YOLO Models for Automatic Lung Nodules Detection from CT Scans

Hristina Biserinska
STUDENT NUMBER: 2047192

THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF SCIENCE IN DATA SCIENCE & SOCIETY
SCHOOL OF HUMANITIES AND DIGITAL SCIENCES
TILBURG UNIVERSITY

Thesis committee:

Supervisor: dr. Sharon Ong
Second Reader: dr. Giovanni Cassani

Tilburg University
School of Humanities and Digital Sciences
Tilburg, The Netherlands
June 2020

Preface

I would like to express my gratitude to dr. Sharon Ong for agreeing to become my thesis supervisor. I would also like to thank her for the support, inspiration and honest opinion on my performance throughout the process.

YOLO Models for Automatic Lung Nodules Detection from CT Scans

Hristina Biserinska

In this work, I propose an application of an object detection algorithm - You Only Look Ones (YOLO) for lung nodules detection from Computed Tomography (CT) scans. YOLO is a convolutional neuron network, pre-trained on ImageNet, and usually used for real-time object detection. This application's purpose is to provide a second opinion to radiologists when analyzing CT scans. What distinguishes this work from existing research is its focus on the very small nodules that are hard to detect. These nodules help to identify lung cancer in its early stage when the cancer is most treatable. I experimented with 2 different architectures of YOLO (version 3 and version 4) and 3 image enhancement pre-processing pipelines. The training was performed with CT chest scans from a new dataset - Lung Nodule Database (LNDb) which was introduced few months prior, and hence not widely used yet for development of lung nodule detection algorithms. I showed that a model pre-trained on natural images can have promising results when applied in the medical domain. The nodule detection with this single neural network and the right contrast enhancement technique resulted in low false positives with reasonably high sensitivity.

1. Introduction

1.1 Motivation

Lung cancer is the leading cause of cancer-related deaths in the world for both men and women. As most lung cancers are found at an advanced stage, lung cancer cells have already spread widely, and making them very hard to cure. Lung nodules are the major radiographic indicator of lung cancer. The survival rate can be improved if the presence of lung nodules is detected early. This emphasizes the need for preemptive screening examinations to ensure that malignant nodules are found before symptoms begin. The screening examinations require radiologists to analyze computed tomography (CT) chest scans. Through their expertise in human anatomy and experience in looking for structures of interest, they identify nodules in the image and annotate them for further evaluation. However, repetitive analysis and precise detection of nodules consume huge amount of radiologist's effort and time. A Computer-Aided Detection (CAD) system for lung nodule detection can act as a second reader, assisting the radiologist in this scenario. These systems could facilitate the lung cancer screening workflow, which is efficient as well as cost-effective. In conventional CAD systems, the whole process of detection involves identification of candidate nodules and classification of such nodules based on features that differentiate a true nodule from a non-nodule (van Ginneken 2010; Messay 2010).
The task of lung nodule detection is particularly challenging because of the variability associated with the shape, size texture, and location of the nodules (Anirudh 2016).

Despite the many CAD systems for automatic detection of lung nodules that have been developed, there are still persisting challenges that need to be addressed. One major challenge is that nodules show up as relatively low-contrast white circular objects within the lung fields and are difficult to detect due to overlapping shadows from other structures such as vessels and ribs (Howard Lee 2015). Therefore, this research will experiment with different pre-processing contrast enhancement techniques and compare how the different pre-processing pipelines affect the overall prediction performance.
The second challenge comes from the fact that lung nodules have large variations in size, shape and location. The CAD system development so far relies mainly on the detection of nodules with a diameter of more than 3mm and does not take into account the smaller nodules which usually occur in the early stage of lung cancer. This research address this limitation on a dataset rich with small nodules.
The 3rd challenge of lung nodule detection comes from the many false positive candidates that carry similar morphological appearance to the true lung nodules. The current CAD systems often suffer from a high number of false positives. Therefore, this research aims to develop an algorithm that can localize as many nodules as possible, while keeping the number of false positives low.

1.2 Project Definition

In this work, an object detection application is proposed, using You Only Look Ones (YOLO) framework to detect nodules in CT scan images. YOLO is a convolutional neuron network that deals with object detection in a different way when compared to the already existing object detectors like R-CNN (Ross 2015b), Fast R-CNN (Ross 2015a), and Faster-R-CNN (Shaoqing 2015). YOLO is a one-stage detection algorithm which takes the entire image in a single instance and predicts the bounding box coordinates and class probabilities for these boxes.
In this research, the experiments were conducted with 2 different YOLO architectures – YOLO version 3 (v3) (Redmon 2016) and YOLO version 4 (v4) (Bochkovskiy 2020). The later was recently released in the end of April 2020. The two experiments were performed by applying transfer learning to the model. Transfer learning is a popular approach in deep learning where a model developed for one task, called pre-trained model, is reused as a starting point for a model on another task. In this research, transfer learning was used to speed up training and improve the performance of the models. Obtaining a large number of CT images which are annotated by radiologists is a very challenging task, therefore, an opportunity to improve the current CAD is to use transfer learning and initialize the weights of the new network during training with the weights obtained from other networks, pre-trained comprehensively on the large scale well-annotated datasets like ImageNet. As a result, medical image recognition tasks with limited amount of data can achieve higher accuracy with fewer computational requirements.
The two model architectures will be trained on the original images and on images pre-processed with 3 different image enhancement pipelines which will result in 8 models that will be compared to answer the research question and the two sub research questions.

1.3 Research Question

The research question to answer along the progress of this thesis is as follows. **"To what extent can YOLO models automatically detect lung nodules from CT scans"** with the following two sub research questions:

1. How do YOLO algorithms trained on images pre-processed with various image enhancement techniques compare with training on the original images?
2. Which YOLO architecture and image enhancement technique achieve fewer false positive detections?

Even though screening is essential for early detection of lung cancer and increased survival rate, the high false positive rate is one of the recognized potential harms from it. From one side, the false positive detections generate costs in terms of follow-up diagnostic procedures. In some cases the false positive test are followed up by needless invasive diagnostic procedures, such as a biopsy, which may lead to complications. In addition to the financial costs of diagnostic follow-up procedures and the possible medical complications from them, as well as the extra ionizing radiation from diagnostic imaging, false positive results may also generate patient anxiety and affect individuals' willingness to continue screening for cancer in the future (Pinsky 2015). A positive test not being correctly identified as a false positive can also result in unnecessary patient treatment. Therefore, it is critical to improve the discrimination of benign from malignant screen-detected lung nodules.

To answer these research questions, a recently released dataset - Lung Nodule Database (LNDb) (Pedrosa 2019) has been used. The LNDb dataset was introduced a few months prior, and hence not widely used yet for the development of lung nodule detection systems. It differs from the already existing datasets by the prevalence of small nodules. Creating a CAD system that is able to detect small nodules is crucial for finding a disease before the symptoms begin, at its earliest and most treatable stage. In addition, the new dataset gives focus to radiologist variability, meaning that the variability in the annotation process is particularly emphasized as annotations are conducted solely in a single-blinded manner, which replicates more closely the clinical reality where images are analyzed by a single radiologist.

