," *JAMA*, vol. 307, no. 22, pp. 2418–2429, Jun. 2012.

- [9] Z. Saghir et al., "CT screening for lung cancer brings forward early disease. The randomised Danish Lung Cancer Screening Trial: status after five annual screening rounds with low-dose CT," *Thorax*, vol. 67, no. 4, pp. 296–301, Apr. 2012.
- [10] T. B. Lanni et al., "Early Results From the Implementation of a Lung Cancer Screening Program," *American Journal of Clinical Oncology*, p. 1, 2015.
- [11] T. N. L. S. T. R. Team and The National Lung Screening Trial Research Team, "Reduced Lung-Cancer Mortality with Low-Dose Computed Tomographic Screening," *New England Journal of Medicine*, vol. 365, no. 5, pp. 395–409, 2011.
- [12] D. E. Wood et al., "Lung Cancer Screening, Version 3.2018, NCCN Clinical Practice Guidelines in Oncology," *J. Natl. Compr. Canc. Netw.*, vol. 16, no. 4, pp. 412–441, Apr. 2018.
- [13] J. H. Austin et al. "Glossary of terms for CT of the lungs: recommendations of the Nomenclature Committee of the Fleischner Society." In: *Radiology* 200.2 (1996), pp. 327–331.
- [14] V. S. Nair, V. Sundaram, M. Desai, and M. K. Gould, "Accuracy of Models to Identify Lung Nodule Cancer Risk in the National Lung Screening Trial," *Am. J. Respir. Crit. Care Med.*, vol. 197, no. 9, pp. 1220–1223, May 2018.
- [15] I. Vlahos, K. Stefanidis, S. Sheard, A. Nair, C. Sayer, and J. Moser, "Lung cancer screening: nodule identification and characterization," *Transl Lung Cancer Res*, vol. 7, no. 3, pp. 288–303, Jun. 2018.
- [16] J. Kalpathy-Cramer et al., "A Comparison of Lung Nodule Segmentation Algorithms: Methods and Results from a Multi-institutional Study," *J. Digit. Imaging*, vol. 29, no. 4, pp. 476–487, Aug. 2016.
- [17] Y. Qin, H. Zheng, X. Huang, J. Yang, and Y.-M. Zhu, "Pulmonary nodule segmentation with CT sample synthesis using adversarial networks," *Med. Phys.*, vol. 46, no. 3, pp. 1218–1229, Mar. 2019.
- [18] P. Badura and E. Pietka, "Soft computing approach to 3D lung nodule segmentation in CT," *Comput. Biol. Med.*, vol. 53, pp. 230–243, Oct. 2014.
- [19] L. Gonçalves, J. Novo, and A. Campilho, "Hessian based approaches for 3D lung nodule segmentation," *Expert Systems with Applications*, vol. 61, pp. 1–15, 2016.
- [20] M. Keshani, Z. Azimifar, F. Tajeripour, and R. Boostani, "Lung nodule segmentation and recognition using SVM classifier and active contour modeling: a complete intelligent system," *Comput. Biol. Med.*, vol. 43, no. 4, pp. 287–300, May 2013.
- [21] "LUNA16-Data." [Online]. Available: <https://luna16.grand-challenge.org/Data/>. [Accessed: 23-Mar-2019].
- [22] "Data Science Bowl 2017." [Online]. Available: <https://kaggle.com/c/data-science-bowl-2017>. [Accessed: 08-May-2019].
- [23] Q. Dou, H. Chen, L. Yu, J. Qin, and P.-A. Heng, "Multilevel Contextual 3-D CNNs for False Positive Reduction in Pulmonary Nodule Detection," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 7, pp. 1558–1567, Jul. 2017.
- [24] "LIDC-IDRI - The Cancer Imaging Archive (TCIA) Public Access - Cancer Imaging Archive Wiki." [Online]. Available: <https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI>. [Accessed: 23-Mar-2019].
- [25] S. G. Armato 3rd et al., "The Lung Image Database Consortium (LIDC) and Image Database Resource Initiative (IDRI): a completed reference database of lung nodules on CT scans," *Med. Phys.*, vol. 38, no. 2, pp. 915–931, Feb. 2011.
- [26] "ITK/MetaIO/Documentation - KitwarePublic." [Online]. Available: <https://itk.org/Wiki/ITK/MetaIO/Documentation>. [Accessed: 08-May-2019].
- [27] A. Murphy, "Windowing (CT) | Radiology Reference Article | Radiopaedia.org," *Radiopaedia*. [Online]. Available: <https://radiopaedia.org/articles/windowing-ct?lang=us>. [Accessed: 09-May-2019].
- [28] Softneta, "DICOM Library - About DICOM format." [Online]. Available: <https://www.dicomlibrary.com/dicom/>. [Accessed: 08-May-2019].
- [29] TesterTi. [Online]. "TesterTi/LIDCToolbox." GitHub. <https://github.com/TesterTi/LIDCToolbox>. [Accessed April 22, 2019]
- [30] LeeJunHyun. [Online]. "LeeJunHyun/Image_Segmentation." GitHub. https://github.com/LeeJunHyun/Image_Segmentation. [Accessed April 22, 2019.]