Master thesis on Computational Biomedical Engineering

Universitat Pompeu Fabra

lucanode: automatic lung cancer nodule detection

Octavi Font Sola

Supervisor: Dr. Mario Ceresa

Co-Supervisor: Dr. Gemma Piella Fenoy

June 2018

Contents

Abstract

Acknowledgments

1 Introduction **1**
1.1 Clinical Context . . . 1
1.1.1 Lung cancer . . . 1
1.1.2 Computed Tomography . . . 2
1.1.3 Lung cancer screening with CT . . . 3
1.1.4 CAD in lung nodule imaging . . . 5
1.2 Objectives . . . 6
1.3 State of the Art . . . 6
1.3.1 Candidate detection . . . 6
1.3.2 Complete nodule detection . . . 7
1.3.3 False positive reduction . . . 8

2 Methods **9**
2.1 Lung segmentation . . . 11
2.2 Nodule detection . . . 11
2.3 False positive reduction . . . 13
2.3.1 Handpicked feature classifier . . . 13
2.3.2 Radiomics-based classifier . . . 16
2.3.3 ResNet based classifier . . . 17
2.4 LUNA performance comparative . . . 19

3 Results **21**
3.1 Lung segmentation . . . 21
3.2 Nodule detection . . . 23
3.3 False positive reduction . . . 26
3.4 LUNA performance comparative . . . 29
3.5 Integration into a clinical workflow . . . 31

4 Conclusions **33**
4.1 Summary . . . 33
4.2 Future work . . . 34
4.3 General discussion . . . 34

Bibliography **41**

Abstract

Lung cancer is both the deadliest and one of the most frequently diagnosed forms of the disease. The main cause for the low survivability of lung cancer is due to its lack of early symptoms, which are only detected in terminal stages of the disease. It has been demonstrated that performing screenings in high-risk population increased the survivability by 20% but detecting lung lesions in CT imaging is time consuming and highly dependent on the skill of the radiologist.

This Master's final thesis details the implementation of a computer-aided detection (CADe) system for lung cancer nodule detection (hence *lucanode*). Its aim is to provide assistance to radiologists for early diagnosis of lung cancer by detecting round abnormalities in the lung (nodules). The thesis contextualizes the impact that such a system could have in the prognosis for this disease, analyzes the current state of the art and then dwells on the implementation of the system.

lucanode is divided into a 4 step pipeline: scan preprocessing, lung segmentation, nodule segmentation and false positive reduction. For each step, there are multiple attempted approaches, which have been quantified and evaluated against one another. Finally, the system as a whole is compared against the state of the art following the approach established in the LUNA grand challenge, a public dataset of CT images aimed at detecting lung nodules.

We discuss on the possible improvements for each step of the pipeline, as well as the required steps to integrate it into a clinical workflow. We conclude the thesis by pointing to future lines of research.

Keywords: lung cancer; lung nodules; image segmentation; image recognition; deep learning; CADe

Acknowledgments

I'd like to address a special thank you to Mario for his patience, support and dedication throughout the academic year. Also, big thumbs up for our lung club. Thanks to Miguel Ángel, Gemma, Jordina and especially to Gabriel and Xavi. Our biweekly meeting have been a great source of insights and also a bit of a group therapy. Which one I needed more of the two I shan't tell.

Chapter 1

Introduction

1.1 Clinical Context

1.1.1 Lung cancer

Lung cancer is both the deadliest and one of the most frequently diagnosed forms of the disease. For this 2018, the estimates predict that there will be 234,030 new cases and 154,050 deaths in the US alone (see Table 1.1). These trends are also shared worldwide, where lung cancer is the leading form of cancer, having caused 1.69 million deaths in 2015 [1].

Table 1.1: Ten leading cancer types for the estimated deaths, United States, 2018. Figures from [2]

Cancer type	Deaths
Lung & bronchus	154,050
Colon & rectum	50,630
Pancreas	44,330
Breast	40,920
Liver & intrahepatic bile duct	30,200
Prostate	29,430
Leukemia	24,370
Non-Hodgkin lymphoma	19,910
Ovary	14,070
Esophagus	12,850

Cigarette smoking is by far the biggest risk factor leading to lung cancer. Over 80% of the cases are still caused by it, although this rate is decreasing yearly (3.8% in men and 2.3% in women), due to the changing trends in smoking uptake. Other risk factors leading to lung cancer include the exposure to radon gas, secondhand smoke, asbestos, certain metals (chromium, cadmium, arsenic), radiation, air pollution and diesel exhaust. Genetics also play a role in the

development of the disease, especially for those who present symptoms at a younger age.

Lung cancer treatment will vary based on its type and stage. Early stage non-small cell lung cancer is usually treated with surgery, sometimes aided by chemotherapy and radiotherapy. The advanced stage of non-small cell lung cancer is usually treated with chemotherapy, targeted drugs, and immunotherapy. Small cell lung cancer is most often treated with chemotherapy combined with radiotherapy.

The 5-year survival rate for lung cancer is only 18%, the main reason being its often late diagnosis. Only 16% of lung cancer is diagnosed still at a localized stage and, in those cases, the 5-year survival rate improves to 56%. The late diagnoses can be explained due to the lack of symptoms until cancer has advanced. These symptoms may include a persistent cough, bloodstained sputum, shortness of breath and recurrent pneumonia or bronchitis.

1.1.2 Computed Tomography

Computed tomography (CT) is one of the most popular imaging methods in radiological practice. CT is based on X-rays, an electromagnetic wave discovered by Wilhelm Conrad Röntgen in 1895 [3]. X-rays have a penetrating ability that makes them useful to see through tissues, making it possible to take detailed photographs of our insides without the need for surgical intervention. In their most basic form, to capture an X-ray you need an X-ray tube, which will produce the wave, and a radiation detector. Each kind of tissue absorbs a different amount of radiation, so the differences between the emitted signal and the one captured by the radiation allow us to reconstruct the image. CT scans are measured on the Hounsfield scale, which quantifies the attenuation of each voxel based on the radiodensity of the tissue it has penetrated. See Table 1.2 for a correlation between captured radiation and tissue type.

Table 1.2: Houndsfield Unit range for different body tissues and fluids.

Substance	HU
Air	-1000
Fat	-120 to -90
Soft Tissue, Contrast	+100 to +300
Water	0
Blood	+13 to +50
Lung parenchyma	-700 to -600
Muscle	+35 to +55
Cancellouus bone	+700
Cortical bone	+3000

In a CT scan, X-ray tubes and detectors are placed in a rotating arc (see Figure 1.1). Meanwhile, the patient is slowly moved along this arc, which enables the scan to take a 360 spiral view of the subject. This is then transformed into a

volumetric view, formed of multiple 2D slices. Nowadays the best CT scanners have up to 320 rows of detectors in their arcs, which allow a thoracic CT scan to have sub-millimeter axial spacing in a matter of seconds, well within the time most people can hold their breath.

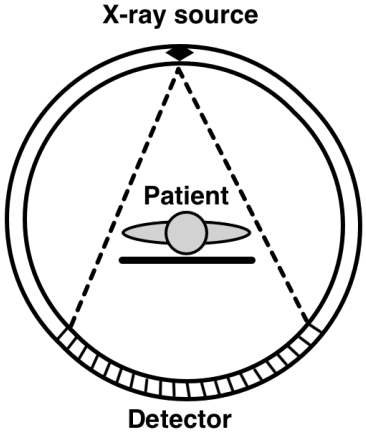

Figure 1.1: Axial view of a CT detector arc

1.1.3 Lung cancer screening with CT

Lung cancer has a low 5-year survival rate, both due to the aggressiveness of the disease and the fact that most patients are diagnosed once the cancer is past its localized stage. A mass screening protocol of high-risk groups could be potentially beneficial, but the use of chest radiography has not shown a reduction in mortality [4]. Other approaches, such as detecting molecular markers in blood or sputum have been studied, but they are currently unsuitable for clinical use [4]. Several studies have shown that the use of low dose CT (LDCT) was suitable for detecting lung nodules, which are round abnormalities that appear in the lung in the first stages of the tumor. This prompted the National Cancer Institute to fund a large-scale lung screening trial: the National Lung Screening Trial (NLST).

The NLST enrolled participants with a high-risk profile of developing lung cancer. Eligible participants were between 55 and 74 years old and had a history of cigarette smoking of at least 30 pack-years. A total of 53,454 persons were enrolled; 26,722 were randomly assigned to screening with low-dose CT and 26,732 to screening with chest radiography. The study [5] demonstrated a 20% reduction in mortality when using LDCT for screening. These results have been also replicated in other large-scale trials, such as NELSON [6].

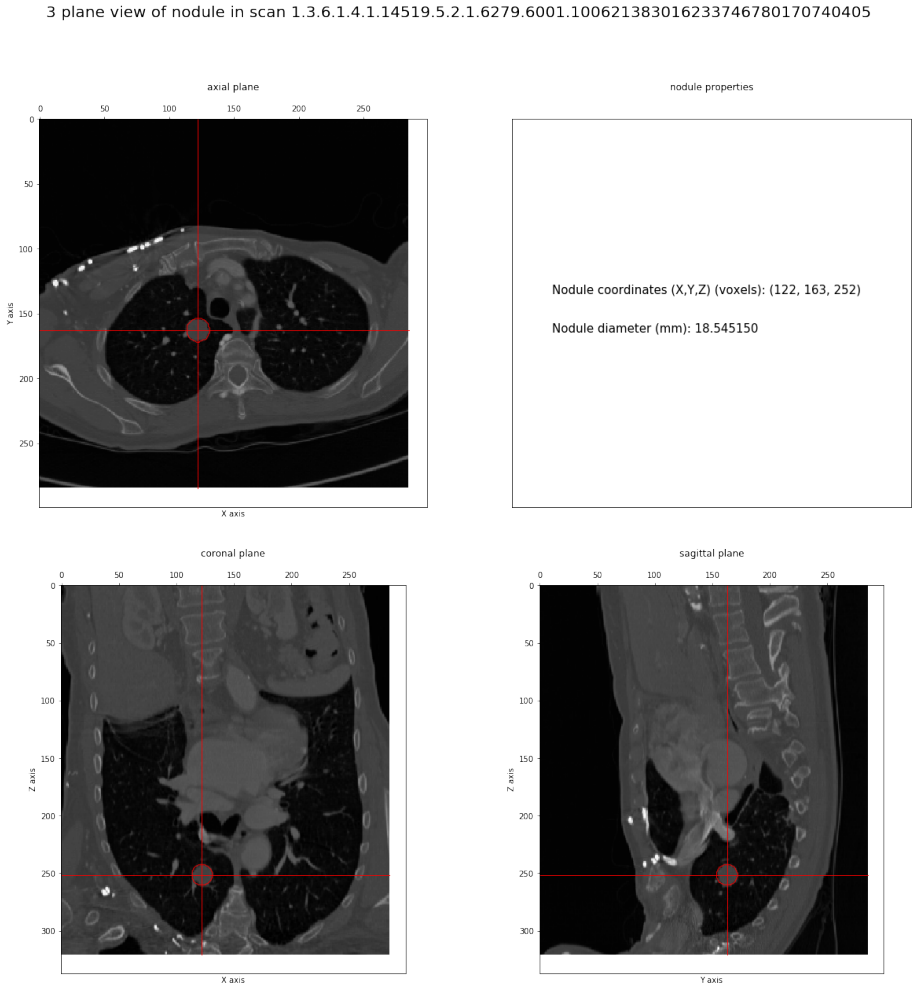

Figure 1.2: A CT scan showing a lung nodule in a 3 plane view.