1.4 Summary of contributions

The contributions of this research can be summarized as follows:

1. A detection system is developed for lung nodule localization and classification, that makes use of the performance and robustness of the CNN architecture of You Only Look Once (YOLO) (Redmon 2016), (Bochkovskiy 2020). This deep learning architecture can automatically detect lung nodules by learning the discriminative features of the nodules and predict the bounding box coordinates without using any handcrafted features like shape, size or texture.
2. This research focuses on the importance of detection system that is able to localize small nodules (< 3mm) and takes into account the clinical reality.
3. The effect which image enhancement techniques have on the detection performance is shown by applying 3 different pre-processing pipelines. The best result was achieved with pipeline 3 which applied normalization and median filtering.

4. During training, the proposed methods also make use of depth information present in the preceding and succeeding images of the scan image of interest, for nodule prediction.
5. The proposed method (YOLO v4 pipeline 3 - normalization and median filtering) achieves promising results on the LNDb dataset with 76% sensitivity at 0.35 false positives per image.

1.5 Thesis outline

In order to briefly outline the thesis structure, chapter 2 grants an overall theoretical background on computed tomography (2.1) and lung nodules (2.2), following up on deep learning methods developed for lung nodules analysis (2.3). Chapter 3 introduces the reader to the experimental set up of this thesis, elaborating on the dataset used (3.1), the pre-processing applied (3.2) and the label data preparation (3.3). Furthermore, a description of the deep learning architectures (3.4) and the training parameters (3.5) used to carry out the experiments in (3.6) and the evaluation criterion (3.7) to obtain the results will be provided. Finally, chapter 5 presents the obtained results which will be discussed in chapter 6. As a closing note, a conclusion will be provided in chapter 7.

2. Related work

2.1 Computed Tomography (CT)

Computed tomography (CT) is a diagnostic imaging test used to create detailed images of internal organs, bones, soft tissue, and blood vessels. The CT scanner is typically a large, donut-shaped machine with a short tunnel in the center. The patient lies on a narrow examination table that slides in and out of the short tunnel. The cross-sectional images generated during a CT scan can be reformatted in multiple planes and can generate three-dimensional images. CT scanning is often the best method for detecting cancers since the images allow doctors to confirm the presence of a tumor and determine its size and location. CT is fast, painless, noninvasive, and accurate (Victor 2014).
Computed Tomography (CT) of the chest uses special x-ray equipment to examine abnormalities found on conventional chest x-rays. Different body parts absorb x-rays in different amounts. The unit of measurement in CT scans is the Hounsfield Unit (HU), which is a measure of radiodensity. The Hounsfield scale assigns water a density of 0 Hounsfield units (HU) and air a value of 1000 HU. Dense materials such as bones have density values approaching +1000 HU. The Hounsfield density of tissues reflects their attenuation of x-ray and is proportional to their physical density. Because the human eye can perceive only a limited number of gray shades, the full range of density values is typically not displayed for a given image. Instead, the tissues of interest are highlighted by devoting the visible gray shades to a narrow portion of the full density range, a process called “windowing” (Broder 2011).

2.2 Lung Nodules

A lung nodule is a small abnormal area that is sometimes found during a CT scan of the chest. Lung nodules are small masses of tissue in the lung that appear as round, white shadows on a chest X-ray or CT scan. There are two types of growths in the lung called lung nodules - benign or malignant. While lung nodules are the major radiographic indicator of lung cancer, they may also be signs of a variety of benign conditions. Measurement of nodule growth rate over time is the most promising tool

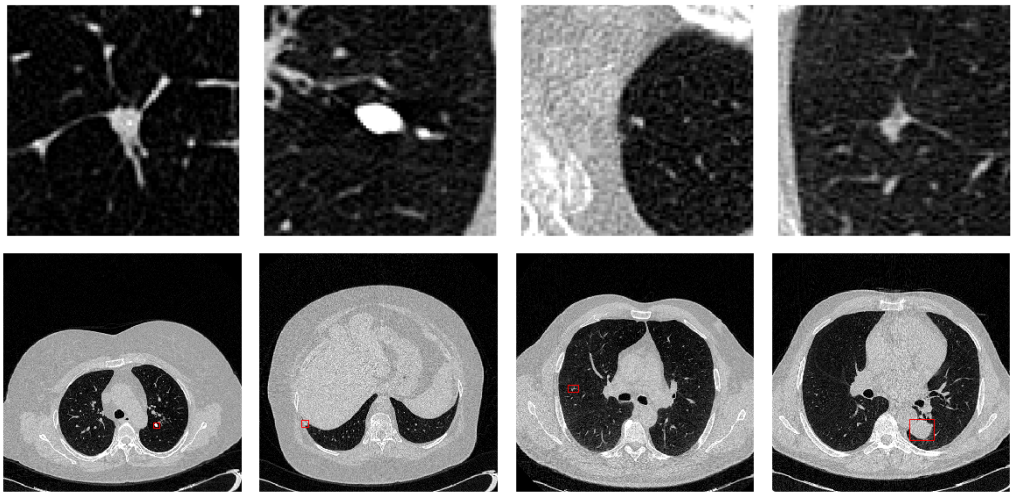


Figure 1: Lung nodules with different size, shape, and location, produced from the LNDb dataset for this research.

in distinguishing malignant from nonmalignant lung nodules. A larger lung nodule, such as one that is 30 millimeters or larger, is more likely to be cancerous than a smaller lung nodule (Brandman 2011). Figure 1 shows images of nodules with different sizes, shapes and location. They were produced from the LNDb dataset for the purpose of this research.

2.3 Deep learning methods for lung nodule analysis

In the last two decades, researchers have been developing CAD systems for automatic detection of lung nodules to analyze CT scans. They aim to improve the diagnosis accuracy, assist in cancer detection at its earlier stage and reduce the time of the radiologist in evaluation. Ginneken (2017) reviewed computer analysis in chest imaging and illustrates how the three types of approaches — rule-based image processing, machine learning, and deep learning — have been applied, and how deep learning is becoming the dominant approach with very promising results. The more recent confirmation of the increasing focus and importance of deep learning in lung nodules detection is published by Riquelme and Akhloufi (2020). In that review they present a summary of the recent deep learning techniques for lung cancer detection and conclude that most of the proposed CAD systems are based on deep Convolutional Neural Networks (CNN). Current mainstream object detection algorithms mainly utilize deep learning models, belonging to one of the following two categories:

1. Two-stage detection algorithm. It generates candidate region (region proposals) first, and then the candidate area classification can be updated by refinement, see e.g., R-CNN (Ross 2015b), Fast R-CNN (Ross 2015a), and Faster-R-CNN (Shaoqing 2015), etc.
2. One-stage detection algorithm. It directly generates the class probability and position coordinate values of objects, see e.g. YOLO (Redmon 2016) and SSD (Wei L 2016).

It is known that two-stage detection algorithms perform well in achieving high accuracy, one-stage detection algorithms are significant in less computational cost.
A relevant difference between the deep learning approaches is whether they are using a two-dimensional or three-dimensional architecture. 3D architectures demand the use of three-dimensional convolutions, which increase the number of parameters, the computational cost, and the training time considerably. For this reason, some approaches use 2D convolutions, which have fewer parameters and allow training deeper and more complex architectures with less powerful hardware.

2.3.1 2D Deep learning approaches. Zheng (2020) have developed a multi-planar nodule detection system using 2D convolutional neural networks, U-Net. By effectively combining results from three planes for the candidate detection task - axial, coronal, and sagittal, they managed to keep some of the contextual information of the CT scan. Their conclusion was that multi-scale features help dense convolutional neural networks to become more effective at removing false positives. Trajanovski (2018) presented a two-stage framework, in which the first part employs a nodule detector based on SVM. The second stage uses both contextual information about the nodule and nodule features as input to a CNN, inspired by a ResNet architecture, to estimate the malignancy risk of the entire CT scan.

2.3.2 3D Deep learning approaches. Ding (2017) used 3D Faster R-CNN for nodule detection on axial slices to reduce false positive (FP) results of lung cancer diagnosis. It was used with deep CNN architecture, the dual-path network (DPN), to learn the features of the nodules for classification. Hamidian (2017) propose a 3D CNN for automatic detection of pulmonary nodules in CT scans. This network is converted into a Fully Convolutional Network (FCN) which can generate a score map for the entire volume efficiently in a single pass, avoiding the sliding window approach and the greater computational cost associated with it. The FCN approach led to an 800-time speedup compared to the sliding window. The overall pipeline consists of the FCN for a fast candidate generation, which is followed by a CNN for the classification task.
After an extensive analysis of the training strategies in this research field, it has become apparent that they are focused on nodules greater than 3mm in diameter and use nodules whose annotation was confirmed by at least 2 or 3 radiologists. However, in the case of an individual radiologist's examination of a CT scan, nodules of sizes less than 6 mm are usually missed (Nasrullah 2019). CT scan analysis techniques are facing a lot of false positive results in the early stage of a lung cancer diagnosis when the nodule size is small. Therefore, the development of a different approach is needed for early-stage lung cancer detection, which is the research gap that will be addressed in this work.

3. Experimental Setup

3.1 Dataset Description

For the last two decades, 7 datasets were developed and used for classification and localization of lung nodules. In an analysis of a clinical CT database for nodule detection Riquelme and Akhloufi (2020) reveal that the lung nodules' detection CAD development relies mainly on "The Lung Image Database Consortium" (Armato 2011) dataset that consists of 1018 cases gathered from a collaboration of seven academic centers and eight medical imaging companies. The second most broadly used dataset is the

LUNA16 dataset. It is a subset of the LIDC-IDRI dataset, in which the heterogeneous scans are filtered by different criteria such as slice thickness.
On November 20, 2019 a "Grand Challenge on automatic lung cancer patient management" (INESCTEC) was released together with a new dataset – LNDb (Lung Nodule Database on Computed Tomography) (Pedrosa 2019). The challenge was organized by INESC TEC, Porto, Portugal in collaboration with the São João Hospital Centre, Porto, Portugal, and the Faculty of Engineering of Universidade do Porto and the Faculty of Medicine of Universidade do Porto. The new database has been created to complement current databases for the development and testing of lung nodules CAD strategies by giving additional focus to radiologist variability and local clinical reality. In the LNDb database, the variability in the annotation process was particularly emphasized as annotations were conducted solely in a single-blinded manner, which replicates more closely the clinical reality where images are analyzed by a single radiologist. Furthermore, in contrast to LIDC-IDRI, where radiologists were instructed to focus on nodules greater than 3mm in diameter, the main task for radiologists in LNDb was to find all nodules, independent of size. This has led to a more diverse dataset, composed of a higher proportion of nodules < 3mm than on LIDC-IDRI. Creating a CAD system that can detect small nodules is crucial for having a higher survival rate. Therefore, I have decided to work with the LNDb dataset, which is richer in small nodules when compared to the other publicly available datasets.
The LNDb dataset contains 294 CT scans collected at the Centro Hospitalar e Universitário de São João (CHUSJ) in Porto, Portugal between 2016 and 2018. Each CT scan was read by at least one radiologist at CHUSJ to identify lung nodules and other suspicious lesions. A total of 5 radiologists with at least 4 years of experience reading up to 30 CTs per week have participated in the annotation process throughout the project. Annotations were performed in a single-blinded fashion, i.e. a radiologist would read the scan once and no consensus or review between the radiologists was made.

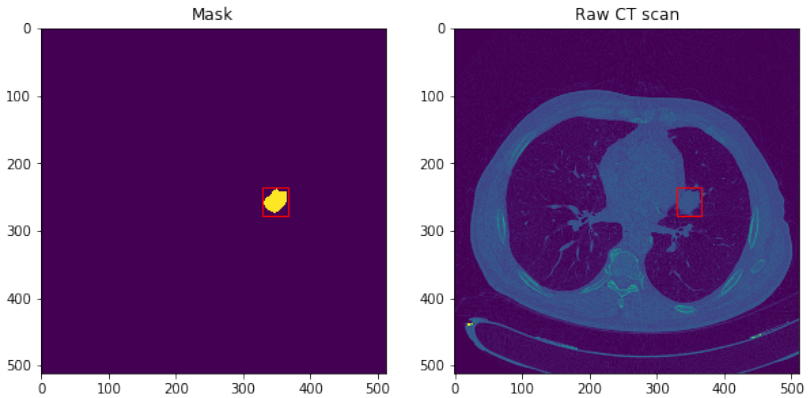


Figure 2: Nodule mask(left) and raw CT scan(right)

CT data and nodule masks (Figure 2), are available on MetaImage (.mhd/.raw) format, a file format standard in medical imaging. Individual nodule annotations are available on a .csv file that contains one finding (nodule) marked by a radiologist per line. The structure of the file can be seen in Table 1, where each line holds the LNDb CT ID, the radiologist that marked the finding (numbered from 1 to n radiologists within each CT), the finding's ID (numbered from 1 to n findings within each CT for each

radiologist), the xyz coordinates of the finding in world coordinates, whether it is a nodule (1) or a non-nodule (0), the corresponding nodule volume and the nodule texture rating is given (1-5) (INESCTEC).