1.1.4 CAD in lung nodule imaging

The introduction of systematic screening programmes in the high-risk population of developing lung cancer will increase the workload of radiologists, which is already high due to the increasing demand for image-assisted diagnoses. Moreover, screening CT scans for lung nodules benefit from thinner sections, which can result in scans with over 500 slices. To complicate the task further, air and blood vessels can be easily mistaken for nodules if only looking at the individual slice. Thus, searching for nodules requires going back and forth across the same area to check for the sphericity of the possible nodule.

There is also a lot of debate around the definition itself of *lung nodule*, which increases the variation in performance between radiologists. This, together with the increase in workload, hampers obtaining a good sensitivity rate. And it is because of all these problems we have just exposed, that an automated CAD system could help decrease the workload and increase both the sensitivity and accuracy of such a system [7].

Double reading (one pair of observers read the scan independently) has been reported to reduce errors and increase diagnostic sensitivity [8, 9], but it is time-consuming and expensive. The goal is to have a CAD system with the potential to perform this double reading in an efficient manner, which would both improve the outcomes and lower the costs of running such a protocol. Still, the initial CAD systems developed for such tasks, released circa the 1980s, were too limited resource-wise to prove useful to physicians, as they returned too many false positives. A CAD system could be helpful to detect, quantify and track the evolution of nodule candidates over time, thus deferring to the physicians the task of interpreting the results.

Until recently, there was not a systematic way to compare CAD systems between one another, so they were difficult to test and compare. As part of the LUNA grand challenge effort, and other challenges such as the Kaggle Data Science Bowl 2017 or ISBI 2018, the situation is progressively getting better. Specifically, in this thesis, we will focus on the LUNA dataset, which provides scans and nodule annotations, created by a team of 4 radiologists. The dataset also provides a set of rules regarding its evaluation, so that all participants in the challenge can test their approaches sharing a same common ground.

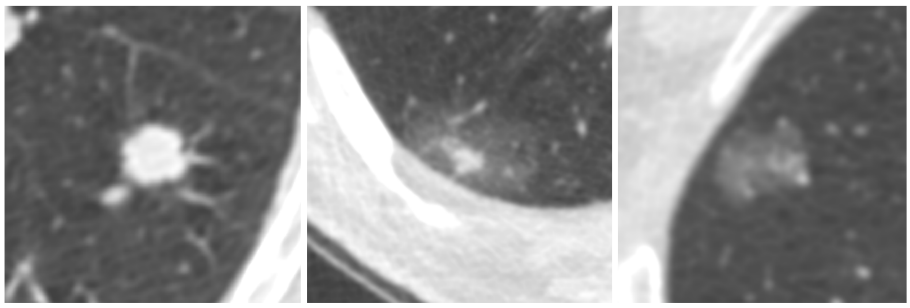

Figure 1.3: Different types of lung nodules. *left*: solid nodule, *middle*: part-solid nodule, *right*: non-solid nodule. Image from [10]

1.2 Objectives

This thesis has two objectives:

1. Developing a CAD system to detect lung nodules from thoracic CT scans competitive with the current state of the art.
2. Developing an integrated pipeline capable of generating lung nodule predictions from an arbitrary CT scan, without manual intervention.

For the first objective, we can use the LUNA dataset, which has 888 thoracic CT scans with 1185 nodule annotations. LUNA is a subset of the LIDC-IDRI dataset, with a curated list of scans and nodules. We can also use its evaluation method to test our system's performance against the state of the art.

The development of the CAD system will be split into two main tasks: nodule detection and false positive reduction. Each one will be assessed individually. In the end, the best-developed methods will be tested in conjunction.

For the second objective, we will have to automate the preprocessing pipeline and we will also have to simplify the deployment of the software. The first task will require to develop a method for automated lung segmentation. We will also need an automated step to normalize the spacing of the input scan. The second task will require the definition of a clear API on inputs and expected outputs. We will also work on the software packaging to make it easily deployable.

1.3 State of the Art

This survey on the state of the art in CAD systems has been performed over the competitors in the LUNA challenge [11]. I've divided the survey into three subsections, based on whether the systems limited themselves to the nodule detection or false positive tracks or they encompassed both problems. I have skipped over commercial solutions that haven't released a detailed version of their algorithms due to confidentiality agreements.

1.3.1 Candidate detection

1.3.1.1 ISICAD [12]

It downsamples the input image from 512x512 to 256x256 and then calculates a shape index and a curvedness feature. These values are thresholded. The resulting seeds returned from the thresholding are expanded repeating the thresholding step, with a lower value. Neighbour masked voxels are merged together to reduce the number of clusters.

1.3.1.2 SubsolidCAD [13]

An algorithm focused on the detection of subsolid nodules, which are less frequent, but more likely to be malignant. The method consists of applying thresholding

between -750 HU and -300 HU, then a morphological opening on the resulting mask to remove partial volume effects caused by the boundaries of the lungs and vessels. The mask is then labeled with connected components, discarding components smaller than $34mm^3$ in volume.

1.3.1.3 LargeCAD [14]

An algorithm focused on detecting large nodules. Large solid nodules have a different texture and shape index values that are not captured properly by the previous two methods.

1.3.2 Complete nodule detection

1.3.2.1 ZNET [15]

ZNET uses convolutional networks both for candidate detection and false positive reduction. In the candidate detection step, it trains a U-Net [16] segmentation network. Scans are resampled to $0.5x0.5x0.5mm$ and then the segmentation network is evaluated over each axial slice of the network. The segmentation mask is obtained by applying thresholding over the UNET probability map, after that, we apply a morphological erosion to remove partial volumes. The candidates are obtained by performing connected components on the resulting volume, using the centroid of each label as the candidate.

For the false positive reduction, ZNET employs a wide residual network trained on patches of 64x64 on the axial, sagittal and coronal axis. To prevent overfitting, ZNET mislabels a small percentage of the batches.

1.3.2.2 JianPeiCAD [11]

It uses a multi-scale rule-based screening to obtain the nodule candidates. False positives are detected by applying a 3D convolutional neural network with augmentation to prevent overfitting.

1.3.2.3 ETROCAD [17]

It applies isotropic resampling of 1x1x1 mm and three different set of filters on the slice. A thresholding is applied to the filtered image at different scales, to account for multiple nodule sizes. Cluster merging is performed at the end to ensure there is a single candidate per correspondent nodule.

On the false positive reduction, a set of features is computed per candidate, including shape and regional features. These features are then used to train a Support Vector Machine classifier with a radial basis function.

1.3.3 False positive reduction

1.3.3.1 CUMedVis [18]

CUMedVis uses an ensemble of 3 different ResNet [19] architectures at different scales to capture the variability between node sizes and shapes. The network preprocesses the inputs by clipping at certain voxel intensities and also uses augmentation on the axial plane to diminish class imbalance.

1.3.3.2 JackFPR [11]

A similar approach to CUMedVis. Instead of performing a linear ensemble, it uses another fully connected layer with 128 units, followed by a softmax layer to perform the final prediction.

1.3.3.3 DIAG CONVNET [20]

It uses multi-view Convnets. Instead of using a fully 3D network, it extracts patches on the 3 planes of the nodule candidate and trains an individual network for each of them. The results of each model are combined in an ensemble.

Chapter 2

Methods

Our pipeline includes three different stages (see Figure 2.1):

- **preprocessing:** To transform an arbitrary CT scan to a unified spacing.
- **nodule candidate detection:** To detect centroids of possible nodules.
- **false positive reduction:** To evaluate annotations from the previous step and to classify them based on a user-configurable threshold.

Since the search space in a CT scan can be very large, and lung nodules are, by definition, within the lung, the segmentation of the lungs is often performed prior to the nodule candidate detection. This reduces the complexity of the problem and avoid implausible false positives (outside of the lung).

We will train 3 models. One to segment lungs, another to segment nodules and finally an image detection module to reduce the error rate. For the segmentation tasks, based on our state of the art appraisal, we will choose a network architecture based on U-Net [16], whereas our image recognition architecture will be based on a 3D ResNet [21, 19].

Once the three models have been trained and evaluated, we will perform an overall evaluation of the system against the LUNA scoreboard, so we will be able to rank ourselves against the current state of the art.

Finally, since we want to use those models on other datasets, we will create a Docker image containing both the code and the network weights. This will serve as an abstraction layer of all the complexities related to dependencies and setup to run such a system. The end user will only need to deal with a simple interface, that takes a CT scan in and returns a list of candidates, with an associated probability.

The next sections describe the training process in detail for each of the models in the pipeline, as well as how the system has been evaluated against the LUNA scoreboard.

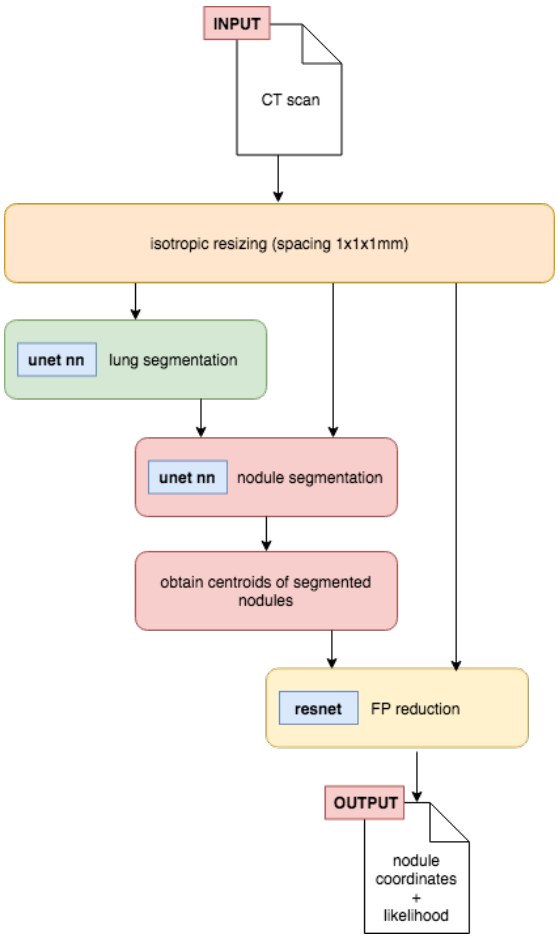

Figure 2.1: the lucanode pipeline

2.1 Lung segmentation

To perform the automated lung segmentation of a CT scan we use the same deep learning U-Net architecture (see Figure 2.2) as in the nodule segmentation. The U-Net uses a batch normalization and ReLU prior to any convolutional layer, as suggested in [21]. The training is performed over 40 epochs. Subsets 0 to 7 of the LUNA dataset are used for training, subset 8 for validation and subset 9 for testing. The learning rate is set to $1e^{-3}$, with Adam[22] as our optimization algorithm, which will adaptatively adjust the learning rate. The batch size is 5 and the weights are randomly initialized. Hardware wise, we employed an Intel i7 7700, 32 GB of RAM and a Nvidia 1080Ti GPU. The network is implemented with Keras[23], using Tensorflow[24] as its backend. The inverse of the Dice coefficient is used as loss function, where X is an array containing the original mask voxels and Y one containing the predicted mask voxels:

$$loss = 1 - DSC = 1 - \frac{2|X \cap Y|}{|X| + |Y|}$$

As part of our preprocessing, the scans will be isotropically resampled to a voxel size of 1x1x1 mm. To train this network, there is no image augmentation process nor any image filtering applied beforehand. The input passed to the network consists of an axial slice of a CT scan sized 400x400. The ground truth is the corresponding binary mask slice. They have also been automatically segmented using [25], which combines both a fast pass applying thresholding and then a second finer phase that corrects erroneous areas by applying state of the art methods. The masks are also provided as part of the challenge.