LNDbID	RadID	FindingID	x	y	z	Nodule	Volume	Texture
1	1	1	-44.61	-119.07	-37.5	1	440.91	5
1	1	2	25.85	-126.97	-45.5	1	152.38	4
1	2	1	-44.00	-118.47	-37.5	1	56.82	5
1	3	1	-44.00	-119.68	-37.5	1	169.35	5
2	1	1	88.90	-123.63	-129.5	1	339.19	5
2	1	2	63.53	-112.76	-117.5	1	163.29	5
2	1	3	-103.85	-117.10	-253.5	1	357.04	5

Table 1: Individual nodule annotations

From the 294 CT scans in the LNDb dataset, 58 CTs have been withheld as a test set by the Grand Challenge, as well as the corresponding annotations, therefore, publicly available were the remaining 236 CT scans. All 236 CTs were annotated by at least one radiologist – 59 were annotated by a single radiologist, 110 by 2 radiologists, and 67 by 3 radiologists. Based on radiologist coordinate (xyz) annotations of the nodules' centroid and more particularly the z dimension which corresponds to the slice where the centroid of the nodule is located in the whole CT scan, I identified 572 images annotated by two or more than two radiologists. Out of the 572 images, 486 were containing nodules (positive images), and 86 were not (negative images).
The minimum nodule size in diameter is 0.57 mm, the maximum size is 30.7 mm, while the average size is 6.4 mm. Figure 3 (left) shows that the most frequent nodule size, among the confirmed by at least two radiologists nodules, is about 5 mm. The maximum number of nodules for one patient's CT scan, confirmed by at least two radiologists is 7. However, in most CT scans, the number of confirmed nodules is 1 or 2, as shown in Figure 3 (right).

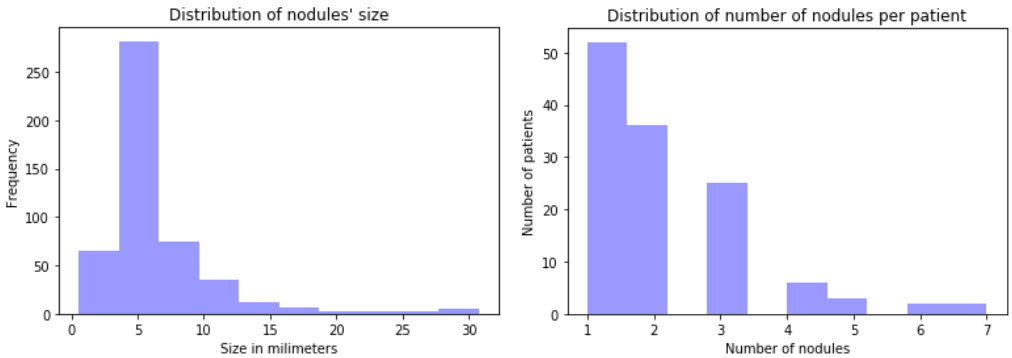


Figure 3: Nodule size distribution (left), Number of nodules per patient (right)

The median number of image slices in a patient's 3D chest CT scan in the LNDb dataset is 318, while the minimum and maximum number of image slices are 251 and 631 respectively. Of all the image slides, the images of interest are the extracted slides that contain the nodules' centroids. 91% of the images of interest hold only one nodule

- the one whose centroid is on that slice, according to the radiologist's annotations. Whereas 9% of the images of interest are holding not only the nodule whose centroid is on that image but also other nodules are visible (subnodules), even though their centroid is on another slice of the 3D scan. For example, in Figure 4, the presented slice (image of interest) is containing the centroid of the nodule in the red bounding box, but we can see that 2 more nodules are appearing, whose centroids are on another slices.

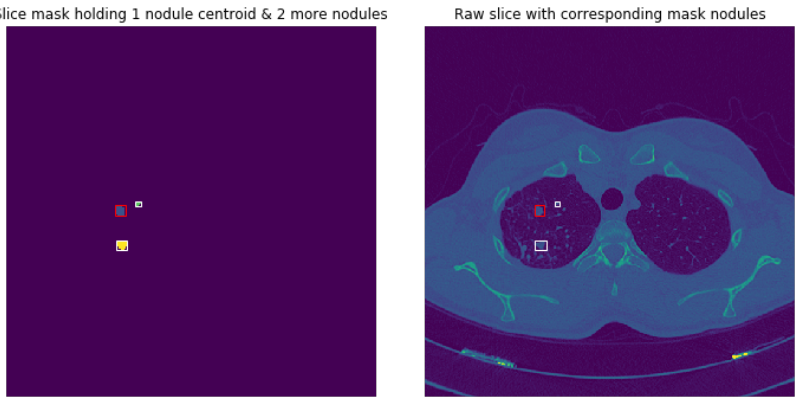


Figure 4: CT scan with nodules and their corresponding masks

3.2 Pre-processing

From the 3D CT scans which contain hundreds of slices (the whole human chest), only the image slices that hold a nodule's centroid (image of interest) are extracted and pre-processed. Furthermore, to accommodate the three-dimensional information in the CT image, the preceding and the succeeding images (neighboring) of the image of interest were used. The consecutive images were included, i.e. preceding image, the image of interest, and succeeding image in the red, green, and blue (RGB) channels of the input data. This was done to help the network to learn the depth and neighborhood image information during training.
Before applying the different pre-processing pipelines, all images were resized to width and height of (512 x 512), by using the pixel area relation interpolation method (Bradski 2000). As a next step, the three pre-processing pipelines (explained below) were applied to the three slices – preceding, the image of interest, and the succeeding image for each nodule. Then the three images were stacked together to form the RGB channel. For example, if patient 1 with CT scan 1 has 2 nodules on two different slices then those two slices and their neighboring slices were pre-processed and saved as two 3 channel jpg images which were later on fed into the models.

3.2.1 Pipeline 1 - Normalization and Zero Centering. Based on the reference HU value for lungs (-700 HU) (Kazerooni and H. 2004), the CT scans were normalized with window width [-1000, 400 HU] to highlight the lungs. The histograms of that transformation are shown in Figure 5, while the image transformation of Pipeline 1 can be seen in Figure 6. Having in mind the reference HU values and the histogram, we can easily see which pixels are air and which are tissue (or bone) on the image and also how important it is to apply a pre-processing technique to improve the image contrast.

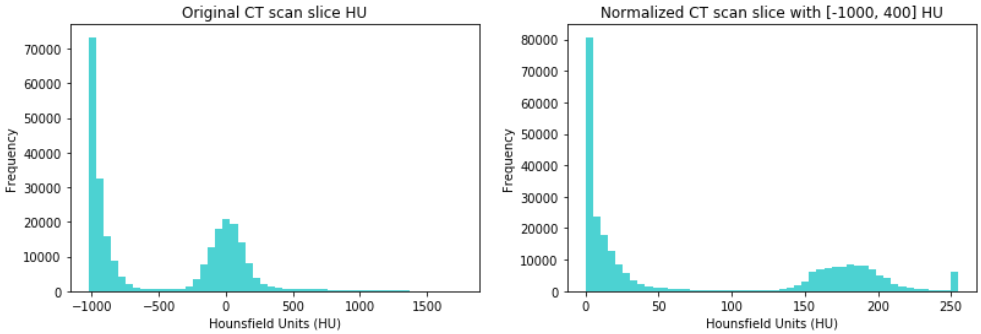


Figure 5: CT scan slice histogram before and after pre-processing with a certain window width of HU.

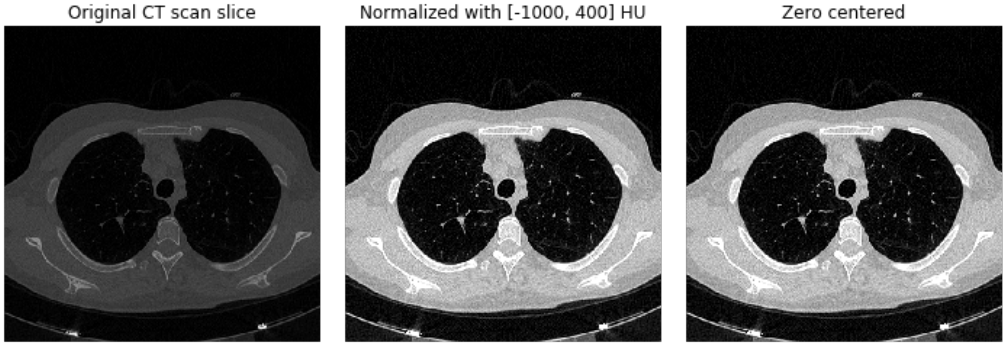


Figure 6: Pipeline 1 transformation

To improve the learning process by reducing the effect of scale differences and only remain the content of the image, as part of pre-processing Pipeline 1, the images were zero-centered by subtracting the mean pixel value (0.25) from all pixels. To zero center, simply means to "shift" the values of the distribution so that its mean is equal to 0. To determine the mean pixel value I averaged all images in the whole LNDb dataset.

3.2.2 Pipeline 2 - Normalization and Logarithmic Transformation. In Pipeline 2 the contrast enhancement of the CT images is achieved through logarithmic transformation. Logarithmic transformation is most often used to brighten the lower intensity values of an image. This transformation is widely used for enhancing dark images. In the case of CT images, the details in the lower intensity regions are not normally seen on the non-contrast CT images. Most of the abnormalities lie in the darker regions as well. Hence, logarithmic transformation can be used for the enhancement of darker regions of the CT images to get the details in the lower intensity regions. The log transformation was implemented for each pixel using the formula:

$$s = c * log(1 + r) \tag{1}$$

where s is the output image, r is the intensity of the input medical image, and c is a constant and the value depends upon the limit of the greyscale window used. The parameter c is used to scale the range of the log function for matching the input domain. For uint8 images $c = 255/log(1 + 255)$ (Jinimole 2017).
Before applying the log transformation the images were normalized. After experimenting with different window width values and log bases - log2, log10, and natural log, the normalization with window width [-250, 1500 HU] and natural log transformation gave the best results. The result is presented in Figure 7.

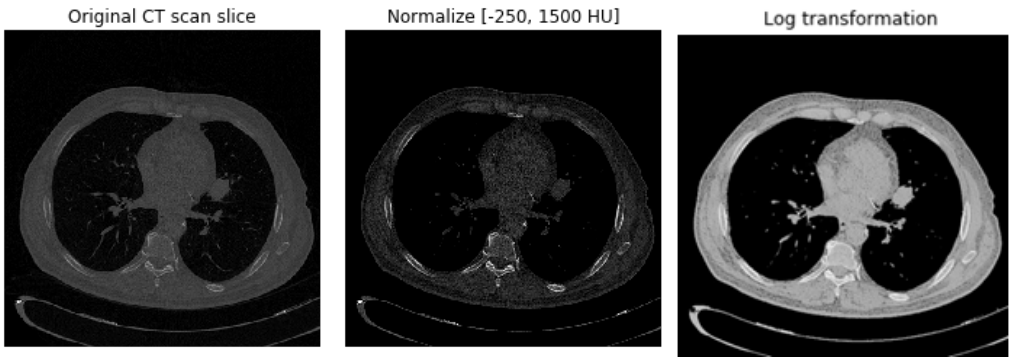


Figure 7: Pipeline 2 transformation

3.2.3 Pipeline 3 - Normalization and Median Filtering. In Pipeline 3 a non-linear filter was applied – median filter. The principle of the median filter is to replace the gray level of each pixel by the median of the gray levels in a neighborhood of the pixels. It is very similar to smoothing and easy to implement. For example, in Figure 8 in a solid bordered rectangle are the pixels of the corrupted image. The 3 × 3 kernel (dashed rectangle) requires zero paddings 3/2 = 1 column of zeros at the left and right edges while 3/2 = 1 row of zeros at the upper and bottom edges.

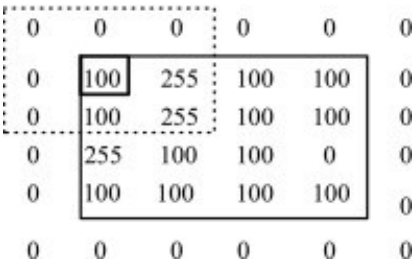


Figure 8: Image padded with zeros

To process the first element, we cover the 3 × 3 kernel with the center pointing to the first element to be processed. The sorted data within the kernel is listed in terms of its value. The median value = median (0, 0, 0, 0, 0, 100, 100, 255, 255) = 0, therefore, zero will replace 100. This method continues for each element until the last is replaced (Tan 2019). Before applying the median filter, I normalized the image with [-1000, 400 HU]. The result of pipeline 3 is shown in Figure 9.

In order to more clearly see the transformation of the different pre-processing pipelines, a comparison of the three approaches with the CT scan image of two patients are shown on Figure 10 and Figure 11. In both figures we can see how applying normalization helps to improve the contrast of the original image which, as we saw earlier, appears dark and with relatively low contrast. The log transformation further improves the contrast and the lung area appears less noisy vs. the zero centering and median

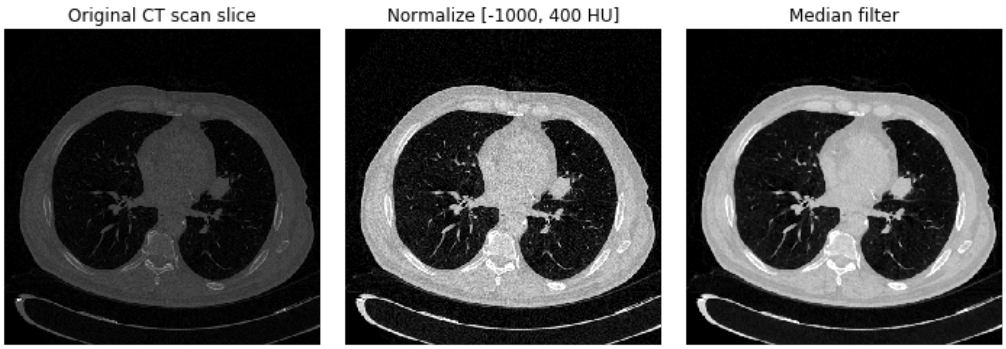


Figure 9: Pipeline 3 transformation

filtering pre-processing. However, removing the noise with the log transformation may erase some useful information especially when the nodules' texture is not solid. The 3rd pipeline - normalization with median filtering produces an image which is blurrier in the lung area when compared to the other two pipelines but keeps some of the information that gets removed by the log transformation pipeline.