To evaluate the results of our resulting network, we will calculate the Dice coefficient of each slice and then average it across the whole scan.

2.2 Nodule detection

The basic setup for the nodule detection is the same segmentation network used on the lung segmentation. That is, a U-Net with batch normalization and ReLU, trained over 40 epochs. The dataset is split in the same way: subsets 0 to 7 for training, subset 8 for validation and subset 9 for testing. We use also a learning rate of $1e^{-3}$, Adam, and the same hardware (Intel i7 7700, 32GB of RAM and a Nvidia 1080Ti GPU). However, the preprocessing steps to perform nodule detection differ greatly from those for lung segmentation.

The LUNA dataset only has annotations of the centroid and diameter of the annotated nodules, so we manually created spherical masks using the annotated diameter on the corresponding centroid. These masks are quite accurate on smaller nodules, but not so much when the diameter increases (>15 mm). These spherical masks have a voxel size of 1x1x1 and a resolution of 400x400 in their axial plane.

As input, we used the axial slices of the CT scans, clipped with their matching lung mask. Values outside the mask were set to -4000 HU (an artificially low

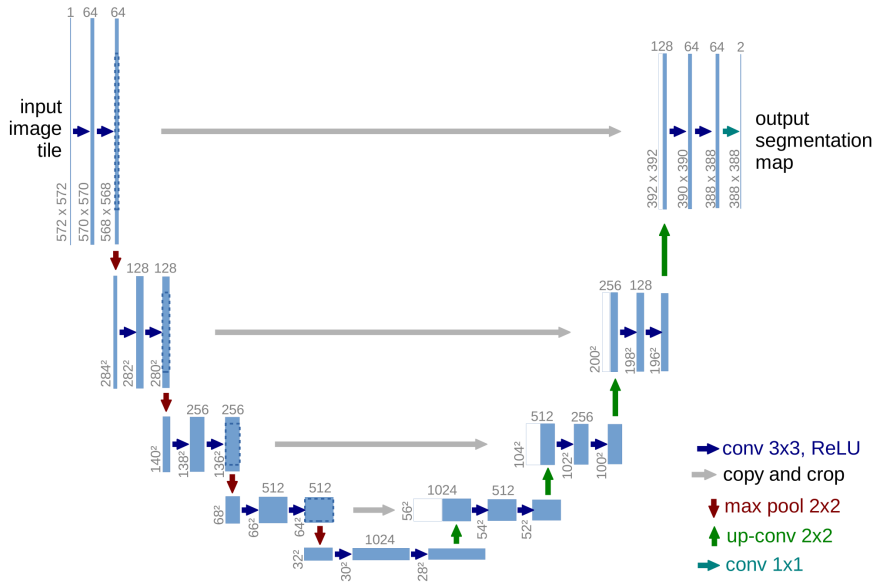

Figure 2.2: Overview of the U-Net architecture. Image from [16].

number, without an equivalent substance). This was a way to tell the network that areas outside the lung were not of our interest. Also, the same lung mask is used to clip the nodule masks, as nodules around the parenchyma could appear otherwise outside the lung, which would confuse the network since it has been clipped.

Only slices with visible nodules were used to train this network. Slices without abnormalities were discarded. This was done to correct the class imbalance. At the slice level we used the Dice coefficient to compute its score, but evaluating a scan requires to:

1. Apply the segmentation network over the whole scan
2. Label the predicted mask using connected components [26, 27, 28]
3. Extract the centroid of such labels
4. Convert the coordinates of the centroid to the real world coordinates of the original scan
5. Check whether the Euclidean distance between a candidate and any of the scan annotations is within its radius. If it is, count that candidate as a True Positive. Otherwise, it is a False Positive.

The evaluation of the system reports two metrics: sensitivity and average false positives per scan. Our main goal is to achieve the highest possible sensitivity, but reducing false positives will simplify the task of our false positive reduction module, so it is worth keeping track of its score.

There is also a set of variations in the preprocessing that we have applied incrementally so that we could study their individual impact in the performance analysis. A different network has been trained from scratch for each of those

variations, using both a binary cross entropy and Dice as loss functions. The variations are:

- **normalization:** Train the network with and without the use of batch normalization in its convolutional layers.
- **augmentation:** Enable the use of randomized image augmentation. Full description of the parameters in Table 2.1.
- **3 channels depth:** Use the 3 color channels of an image to pass the current slice along its two contiguous slices.
- **Laplacian filter:** Add a Laplacian filter to the original slice to increase edge contrast. This makes nodules easier to delimit.

Table 2.1: The range of transformations randomly applied to the axial slices used in the nodule segmentation training.

transformation	range
rotation	[-10º, +10º]
shearing	[-20%, +20%]
scaling	[-20%, +20%]
flip vertically	[True, False]
flip horizontally	[True, False]

Apart from studying the impact on the performance of each of the variations, we will also analyze the diversity of the resulting candidates, to ponder the usefulness of ensembling them into a more complete model.

2.3 False positive reduction

2.3.1 Handpicked feature classifier

2.3.1.1 Selected features

As seen in the previous chapter, the probability map obtained by the segmented slices is not informative enough to calculate the likelihood of the predictions. However, the shape of the labels themselves potentially holds information that can help us distinguish between real and false nodules. As an illustration, consider the segmented nodules A and C in Figure 2.3. The first one is an example of a large nodule, mostly round, mostly contiguous in the Z-axis. Nodule C, on the contrary, while having a spherical segmentation in the axial plane, is almost flat, which typically translates to a false positive. Another frequent source of false positives is caused by the presence of airways in the lung. On a single slice they can be easily mistaken for a nodule, but if we pay attention to their coronal and sagittal projections we will appreciate large displacements, forming an elliptical shape. This effect can be observed to some degree in nodule B, and more aggressively in nodule D.

Based on the visual inspection of the masks obtained by our segmentation, we engineered the following features to characterize the nodules:

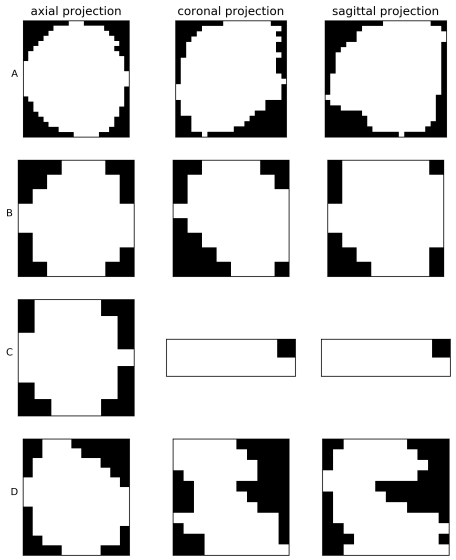

Figure 2.3: axial, coronal and sagittal projections of 4 nodule masks as segmented by our U-Net network. Even though the axial projection is similar in all the examples, the coronal and sagittal views offer a much larger degree of variance.

- **diameter:** measures diameter (in mm) of the bounding box in the axial plane.
- **layers:** measures number of contiguous layers of the bounding box in the z-axis.
- **squareness:** measures how similar the shape is between the axial and its orthogonal planes. Values range between 0 and 1. Value 0 means ratio between axial and the orthogonal planes (sagittal and coronal) is the same. Value 1 indicates that one side is completely square, while the other flat. Formulated as (l, w and d are shorthands for length, wifth and depth):

$$squareness(l, w, d) = abs\left(\frac{min\,(w, l)}{max\,(w, l)} - \frac{min\left(d, \frac{w+l}{2}\right)}{max\left(d, \frac{w+l}{2}\right)}\right)$$

- **extent:** measures the ratio between the masked and unmasked area in a labeled bounding box. Formulated as:

$$extent = \frac{num\ masked\ pixels\ of\ bbox}{num\ total\ pixels\ of\ bbox}$$

- **axial eccentricity:** measures the geometric eccentricity of the segmented nodule projected on the axial plane. Value 0 indicates the projection is a perfect circle.
- **sagittal eccentricity:** measures the geometric eccentricity of the segmented nodule projected on the sagittal plane. Value 0 indicates the projection is a perfect circle.

It should be noted that these features are only capturing basic information about the shape of the segmentation. This model ignores texture or other finer-grained features based on shape.

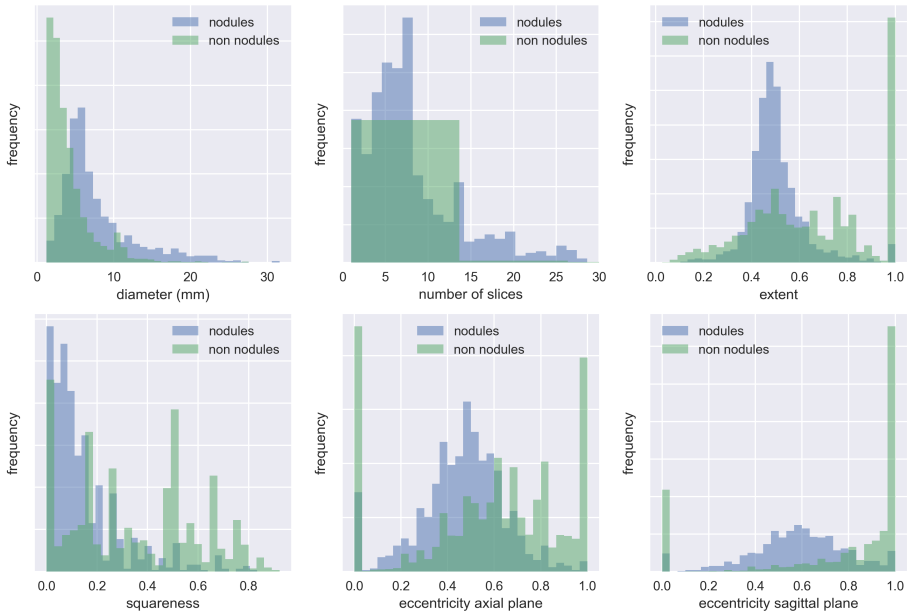

Figure 2.4: frequency distribution of the nodule candidates features, obtained by segmenting the entire LUNA dataset with the *augmented, 3ch, batch normalized, binary cross-entropy U-Net*. The histograms of true positives (TP) and false positives (FP) are overlapped and normalized.