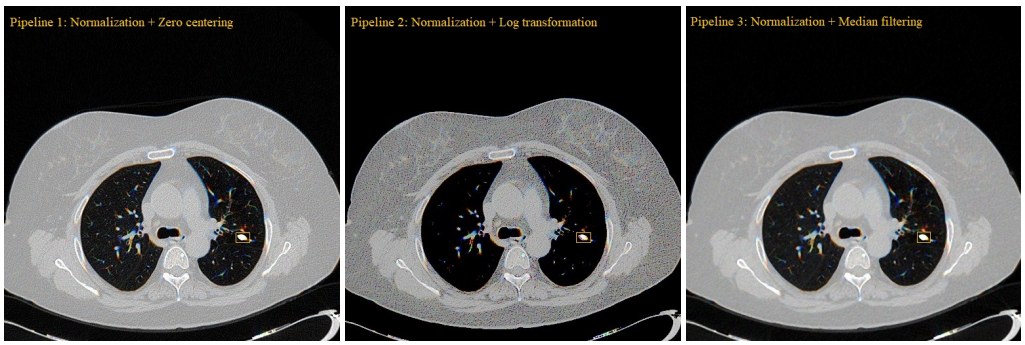


Figure 10: Pipeline transformations on LNDb-11 CT scan

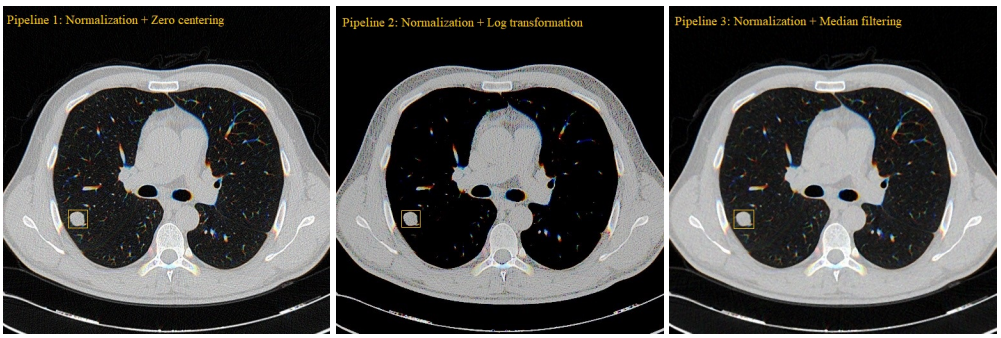


Figure 11: Pipeline transformations on LNDb-22 CT scan

3.3 Label Data Preparation

The input for YOLO algorithm were the 2D CT images of interest in jpg format, which were derived from the pre-processing techniques. For each .jpg-image-file a corresponding label (.txt-file) was created with the same name and the following information:

<object-class> <x-center> <y-center> <width> <height>, $where$

- **<object–class>** is an integer object number from 0 to the number of classes you want to detect - 1. This project aims to detect only one object (nodule), therefore, <object class> = 0;
- **<x-center> <y-center>** are coordinates of the bounding box's center;
- **<width> <height>** are the width and height of the bounding box.

An example is shown in Figure 12 with 1 nodule in the red bounding box, where the centroid of the nodule is located at (x,y) = (348, 256) and the width and height of the bounding box are respectively 34 and 36.

YOLO requires <x-center> <y-center> <width> <height> to be float values relative to the width and height of image (this is important because of the different online data augmentation methods). All images are with size (512, 512). Therefore, to make the coordinates compatible with the label representation of YOLO for that example we have 0.6797 = 348/512; 0.4991 = 256/512 and so on until we reach the following representation:

<object-class>	**<x-center>**	**<y-center>**	**<width>**	**<height>**
0	**0.6797**	**0.4991**	**0.0664**	**0.0703**

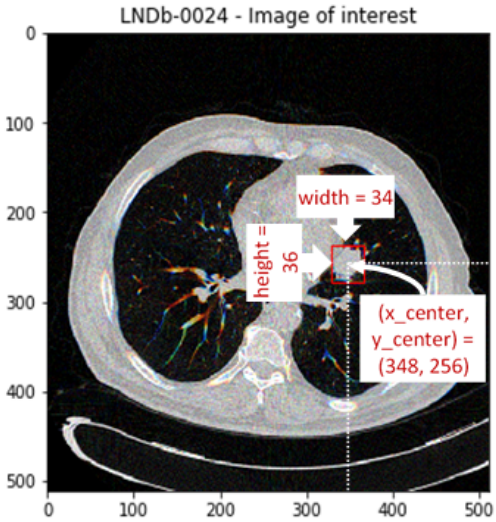


Figure 12: YOLO compatible label file

3.4 Deep learning architectures

The YOLO framework uses a custom network based on the GoogleNet architecture (Szegedy 2015) to detect the objects in an image. Features from the entire image are used by the neural network to predict each bounding box and class probabilities. The network learns about the full image as well as all the objects in the image. The input image is divided into a regular grid as shown in Figure 13. Each grid cell has a label associated with it which contains the probability the class is present in the grid cell, the pixel coordinates defining the bounding box of that object relative to the center of the grid square and a coverage value of 0 or 1 to indicate whether an object is present within the grid square. The grid cell in which the center of an object falls is responsible for detecting that object. In order to establish a fixed size for the data representation, a 'don't care' class is assigned to a grid square where no object is present (Ramachandran 2018).

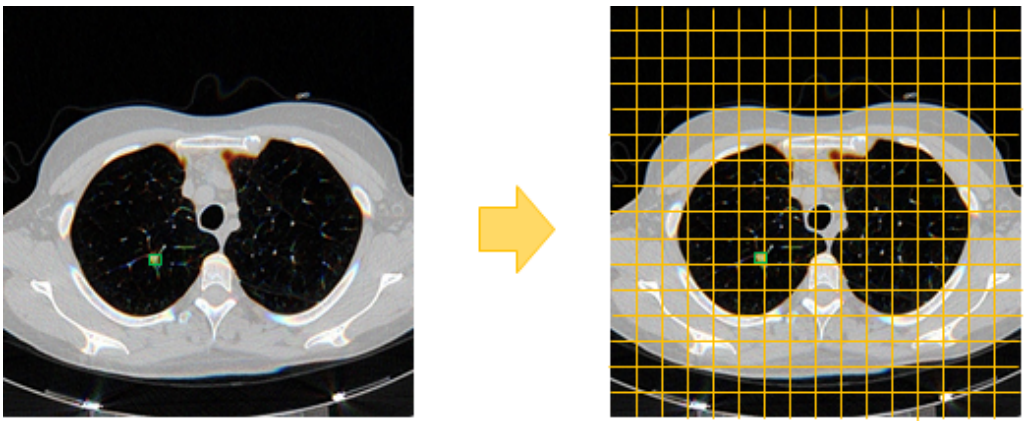


Figure 13: Data Representation for lung CT image using YOLO architecture

In this work, there is only one class to be detected – the nodule. Therefore, there is a total of 1 class in which the objects can be classified into. So, for each grid cell, the label y will be a six-dimensional vector with the following information:

$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_w \\ b_h \\ c \end{bmatrix}$$

- p_c defines whether an object is present in the grid or not (it is the probability);
- b_x, b_y, b_w, b_h specify the bounding box if there is an object in that grid cell, where b_x, b_y are the x and y coordinates of the midpoint of the object for this grid and b_w, b_h respectively are the ratio of the width and height of the bounding box (green box in the example above) to the height of the corresponding grid cell;
- c is a coverage value of 0 or 1 to indicate whether an object is present within the grid square.