2.3.1.2 Training the model

We train multiple binary classifiers with the features presented above and compare their performance quantitatively employing the area under the ROC (AUROC). The ROC curve indicates the true positive rate (sensitivity) as a function of the false positive rate. We plot the entire ROC curve to qualitatively assess the behavior of the classifier as the false positive rate increases. The tests will be performed both on the training and test sets, so we can also compare the performance of both side-by-side and assess the tendency to overfit of each of the classifiers.

The training and testing performed on the candidates obtained by the segmentation network based on binary cross entropy plus all variations (see the previous chapter). Candidates from subsets 0 to 8 are used as training data, while candidates in subset 9 serve as the test dataset. No validation set will be employed in this part of the pipeline, as we're not tuning the hyperparameters of the different classifiers. This basically leaves us a dataset with a 4 to 1 ratio in FP vs TP that we will not rebalance. More details about the dataset can be found in Table 2.2.

Table 2.2: Baseline from running the segmentation network. The classifier will be trained and evaluated on the features extracted from those candidates.

	Training (subsets 0 to 8)	**Test (subset 9)**
number of scans	776	84
number of candidates	5415	599
TP	1032	93
FP	4383	506
average FP per scan	5.6482	6.0238

We have selected a list of 5 classification algorithms (see Table 2.3), from simple logistic regression models to more advanced tree boosting classifiers, in an attempt to understand what sort of classification strategy works best both in terms of performance and generalization. We have used the *scikit-learn* [29] implementation of those algorithms, initialized with default parameters, for training and evaluation purposes.

Table 2.3: Types of classifiers trained on the candidates' dataset

Classifiers
Logistic regression
Decision tree
Random forest
AdaBoost
Gradient boosting

2.3.2 Radiomics-based classifier

We wanted to further investigate the applicability of radiomics [5] to discriminate between nodules and false positives. In a radiomics-based classifier, instead of training it based on manually handpicked features, we have used the software package pyradiomics [30] to automatically extract 105 features from each nodule segmentation. Instead of comparing different classification algorithms, we have used the same AdaBoost with different subsets of radiomic features. We have tested its predictive performance with 105 features, 20 and 5 (the last two cases after applying a dimensionality reduction with PCA). This allows us to determine how much of a predictive advantage we obtain in comparison to the features we have manually engineered. A complete list of the extracted features can be reviewed in [31].

Methodology-wise, the only difference in reference to the system trained on the handpicked features lies in the feature extracting process. We have transformed the scan segmentation masks to individual nodule masks, each of which is used in conjunction with the scan to automatically extract the features.

2.3.3 ResNet based classifier

We train multiple volumetric ResNet networks (see Figure 2.5) with different depths and compare their performance quantitatively employing the AUROC. As before, both training and testing curves are plotted side by side, to assess the overfitting of the model.

Regarding the network architecture itself, we introduced the suggestions by [21] and added a batch normalization and ReLU layer before each convolutional layer on the residual module, to facilitate convergence and weight stability while training. The same network was trained on different layer depths: 34, 50, 101 and 152.

As training data, we use the annotations provided by LUNA for the false positive reduction track of the challenge. They contain the world coordinates of the candidate centroid and a label indicating whether or not it is a nodule. See Table 2.4 for details regarding the distribution of this dataset. We evaluate the model against the candidates obtained by the same segmentation network as in the previous section so that we can compare the performance between the two different methods.

Table 2.4: The number of entries per class in the candidate annotations dataset, divided by split. The class imbalance between the two categories is very prominent, which we have to take into account when training the network.

dataset split	FP	TP	ratio
training (subsets 0 to 7)	603345	1218	495 to 1
validation (subset 8)	74293	195	381 to 1
test (subset 9)	75780	144	526 to 1

Since we are not using an ensemble of multiple models, the volumetric patch we use as input should capture the entire nodule. Based on the data observed in Figure 2.4, the dataset does not contain diameters above 32 mm, so we set the input resolution to patches of 32x32x32 voxels. The scans have been rescaled to a spacing of 1x1x1 mm and the images only have 1 color channel, with values corresponding to the Hounsfield value of the voxel (no normalization or clipping applied in the preprocessing).

The training is performed for a maximum of 50 epochs, only saving the weights in the iterations with better validation loss. We use Adam as our method for stochastic optimization, initialized to a learning rate of $1e^{-3}$. Early stopping is applied if the validation loss is not shown to improve in 10 consecutive epochs. The batch size for ResNets {34, 50 and 101} was 64, while the batch size for ResNet 152 was 32 due to memory constraints on the GPU side. Binary cross-entropy was used as the loss function. The hardware employed during training consisted on an Intel i7 7700, 32GB of RAM and a Nvidia 1080Ti GPU.

To offset the data imbalance observed in the dataset (see Table 2.4) we will oversample the nodule annotations with replacement so the training and validation

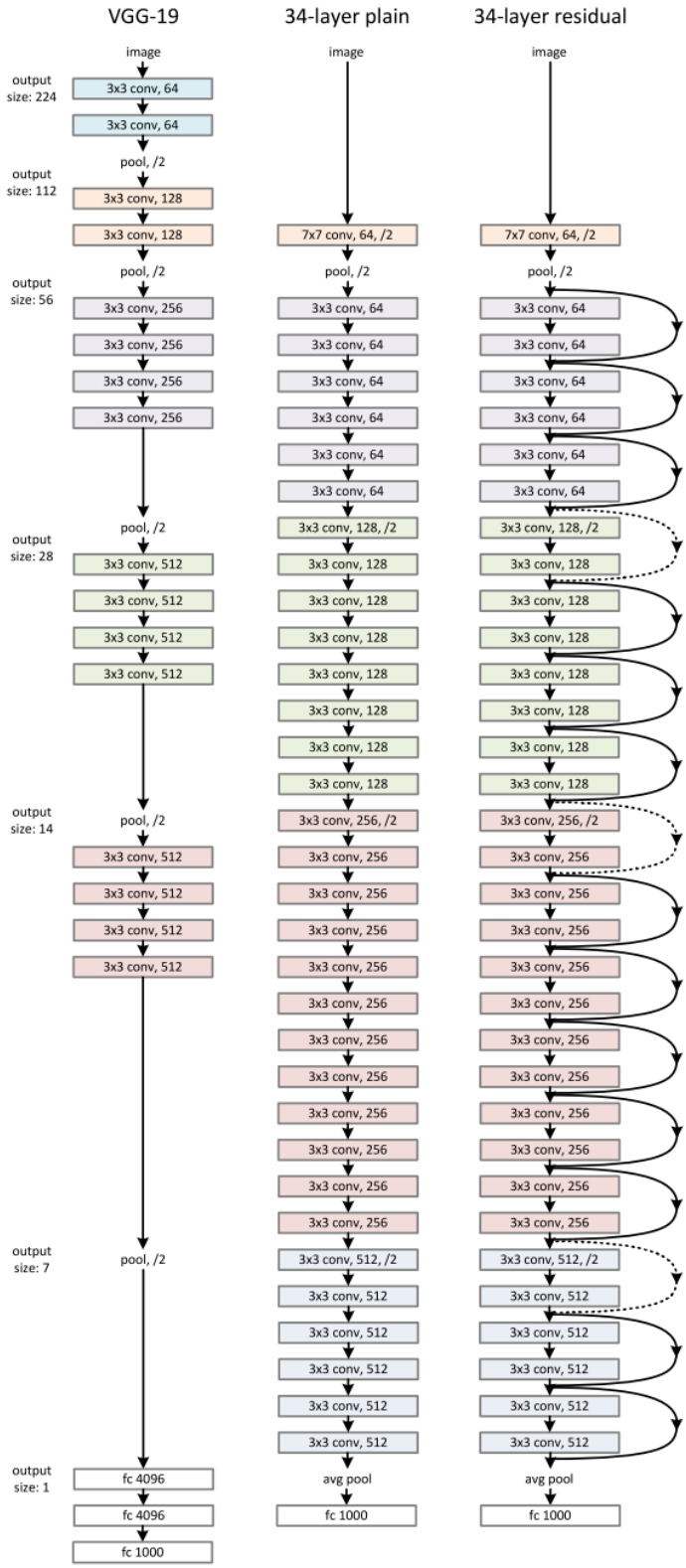

Figure 2.5: ResNet architecture (right) vs other image recognition models: VGG-19 (left) and a a plain network with 34 parameter layer (middle). Image from: [19].

ratio is 2 to 1 (FP vs TP). This effectively means that a nodule annotation will be seen during training 250 times per each non-nodule one, which causes overfitting in the network. We mitigate this effect by using 3D image augmentation. As detailed in Table 2.5, affine transformations are randomly applied to the input cube before passing it to the neural network. Since this transformations would be lossy if applied to the actual cube of 32x32x32, we actually retrieve a larger patch of 46x46x46, apply the augmentation, and return a centered view of 32 pixels per side. The augmentation cube side needs to be larger than the diagonal of the input one to be valid. The augmentations are randomly applied to each sample each time and the dataset is shuffled on each epoch.

Table 2.5: The range of transformations randomly applied to both the axial and coronal planes of the input volume

transformation	range
rotation	[-90º, +90º]
shearing	[-20%, +20%]
scaling	[-10%, +10%]
flip vertically	[True, False]
flip horizontally	[True, False]
translation width	[-2 px, +2 px]
translation height	[-2 px, +2 px]

Due to hardware limitations, training and validations were performed on a smaller fraction (35%) of the original data. Extracting small patches of data from a much larger image is only fast if such an image is already loaded, so we reduced the dataset size until it could fit in memory (32GB). Preloading the scans in-memory instead of reading them from disk speeded up the training in more than 2 orders of magnitude per epoch, so we considered the trade-off worthwhile.

2.4 LUNA performance comparative

Once the individual systems have been evaluated, we choose the best variations of nodule detector and false positive reduction and rank it according to the LUNA grand challenge rules. For this, we obtain the FROC curve (which indicates the true positive rate as a function of the number of false positives) at the average false positives rates between 0.125 and 8. Also, we will report the average sensitivity at the selected false positive rates of {0.125, 0.25, 0.5, 1, 2, 4, 8}.

It should be noted that there is a set of excluded annotations available in the LUNA dataset that neither count as false positives nor as true nodules. Any candidate matching one of those annotations need to be ignored. Another caveat of our particular comparison is the fact that the challengers in the LUNA scoreboard train their models performing a 10-fold cross validation over the whole dataset, and then evaluate their results on all the annotations, using 1000 bootstraps. The reported metrics for our system is only calculated over the test

split (subset 9 of LUNA), on a model that has not been cross-validated due to time and resources constraints. Nonetheless, we think it is a fair comparison in the sense of not being biased in our favor.

Chapter 3

Results

The results chapter is divided into five sections. The first three sections report the metrics for each individual problem of the CAD pipeline, putting special emphasis to the differences in performance between different approaches. The 4th section compares the metrics of the system against other competitors of the LUNA grand challenge to contextualize it within the state of the art. Finally, there is a 5th section that qualitatively assesses the efforts placed to integrate the system into a clinical context.

3.1 Lung segmentation

U-Net for lung segmentation achieves a Dice score over 98% in 40 epochs of training, although it only takes two epochs for the scores to be above 96%. As we can see in Table 3.1, there are no signs of overfitting. In fact, the Dice coefficients for both the validation and testing sets are 0.5% better than the results on the training dataset, although this could be explained due to the extra variability found in the larger training dataset (616 CT scans vs 88 each on the validation and test splits).

Table 3.1: mean Dice coefficient for each dataset split. For each scan, the mean Dice score is computed by individually evaluating each axial slice. Then, the mean score of the dataset is obtained by averaging the scores of all scans. .

dataset split	**Dice score**
training	0.977
validation	0.983
test	0.984

The U-Net lung segmentation is especially accurate in the superior and middle lobe of the lung, as we can appreciate in the bottom left slice on Figure 3.1. Irregular areas with lower contrast, like the one in the left lung shown in the

bottom right slice (same figure), can confuse the network. The network errs to towards more expansive masks, which in our case is a valid trade-off, since this is only a preprocessing step done towards reducing the complexity of the nodule segmentation task, and we don't want to discard lung mass (which may contain a nodule).

Lower lung lobes are generally those with lower Dice scores (see top left and right slices in Figure 3.1). It is also possible to observe holes inside a segmented lung (bottom right), which could be fixed by applying a morphological closing in the mask.

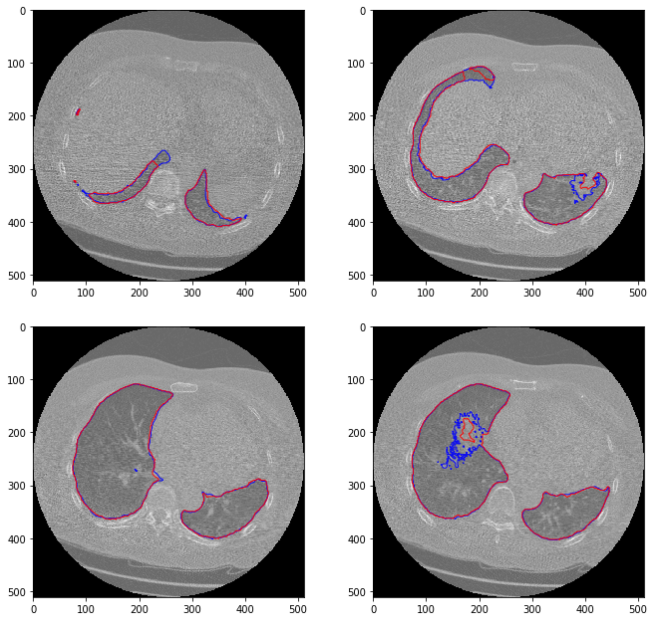

Figure 3.1: Axial slices of a lung with both masks superposed. The blue segmentation corresponds to the ground truth used to train the U-Net, while the red segmentation is the prediction returned by the model.

We chose to employ the same architecture we had been using to segment nodules for the lungs themselves and it is interesting to see how well this network architecture adapts to a different segmentation problem. Even the current issues that the network has segmenting the lower lobule could potentially be fixed by rebalancing the dataset to oversample that part of the lung. Some segmentation holes in the mask could be corrected (e.g. using a morphological closing) and other issues around edge detection could be improved by applying a Laplacian filter to enhance the edges, as well as oversampling the slices yielding worse performance.

3.2 Nodule detection

Our main metric for the nodule detection module is its sensitivity. This determines the upper limit performance of the system. We also keep track of the number of false positives returned since this directly affects the complexity of the false positive reduction module. More false positives require a more complex model in order to be competitive with the state of the art.

Table 3.2 shows both metrics divided by the loss function used during the training phase and the different image processing variations applied. We have not included the figures of the network trained without batch normalization as they were not very telling themselves, but we still wanted to report those negative results, because they were the key that allowed the network to learn. As we can see in Figure 3.2, a U-Net without normalization, neither on the input image nor on its convolutional layers, is incapable of learning the true representation of a nodule. Basically the only information it can extract from the original image is a rough segmentation of the lung parenchyma, which happens to be the area of major contrast in the original slice (as a reminder, a lung mask is applied as a preprocessing step, fixing the value of any voxel outside the lung tissue to -4000 HU). It was only by applying batch normalization to the U-Net convolutional layers that the network architecture was able generalize.

Both loss functions achieve similar top sensitivity scores in Table 3.2. In general, binary cross entropy displays a more stable behavior during training and it penalizes false positives more heavily, as we can see from the false positive rates of both networks (3 to 1 ratio favoring binary cross entropy). The downside of this heavy penalization of false positives is a slight drop in sensitivity (*0.915* vs *0.930*) that caps the maximum performance of the system.

It is also worth mentioning the effect of applying augmentation to mitigate overfitting. If we take a look at the differences between the augmented binary cross entropy variation vs the non-augmented (Table 3.2), the differential in training sensitivity barely achieves a 0.5%, but the gap between the training and testing scores goes from a 19% to 11%, and down to 6% in its best performing variation.

We also analyzed whether the different variations introduced diversity in the nodules detected by the network. As we can see in Figure 3.3, the nature of each variation is purely additive. The network is able to detect more nodules while still discerning the previous subset. In practice, this would mean that we might need to develop an entirely different model to detect the nodules we are currently missing so that an ensemble based on both models would beat their individual performance.

Finally, we compared the individual performance of our best segmentation networks against the results presented in the LUNA16 challenge survey [11] in Table 3.3. Our network is able to beat the individual systems described in the paper both in sensitivity and in the number of false positives reported by a scan. It is especially in this last metric where the differences are the most striking. The U-Net trained with binary cross entropy is able to match the sensitivity of ETROCAD (best reported) within 1%, while using 47 times lesser amount of candidates. The levels of accuracy provided by our network enable us to develop

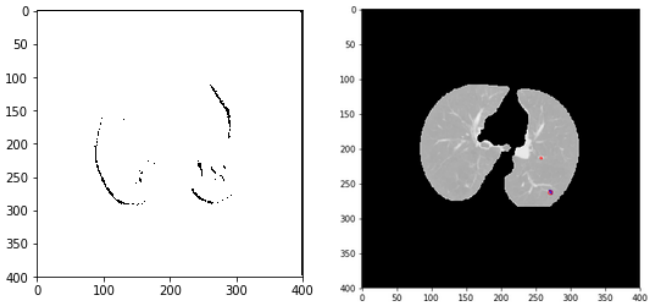

Figure 3.2: On the left, we see the nodule segmentation network results when the U-Net is trained without applying batch normalization on each convolutional layer. On the right, we see the output of the same network after enabling batch normalization. The red contour corresponds to the predicted mask while the blue markings match the ground truth used for training.

Table 3.2: U-Net nodule segmentation variations along with their sensitivity and number of FP per slice. It should be noted that nodule candidates may refer to the same annotation, so it is possible for the sensitivity to take values over 1.

loss function	variation	set	sensitivity mean	FP mean
cross-entropy	no augmentation, normalization	test	0.783	7.329
		train	0.977	6.707
		validation	0.796	6.840
cross-entropy	augmentation, normalization	test	0.859	6.011
		train	0.972	5.703
		validation	0.922	5.4886
cross-entropy	augmentation, normalization, 3ch, Laplacian	**test**	**0.915**	**5.750**
		train	0.974	5.515
		validation	0.940	5.181
dice	no augmentation, normalization	test	0.740	7.125
		train	0.828	7.063
		validation	0.795	6.784
dice	augmentation, normalization	test	0.339	1.443
		train	0.390	2.252
		validation	0.399	1.750
dice	augmentation, normalization, 3ch	test	0.803	34.125
		train	0.818	34.228
		validation	0.806	33.715
dice	augmentation, normalization, 3ch, Laplacian	**test**	**0.930**	**15.420**
		train	0.944	17.234
		validation	1.044	14.193

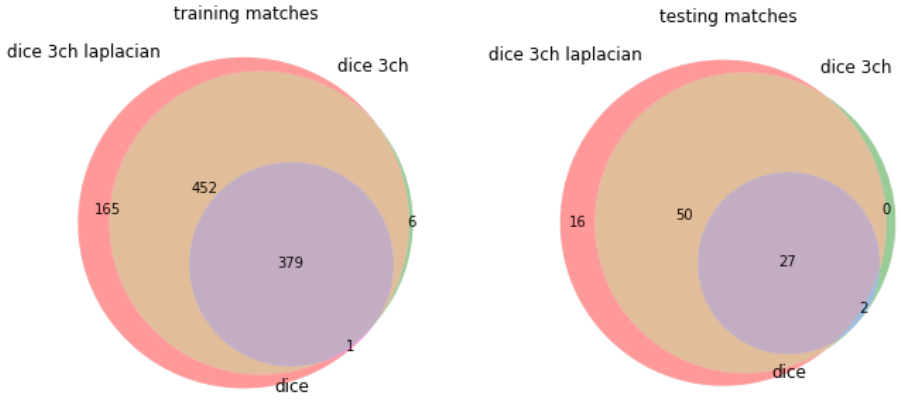

Figure 3.3: This Venn diagram showcases the additive nature of the variations performed in the U-Net. The lack of diversity in its predictions discouraged the use of an ensemble to increase its overall performance.

false positive reduction methods based on the nodule segmentation themselves, and still be competitive in the general LUNA scoreboard.

Table 3.3: Candidate detection systems performance as reported by 11. Even though each individual system is offering worse performance than our custom U-Net, an ensemble combining them reported sensitivity rates up to 0.983.

system	sensitivity	avg num candidates / scan
ISICAD	0.856	335.9
SubsolidCAD	0.361	290.6
LargeCAD	0.318	47.6
M5L	0.768	22.2
ETROCAD	0.929	333.0
lucanode binary cross-entropy	0.915	7.0
lucanode Dice	0.930	18.0

We visualize the currently undetected nodules and understand what similarities do they share. If they are mostly non-solid nodules, we could find another image filter that highly increases their contrast. If that does not help, it might be time to consider using a 3D U-Net architecture, or maybe an object detection network (based on an image recognition model) instead of a segmentation one. The intuition behind this affirmation, apart from our review of the state of the art, is also based in the resulting masks for the false positive candidates. They tend to be too flat, which is a sign that the network is not capturing depth properly. In this case, it might also be worth it to train another 2D segmentation layer over another one of the CT planes (i.e. sagittal). Once that would be ready, we could use both models to create an ensemble that better captures the shape of a nodule.

This section has been a practical demonstration of how small details can make or break a network's performance. Originally, we were trying to train the U-Net without a batch normalization layer, which did not converge. That was one of the multiple times where the non-linear progress on the development of this project came into full display. There are also other normalization strategies, such as [32], that might work best when the batches are small due to memory constraints.

3.3 False positive reduction

Figure 3.4 shows the resulting ROC curve for the different classifiers trained on the handpicked features of the nodule segmentation. Both classifiers based on trees (decision tree and random forest) overfit, and while the logistic regression does not, it does not perform as well as the boosting classifiers.