In the example, there is only one grid cell that contains an object in it (nodule), therefore, this grid cell p_c will be equal to 1. b_x, b_y, b_w and b_h will be calculated relative to the particular grid cell we are dealing with. c will be 1 (in case we were interested in detecting 2 classes, the y label vector would have c_1 and c_2 instead of just c. When the grid cell is holding the object corresponding to c_1, the same will be 1 while c_2 will be equal to 0). For all other grid cells we assign a "don't care" label in which p_c will be equal to 0 as shown in Figure 14 (right). With input image – 512 x 512 x 3, the output

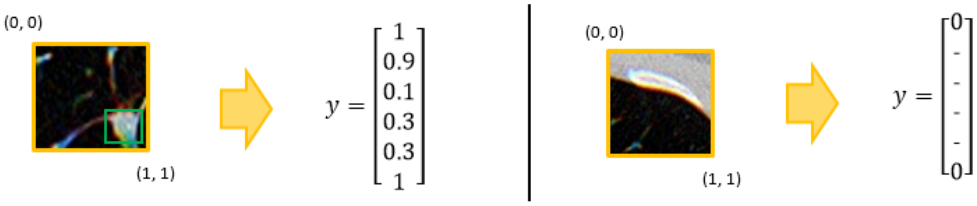


Figure 14: Grid cell with nodule (left) and without nodule (right) with corresponding label vectors

will have a shape of 16 x 16 x 6 meaning that for each of the 16 x 16 grids we will have a six-dimensional output vector.
YOLO is a convolutional neural network. The total number of layers in YOLO v3 is 107. There are 75 convolutional layers, with skip connections and upsampling layers. No pooling is used, and a convolutional layer with stride 2 is used to downsample the feature maps. YOLO v4 is also a fully convolutional network. It has more layers than YOLO v3 – 162, out of which 110 are convolutional. In YOLO v4 there are 3 max layers that are not present in version 3.

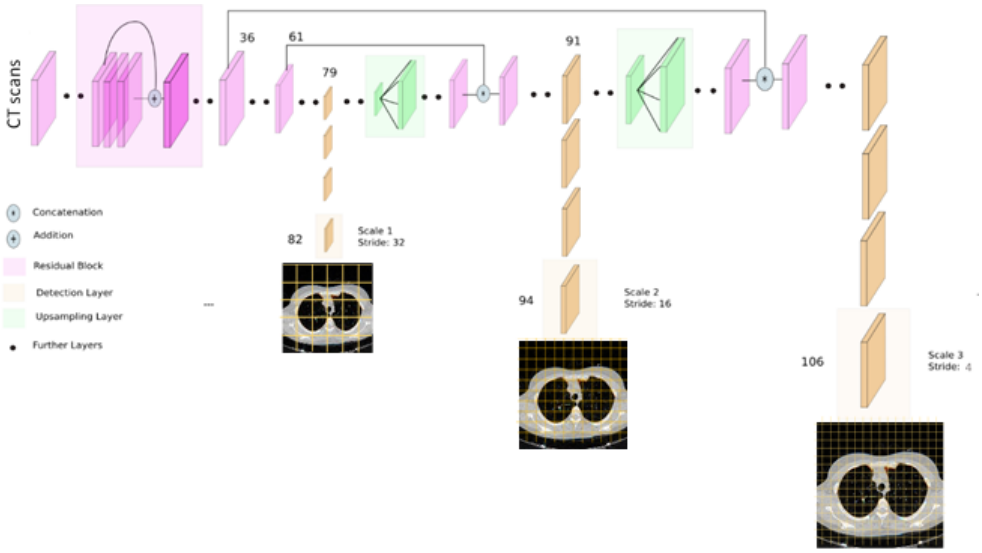


Figure 15: Architecture

YOLO makes detections at three different scales. Each scale is responsible for detecting objects of different sizes. The detection is done by applying 1 x 1 detection kernels on feature maps of three different sizes at three different places in the network which are given by downsampling the dimensions of the input image by 32, 16, and 4 respectively (see Figure 15 above).
An output of the algorithm at each of the three scales are coordinates of the detected object and a float between 0 and 1, corresponding to the predicted probability of the finding being a nodule. That information will be compared with the ground truth annotations. YOLO predicts the bounding boxes using anchor boxes. Anchor boxes are

chosen by applying k-means clustering to choose reasonable height/width ratios that represent the different sizes of the object of interest. In this work 11 anchor boxes were used to detect nodules with different sizes.

As shown on Table 2, 1 anchor box with size (60, 60) is placed on the first scale, which is responsible for detecting the biggest objects. Then, 1 anchor box with size (30, 30) is placed at the second scale, which is responsible for detecting the medium size objects. Finally, 9 anchor boxes with size (1, 1), (4, 4), (5, 7), (6, 7), (9, 7), (8, 10), (12, 11), (16, 17), (30, 26) are placed at the third scale, which is responsible for detecting the smallest objects.

	# Anchors	*Anchor size*	*Downsample factor*	*Feature map*
Scale 1	1	(60, 60)	32	16 x 16 x 6
Scale 2	1	(30, 30)	16	32 x 32 x 6
Scale 3	9	(1, 1), (4, 4), (5, 7), (6, 7), (9, 7) (8, 10), (12, 11), (16, 17), (30, 26)	4	128 x 128 x 54

Table 2: Three scales for detection

The shape of the detection kernel at the three different scales is 1 x 1 x (B x (5 + C)). Here B is the number of bounding boxes a cell on the feature map can predict (number of anchor boxes), "5" is for the 4 bounding box coordinates and 1 object probability value, and C is the number of classes. In this work, the shape of the detection kernel is 6 (1 x 1 x (1 x (5 + 1)) for the first two prediction layers (scale 1 and scale 2). For scale 3 the detection kernel is 54 (1 x 1 x (9 x (5 + 1)), because of the 9 anchor boxes.
Furthermore, the first detection (at scale 1) is made by the 82nd layer. For the first 81 layers, the image is downsampled by the network, such that the 81st layer has a stride of 32. For an image of size 512 x 512, the resultant feature map would be of size 16 x 16. One detection is made here using the 1 x 1 detection kernel, giving us a detection feature map of 16 x 16 x 6. Then, the feature map from layer 79 is subjected to a few convolutional layers before being upsampled by 2x to dimensions of 32 x 32. This feature map is then depth concatenated with the feature map from layer 61. Then the combined feature maps are again subjected a few 1 x 1 convolutional layers to fuse the features from the earlier layer (61). Then, the second detection is made by the 94th layer, yielding a detection feature map of 32 x 32 x 6. A similar procedure is followed again, where the feature map from layer 91 is subjected to few convolutional layers before being depth concatenated with a feature map from layer 36. Like before, a few 1 x 1 convolutional layers follow to fuse the information from the previous layer (36). The final of the 3 detections is made at 106th layer, yielding feature map of size 128 x 128 x 54 (Kathuria). A more detailed representation of the layers in the two architectures - YOLO v3 and v4 is given in the Appendix section.