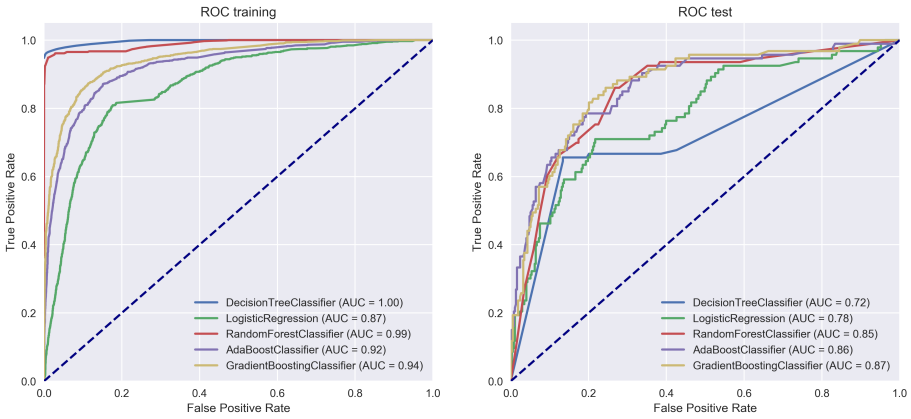

Figure 3.4: ROC curves and AUC of the handpicked feature classifiers.

Figure 3.5 shows the probability histogram of both nodules and non-nodules. As expected, even though the distributions are different in all classifiers, only in the boosting algorithms there is a clear distinction between the two classes, which translates to its performance in the ROC curve.

We have performed the same tests on the radiomics-based classifier, as seen in Figure 3.6. In this plot, instead of testing different classifiers, we have decided to plot the ROC of the same classifier trained on different subsets of radiomic features. As we can see, having more features does not necessarily translate to better performance. In fact, there is a sweet spot around 20 (vs the original 105). Still, the same figure also shows that the radiomics classifier is lagging behind the one based on only 6 handpicked features (Figure 3.5).

Figure 3.7 shows the results for a classifier based on a 3D ResNet trained at different depths. Apart from offering the best performance of the three systems, the AUC of both training and testing is almost identical, being the set of classifiers with the least amount of overfitting. Still, it should be noted that this classifier model has been trained on a dataset two orders of magnitude larger

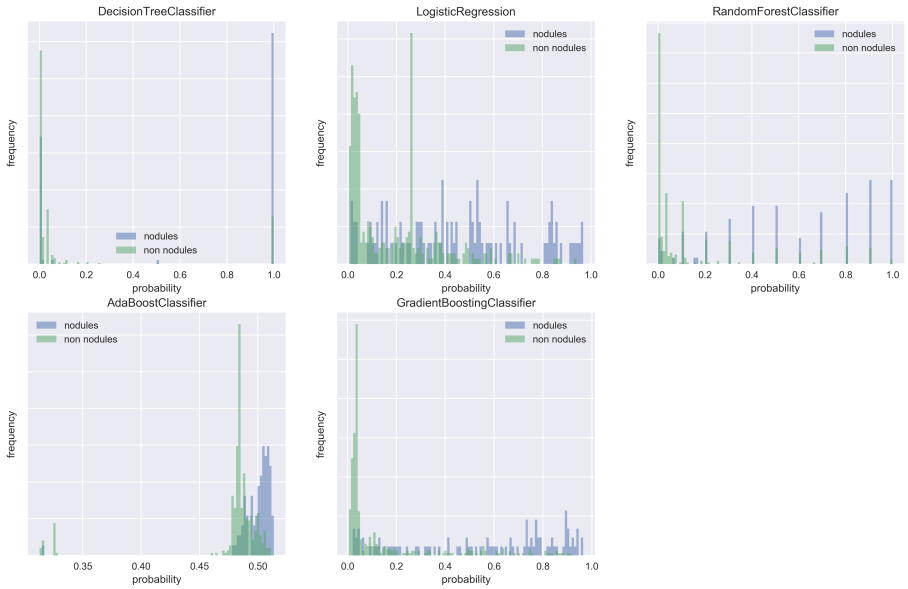

Figure 3.5: Probability histogram for handpicked features

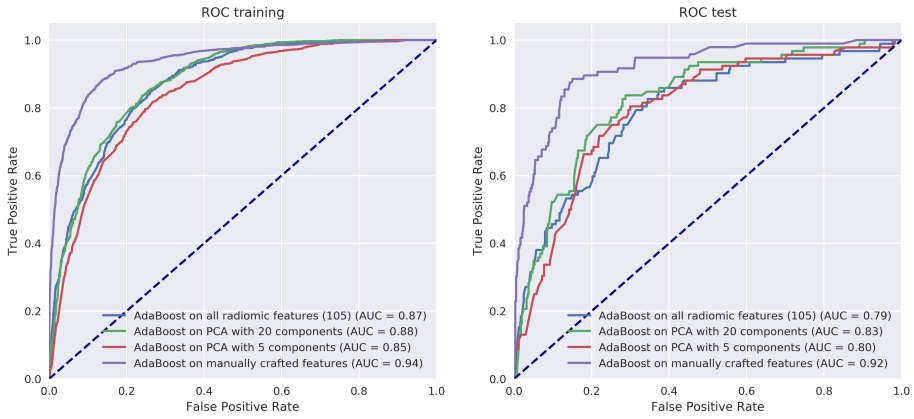

Figure 3.6: ROC curves and AUC of the radiomics based classifiers.

than the previous two approaches. The ResNet network has signs of saturation as its depth increases. Past 50 layers, AUC slightly decreases, and even during its training phase, it was rare for the validation loss to reliably decrease on each epoch.

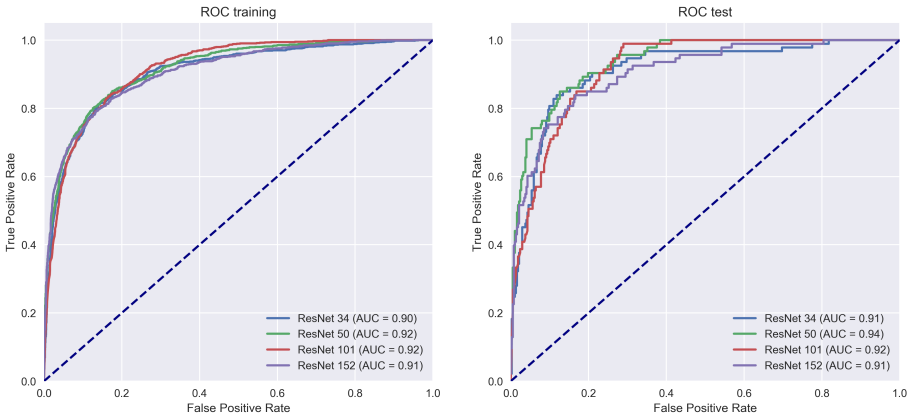

Figure 3.7: ROC curves and AUC of the ResNet based classifiers.

Figure 3.8 displays the ResNet 50 and the handpicked based classifiers side by side. As expected, the ResNet has a better AUC and plateaus at a lower false positive rate than the AdaBoost classifier, which is a very desirable property for these kinds of systems.

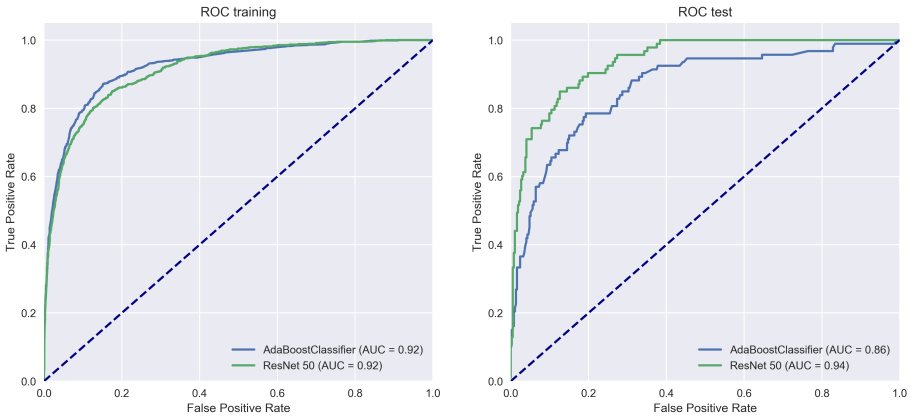

Figure 3.8: ROC curves and AUC comparing the best 2 variations of FP reduction method.

The classifier based on handpicked features was an interesting use case of the remarkably low number of false positives returned by our U-Net model. It was only thanks to the strength of the original segmentation network that such approach could ever prove useful. Still, an approach like this is not without drawbacks. It simplifies the system, but it introduced dependencies between layers. Can we affirm that the same model would equally work if the underlying U-Net had been trained with another loss function? Or maybe another set of

masks? It probably would not, since the results are very much dependent on the original network. This feedback also introduces some questions regarding the generalization of the model (performance between training and testing suffers a noticeable drop, especially compared to the model trained with residual networks). Finally, the fact that we are relying on the output of the segmentation network also limits the number of examples we have to train the classifier, which caps its performance. The usage of image augmentation is a worthwhile idea to explore, although this would involve a considerable amount of image preprocessing.

To our surprise, the classifier based on automatically extracted radiomic features actually performed worse than our manually handpicked classifier. A possible explanation to this fact could be that the segmentation itself we use as reference is already far from a proper representation of a nodule, and this handicaps its performance (radiomics are usually applied on manual segmentation of tumors to characterize them, not to distinguish lesions from non-lesions). It is also interesting noting that a PCA with lesser components is able to outperform the original 105 features. This suggests that the scattershot approach of radiomics might not actually deliver on its promises. Still, we would like to hold our judgment on the technique unless applied on a dataset with proper segmentation and maybe another kind of classification task. For example, applying radiomics to predict malignancy from a set of nodule segmentations would probably provide a fairer appraisal of the technique.

Finally, we prove again why image recognition methods based on deep learning are the current state of the art. A volumetric convolutional network, based on a residual architecture, beats previous systems by more than 10% margin in the final LUNA FROC score. Even though the ResNet beats the other systems, we still face the problem that the deeper layered versions saturate (probably due to the small input size of the image). For this particular experiment, we have also had to face a few technical limitations that prevented us from fully utilizing the whole available dataset, which surely didn not help on the final score.

In an improved version of the system, it would be interesting to explore the impact of using multiple ResNets trained on multiple input fields. Having multiple models trained at different scales would be best to capture the inherent nodule heterogeneity. If we only use one model we have to ensure that the size of the input image is going to be big enough to capture the vast majority of nodules inside it, which might be a cube too big for the majority of nodules (below 8.5 mm in diameter). Also relevant for the false positive reduction track, we are using an isotropic resize to 1x1x1 mm, sacrificing resolution in the axial plane (originally around 0.6 mm). If the resize had been performed at 0.6 mm or 0.5 mm for this particular part of the pipeline, we would also have more information from the original image, which might positively impact in the overall performance of the system.

3.4 LUNA performance comparative

Figure 3.9 shows the FROC curves of the best two variations of FP reduction systems along with the U-Net nodule segmentation (binary cross entropy loss,

normalization, augmentation, three channels and Laplacian filters applied). The reported score is an average of the sensitivity at selected FP rates [0.125, 0.25, 0.5, 1, 2, 4 and 8]. Both average false positive ranges and metrics have been set by the LUNA grand challenge to enable fast and objective comparisons between CAD systems.

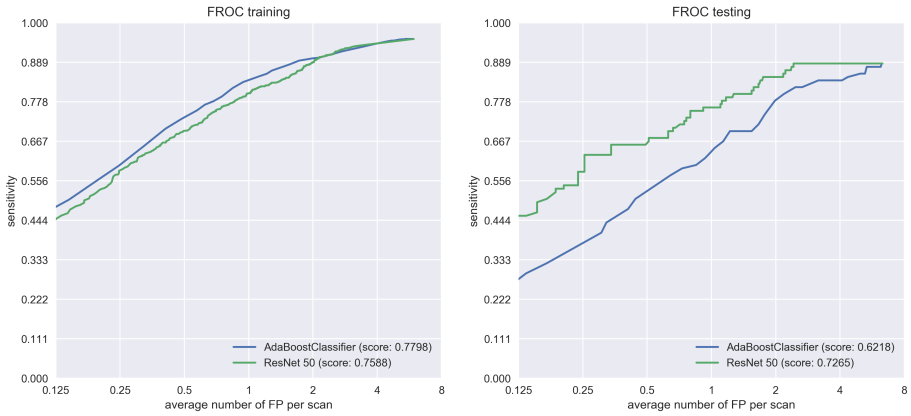

Figure 3.9: FROC curves and averaged sensitivity at selected FPR comparing the best 2 variations of false positive reduction.

Our system, *lucanode*, achieves a top-18 performance in the general LUNA leaderboard (as of June 2018, see Figure 3.10). Even the model with handpicked features would be worthy of a top-20, which is commendable for a system essentially based on a single model. In our favor, we would like to remind that our results come exclusively from the testing split (subset 9 of LUNA), while the other systems are the results of a model trained on a 10-fold cross validation and evaluated over a 1000 nodules with bootstrapping. We would expect that training the existing models in this manner, in addition to using the higher sensitivity segmentation U-Net (with Dice as its loss function) would bring us closer to the top 14. Still, even if we could easily improve the upper bound of our system, the slope of our false positive reduction is too steep compared to *ResNet's*, *ZNET's* or *PATech's*, so our performance at lower averages of false positives per scan would still be subpar.

All of these results have been obtained with the binary cross entropy variation of the segmentation network. Even though its sensitivity rate was 1.5% lower, it was 3 times as accurate, which pushed the overall score slightly above the variation trained with Dice. This also demonstrates that a good segmentation step simplifies the false positive reduction problem.

We have already commented on possible improvements in both nodule detection and false positive reduction in the previous two sections, so we are not going to cover the topic. Instead, we would like to focus on the importance of the FROC metrics to determine where the development efforts are best spent.

Compared to our immediate competitors, we have a competitive upper bound but the FROC curve is not as flat. This readily suggests that our false positive reduction system is not as competitive and should be the first component of the

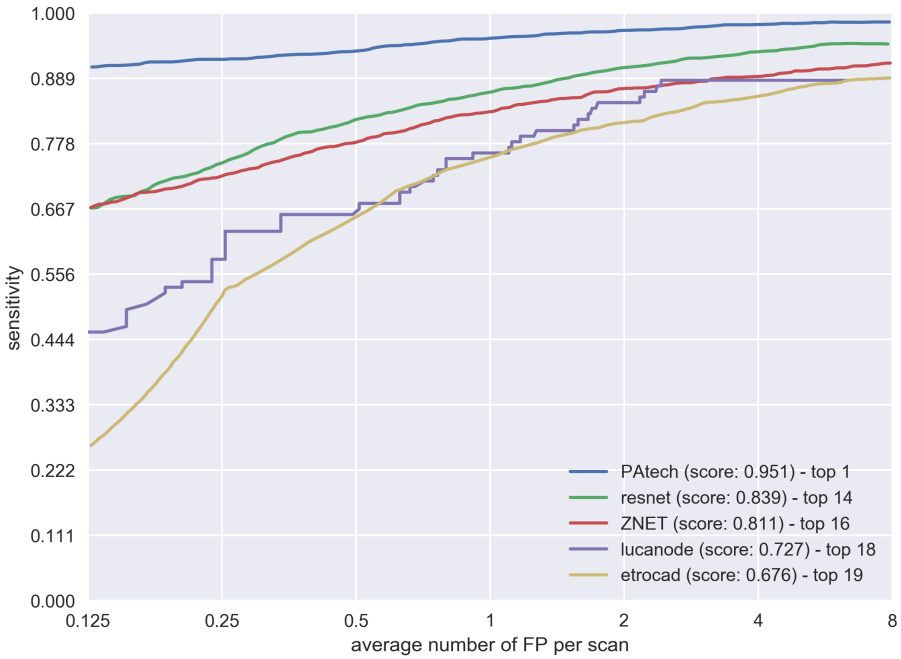

Figure 3.10: FROC comparison of lucanode with U-Net binary cross entropy + ResNet50 against other LUNA contenders.

system to be brought for assessment. This is just an example of how visualizations are key to gaining insights into the system, which has been a constant throughout the project.

It is also interesting to note how small gains in the ranking actually require an exponential amount of work. We can achieve top-20 results by employing a single U-Net model and engineering a few features from the resulting segmentations, but we require training a deep residual network for days to improve the ranking by 2 spots. Going further than that would surely demand longer training times, a more creative use of image preprocessing, fully switching to volumetric models and embracing ensembling, which would force us to deal with the vicissitudes of each one of its models, which just keeps raising the bar for such a system to be ever deployed in production.

3.5 Integration into a clinical workflow

The final objective for any CAD system is for it to be of clinical use. Ideally, a CAD such as this one would be part of the hospital protocol and automatically executed whenever a thoracic CT scan is performed. This would always provide a second opinion to the radiologist in charge of diagnosing the results and, hopefully, result in a lower *interobserver variability*

Too many times competition-winning models end up shelved because they are too

complex and is not computationally feasible to run them as part of a standardized protocol or, even more often, because there is no automated pipeline that can transform the raw input into usable answers for the clinician.

All of the publically available systems that participated in the LUNA challenged suffered from that same disease. The code was available but it was unusable. The network weights were not available and each step in the pipeline was fragmented and required their own manual setup. Since we did not want lucanode to share that same fate, we took steps towards automatizing those essential preprocessing steps. As we have mentioned, the efforts to automatically segment the lung are part of this endeavor. We have also paid special attention to package the software in a self-contained manner so that it would be easy to deploy on other hardware.

In our case, thanks to Conda and Docker, we have readily available images that contain both the code and the model weights to execute the *lucanode* system as a self-contained binary. There is even experimental support for Nvidia Docker [33], a containerization technology capable of using the underlying GPU resources of its host. In fact, even though we have not properly benchmarked the system for this thesis, the results of our effort are already live and have been presented in Albert Moral's graduation final project [34]. In Figure 3.11 we can see *lucanode* integrated as part of a medical web app platform.

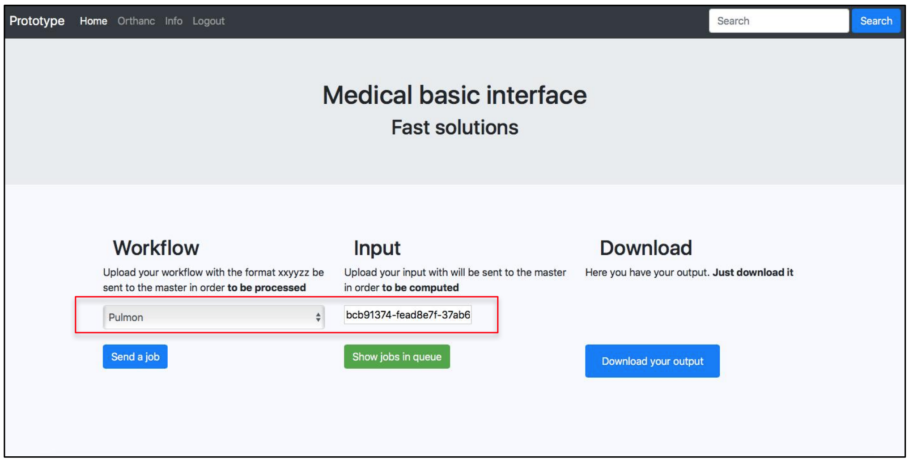

Figure 3.11: lucanode has already seen deployment inside an integrated web platform designed for medical use [34].

We would like to end this section to emphasize that having a nodule detection system working end to end is not only good for patients, but also to test the same model in different datasets. With the current pipeline it is possible to evaluate lucanode not only on LUNA, but also on other annotated datasets, and thus confirm whether the model would be applicable for the general population.

Chapter 4

Conclusions

4.1 Summary

In this thesis, we have introduced lung cancer, an aggressive and very heterogeneous disease, and also the deadliest among cancer patients in the US (among other countries). The main cause for the low survivability of lung cancer is due to its lack of early symptoms. Most often, by the time cancer has been detected, it is too late for treatment to be effective, leading to the patient's death.

A major screening trial set up in the US (NLST) demonstrated that the use of preemptive CT scan screenings on a set of the high-risk population improved the survivability up to 20%, which has caused different healthcare organizations throughout the world to consider its use as part of their health and safety protocols. This spike in the use of CT scans, along with the inherent difficulty of interpreting them has opened a new line of research in computer-assisted detection and diagnosis tools that make use of the CT imaging to predict the location of possible lung nodules (lung cancer manifest itself as a lung nodule in its early stages).

Our goal has been to create an automated workflow to convert a thoracic CT image into a set of coordinates. This set of coordinates would point to the location in the scan where the lesion is, which in turn would decrease the burden currently placed onto radiologists, and it would also decrease their *interobserver variability*, that is, the performance gap between an experienced radiologist and a resident, for example.

For such a CAD system to work, we need to solve two main problems. The first one is to detect nodule candidates on the CT scan. The second is to discriminate whether those candidates are actually a nodule or not. Since most of the information contained on a CT scan is not relevant to the task, we have also introduced a lung segmentation preprocessing step, which shrinks the area that the nodule detection module needs to take into account. This lung segmentation has been implemented with a deep learning segmentation architecture and is competitive with current segmentation methods based on thresholding and image registration.

On the nodule detection module, we have implemented multiple variations of a U-Net segmentation mask and discussed the effects of different parameters in its overall performance. We have also demonstrated how our approach is able to beat the performance of many of those systems, while also preserving a very low rate of false positives.

On the false positive reduction module, we have attempted different models. One of them was based on features extracted from the nodule segmentation, while the other employed a volumetric version of a popular image recognition deep learning architecture (ResNet). Our results show that the deep learning model is more competitive than the one based on mask features.

We compare the overall performance of the system against other CADs according to the LUNA grand challenge rules. Our final score would place us in the top 18, which corroborates the competitiveness of our solution against the current state of the art. Suggestions are also made regarding possible changes to further increase the performance of the current model.

Finally, we present the system as an integrated pipeline capable of generating a predictions file from a CT scan as input. By providing a clear interface, we allow our model to be easily pluggable with other platforms, as the work by [34] shows.

4.2 Future work

To improve the overall performance of the current system, the logical step forward is to create an ensemble of multiple models. There are also other paths worth pursuing. For example, the candidate detection network could be changed from a segmentation network into an object detection one. In this way, we could use the same model we employ for false positive reduction to detect the nodules themselves. It would be interesting to test the usage of an R-CNN [35] or a YOLO [35]. We are not aware of such architectures having been used to detect objects in a volume, so that could be a worthy contribution in and of itself.

Apart from incremental improvements on the performance of the current system, there is an angle which has not been explored, and that is inferring the malignancy of the nodules. The LUNA dataset does not contain such information, but the dataset it is based on (LIDC-IDRI) contains annotations by radiologists with the estimated malignancy per nodule, a handmade segmentation, and even the real diagnostic for a few of them. All this information could be used to transform the false positive reduction system into a malignancy predictor, so the system would not only detect abnormalities, it would also diagnose them.

4.3 General discussion

The most unexpected side effect of developing *lucanode* has been the paradigm shift from traditional software development, where you are iteratively coding an increasing set of rules to make the system smarter to training models, where you

are repackaging your data the way your network understands it best. In a way, training a deep learning model is just about finding the metaphor that makes it click. Data scientists, like comedians, need to find the right kind of humor for their audience.

I come from an engineering background and I am more used to blacks and whites than shades of grey, so the change has been quite a shock. On the one hand, the performance of such systems is first-class, but they also make you wonder about their maintainability in the long term. How can you fix what you can not explain? See [36] for more on the topic.

Another aspect I would like to mention is the brittleness of these models, especially on systems with pipelines that feed one another. Any error, anywhere in the pipeline, can invalidate the whole experiment. And it is very typical to have them, especially since there are shared elements that need to be set up in slightly different, but very significant, ways. It gets especially bad when you want to train and evaluate different variations of the same model side by side. You need to do it to properly test your hypotheses, but that in itself increases the error rate of the said experiment.

My best recommendation to the grievings above is to work on the visualizations. Whenever I have been stuck, whenever I wanted to confirm a result, I have always found solace in a good visualization. They are the best way to confirm your system works as intended and the best way to understand where it does not.

List of Figures

1.1 Axial view of a CT detector arc 3
1.2 A CT scan showing a lung nodule in a 3 plane view. 4
1.3 Different types of lung nodules. *left*: solid nodule, *middle*: part-solid nodule, *right*: non-solid nodule. Image from [10] 5

2.1 the lucanode pipeline 10
2.2 Overview of the U-Net architecture. Image from [16]. 12
2.3 axial, coronal and sagittal projections of 4 nodule masks as segmented by our U-Net network. Even though the axial projection is similar in all the examples, the coronal and sagittal views offer a much larger degree of variance. 14
2.4 frequency distribution of the nodule candidates features, obtained by segmenting the entire LUNA dataset with the *augmented, 3ch, batch normalized, binary cross-entropy U-Net*. The histograms of true positives (TP) and false positives (FP) are overlapped and normalized. 15
2.5 ResNet architecture (right) vs other image recognition models: VGG-19 (left) and a a plain network with 34 parameter layer (middle). Image from: [19]. 18

3.1 Axial slices of a lung with both masks superposed. The blue segmentation corresponds to the ground truth used to train the U-Net, while the red segmentation is the prediction returned by the model. 22
3.2 On the left, we see the nodule segmentation network results when the U-Net is trained without applying batch normalization on each convolutional layer. On the right, we see the output of the same network after enabling batch normalization. The red contour corresponds to the predicted mask while the blue markings match the ground truth used for training. 24
3.3 This Venn diagram showcases the additive nature of the variations performed in the U-Net. The lack of diversity in its predictions discouraged the use of an ensemble to increase its overall performance. 25
3.4 ROC curves and AUC of the handpicked feature classifiers. . . . 26
3.5 Probability histogram for handpicked features 27
3.6 ROC curves and AUC of the radiomics based classifiers. 27
3.7 ROC curves and AUC of the ResNet based classifiers. 28

3.8 ROC curves and AUC comparing the best 2 variations of FP reduction method. 28
3.9 FROC curves and averaged sensitivity at selected FPR comparing the best 2 variations of false positive reduction. 30
3.10 FROC comparison of lucanode with U-Net binary cross entropy + ResNet50 against other LUNA contenders. 31
3.11 lucanode has already seen deployment inside an integrated web platform designed for medical use [34]. 32

List of Tables

1.1 Ten leading cancer types for the estimated deaths, United States, 2018. Figures from [2] 1
1.2 Houndsfield Unit range for different body tissues and fluids. . . 2

2.1 The range of transformations randomly applied to the axial slices used in the nodule segmentation training. 13
2.2 Baseline from running the segmentation network. The classifier will be trained and evaluated on the features extracted from those candidates. 16
2.3 Types of classifiers trained on the candidates' dataset 16
2.4 The number of entries per class in the candidate annotations dataset, divided by split. The class imbalance between the two categories is very prominent, which we have to take into account when training the network. 17
2.5 The range of transformations randomly applied to both the axial and coronal planes of the input volume 19

3.1 mean Dice coefficient for each dataset split. For each scan, the mean Dice score is computed by individually evaluating each axial slice. Then, the mean score of the dataset is obtained by averaging the scores of all scans. 21
3.2 U-Net nodule segmentation variations along with their sensitivity and number of FP per slice. It should be noted that nodule candidates may refer to the same annotation, so it is possible for the sensitivity to take values over 1. 24
3.3 Candidate detection systems performance as reported by 11. Even though each individual system is offering worse performance than our custom U-Net, an ensemble combining them reported sensitivity rates up to 0.983. 25

Bibliography

1. WHO. Fact sheets about cancer. (2018).

2. Siegel, R. L., Miller, K. D. & Jemal, A. Cancer Statistics , 2018. **68,** 7–30 (2018).

3. Röntgen, W. C. Über eine neue Art von Strahlen. Sitzungsberichte der Physik.-med. Gesellschaft zu Würzburg: 132–141. *Annalen der Physik und Chemie* **64,** 1–11 (1895).

4. Doria-Rose, V. P. & Szabo, E. Screening and prevention of lung cancer. *Lung cancer: a multidisciplinary approach to diagnosis and management* **2,** (2010).

5. Gillies, R. J., Kinahan, P. E. & Hricak, H. Radiomics: Images Are More than Pictures, They Are Data. *Radiology* **278,** 563–577 (2016).

6. Ru, Y., Xie, X., Koning, H. J. D. & Vliegenthart, R. NELSON lung cancer screening study. 79–84 (2011). doi:10.1102/1470-7330.2011.9020

7. Al Mohammad, B., Brennan, P. C. & Mello-Thoms, C. A review of lung cancer screening and the role of computer-aided detection. *Clinical Radiology* **72,** 433–442 (2017).

8. Quekel, L. G. B. A., Goei, R., Kessels, A. G. H. & Engelshoven, J. M. A. V. Detection of lung cancer on the chest radiograph : impact of previous films , clinical information , double reading , and dual reading. **54,** 1146–1150 (2001).

9. Beyer, F., Ludwig, K. & Düsseldorf, M.-h. Detection of pulmonary nodules at multirow-detector CT : effectiveness of double reading to improve sensitivity at standard-dose and low-dose chest CT. 14–22 (2005). doi:10.1007/s00330-004-2527-6

10. Jacobs, C. *Automatic detection and characterization of pulmonary nodules in thoracic CT scans.* (2015).

11. Arindra, A. *et al.* Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: The LUNA16 challenge. (2017).

12. Murphy, K. *et al.* A large-scale evaluation of automatic pulmonary nodule detection in chest CT using local image features and k-nearest-neighbour classification. *Medical Image Analysis* **13,** 757–770 (2009).

13. Jacobs, C. *et al.* Automatic detection of subsolid pulmonary nodules in thoracic computed tomography images. *Medical Image Analysis* **18,** 374–384 (2014).

14. Setio, A. A. A., Jacobs, C., Gelderblom, J. & Ginneken, B. V. Automatic detection of large pulmonary solid nodules in thoracic CT images Automatic detection of large pulmonary solid nodules in thoracic CT images. **5642,** (2015).

15. Berens, M., Van Der Gugten, R., De Kaste, M., Manders, J. & Zuidhof, G. ZNET -LUNG NODULE DETECTION.

16. Ronneberger, O., Fischer, P. & Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. 1–8 (2015). doi:10.1007/978-3-319-24574-4_28

17. Tan, M., Deklerck, R., Jansen, B., Bister, M. & Cornelis, J. A novel computer-aided lung nodule detection system for CT images. *Medical Physics* **38,** 5630–5645 (2011).

18. Dou, Q., Chen, H., Yu, L., Qin, J. & Heng, P. A. Multilevel Contextual 3-D CNNs for False Positive Reduction in Pulmonary Nodule Detection. *IEEE Transactions on Biomedical Engineering* (2017). doi:10.1109/TBME.2016.2613502

19. Wu, S., Zhong, S. & Liu, Y. Deep residual learning for image recognition. 1–17 (2017). doi:10.1007/s11042-017-4440-4

20. Arindra, A. *et al.* Pulmonary Nodule Detection in CT Images: False Positive Reduction Using Multi-View Convolutional Networks. *Ieee Transactions on Medical Imaging* **35,** 1160–1169 (2016).

21. Chen, H., Dou, Q., Yu, L., Qin, J. & Heng, P. A. VoxResNet: Deep voxelwise residual networks for brain segmentation from 3D MR images. *NeuroImage* **170,** 446–455 (2018).

22. Kingma, D. P. & Ba, J. Adam: A Method for Stochastic Optimization. 1–15 (2014). doi:http://doi.acm.org.ezproxy.lib.ucf.edu/10.1145/1830483.1830503

23. Chollet, F. & Others. Keras. (2015).

24. Agarwal, A. *et al.* TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems. (2015).

25. Van Rikxoort, E. M., De Hoop, B., Viergever, M. A., Prokop, M. & Van Ginneken, B. Automatic lung segmentation from thoracic computed tomography scans using a hybrid approach with error detection. *Medical Physics* **36,** 2934–2947 (2009).

26. Walt, S. van der *et al.* scikit-image: image processing in Python. *PeerJ* **2,** e453 (2014).

27. Fiorio, C. & Gustedt, J. Two linear time Union-Find strategies for image processing. *Theoretical Computer Science* **154,** 165–181 (1996).

28. Wu, K., Otoo, E. & Shoshani, A. Optimizing connected component labeling algorithms. 1965 (2005). doi:10.1117/12.596105

29. Nielsen, D. Tree Boosting With XGBoost Why Does XGBoost Win "Every" Machine Learning Competition? *NTNU Tech Report* 2016 (2016).

30. Griethuysen, J. J. van *et al.* Computational Radiomics System to Decode the Radiographic Phenotype. *Cancer Research* **77,** e104–e107 (2017).

31. Griethuysen, J. J. van. PyRadiomics documentation - features. (2018).

32. Wu, Y. & He, K. Group Normalization. (2018).

33. NVIDIA. Nvidia docker. (2018).

34. Moral Lleo, A. (. Creacio d'un entorn tecnologic destinat al calcul de resultats de recerca mitjancant orquestracio al nuvol. (2018).

35. Ren, S., He, K., Girshick, R. & Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2017). doi:10.1109/TPAMI.2016.2577031

36. Sculley, D. *et al.* Machine Learning : The High-Interest Credit Card of Technical Debt. *NIPS 2014 Workshop on Software Engineering for Machine Learning (SE4ML)* 1–9 (2014). doi:10.1007/s13398-014-0173-7.2