3.5 Training

A group-wise (patient wise) split of the 486 images, with nodules confirmed by at least 2 radiologists, was performed to divide the dataset into 80% train and 20% test images and make sure that one patient is present either in the training dataset or in the test dataset. A perfect 80%/20% was not possible because of the group-wise shuffle split, therefore, the training dataset consist of 402 positive images, which is 82.7% of the

486 positive (nodule) images. The test dataset became naturally class balanced with 84 positive images and 86 negative images after splitting the data into train and test sets.
The input for training the model will be the images and their corresponding y labels. Online data augmentation is then applied to the images which consist of random scaling, cropping, flipping, rotating, and changing the exposure of the image in a certain range. In YOLO v4 additional data augmentation method is applied – a mosaic that mixes 4 training images.
The learning process was completed in 4000 epochs with a batch size of 64. The network is trained using stochastic gradient descent with Adam optimizer and a standard learning rate of 0.001 which drops to 0.0001 after 3200 epochs and to 0.00001 after epoch 3600. A momentum of 0.9 and a decay of 0.0005 were used.
Linear combination of two separate loss functions was used to produce the final loss function – one for the bounding box loss and one for the class probability loss. The bounding box loss is the Mean Square Error (MSE) loss which directly performs regression on the center point coordinates and the top left corner of the Bbox, using Formula 2. A binary cross-entropy with logistic activation was used for loss function for the class probability prediction, using Formula 3.

$$Bounding\ Box\ Loss = \frac{1}{2N}\sum_{i=1}^{N}((x_1^t - x_1^p) + (y_1^t - y_1^p) + (x_2^t - x_2^p) + (y_2^t - y_2^p))^2 \quad (2)$$

where $(x_1^t, y_1^t, x_2^t, y_2^t)$ are the co-ordinates of the ground-truth bounding box and $(x_1^p, y_1^p, x_2^p, y_2^p)$ are the coordinates of the predicted bounding box. N is the batch size. Minimizing the weighted sum of these loss values is the training objective.

$$Binary\ Cross - Entropy = -(y * log(p) + (1 - y) * log(1 - p)) \quad (3)$$

where log is the natural logarithm, y is a binary indicator (0 or 1) if class label c is the correct classification for certain observation and p is the predicted probability the observation is of class c.

3.6 Methods

After obtaining all input images and their corresponding labels, eight experiments were conducted by training the 2 model architectures – YOLO v3 and YOLO v4 with the 3 pre-processing pipelines' images and the original images. The experiments were conducted by applying transfer learning to the models. I decided to experiment with transfer learning for nodule detection because of the following reasons. The popular deep convolutional network architectures like GoogLeNet (Szegedy 2015) and AlexNet (Krizhevsky and Hinton 2012) contain millions of parameters to train. These networks have been trained on databases like ImageNet which contains millions of images. YOLO architecture is based on GoogLeNet and therefore should give a better performance if trained on sufficiently huge numbers of labeled CT images. But obtaining such a large number of CT images and getting them annotated by radiologists is a very challenging task. An effective option is to initialize the weights of the new network during training, with the weights obtained from other networks comprehensively trained on the large scale well-annotated datasets like ImageNet (Ramachandran 2018). As a result, medical

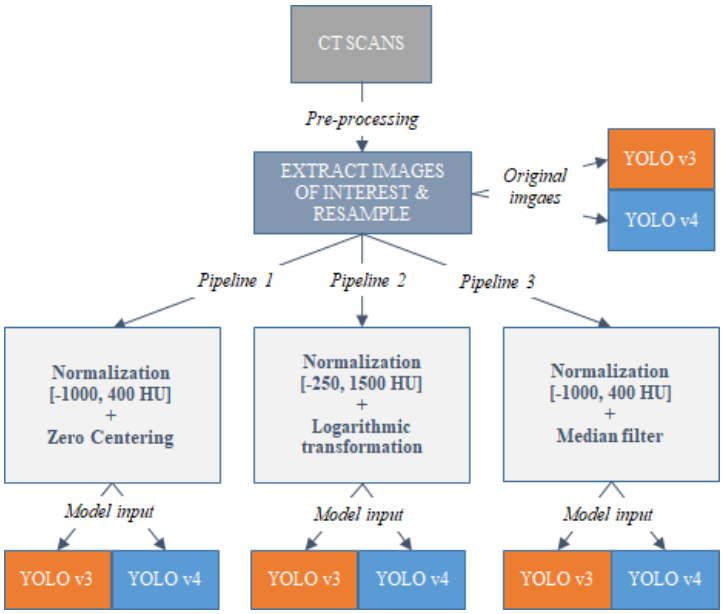


Figure 16: Project flowchart

image recognition tasks with limited amount of data can achieve higher accuracy with fewer computational requirements.

3.7 Evaluation metrics

For each grid cell there will be as many predicted bounding boxes as the number of anchor boxes. For example, the output of the last layer as calculated earlier will be 128 x 128 x 54, meaning that there are 54 predicted bounding boxes in each grid cell. Two approaches help us evaluate which of the many predicted bounding boxes is closest to the ground truth – Intersection over Union (IoU) and Non-Maximum Suppression.

Intersection over Union (IoU) score is computed for each predicted bounding box and the ground truth bounding box. IoU is the ratio of the overlapping areas of two bounding boxes to the area of union as shown in Figure 17. An Intersection over Union score > 0.5 is normally considered a "good" prediction.

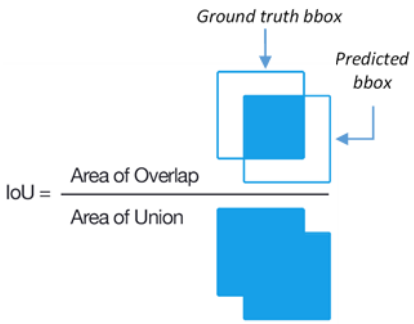


Figure 17: Computing the Intersection over Union by dividing the area of overlap between the bounding boxes by the area of union

One of the most common problems with object detection algorithms is that rather than detecting an object just once, they might detect it multiple times. There is one more technique that can improve the output of YOLO significantly – Non-Max Suppression. The Non-

Max Suppression technique leaves only a single detection per object by looking at the probabilities associated with each detection and taking the largest one.

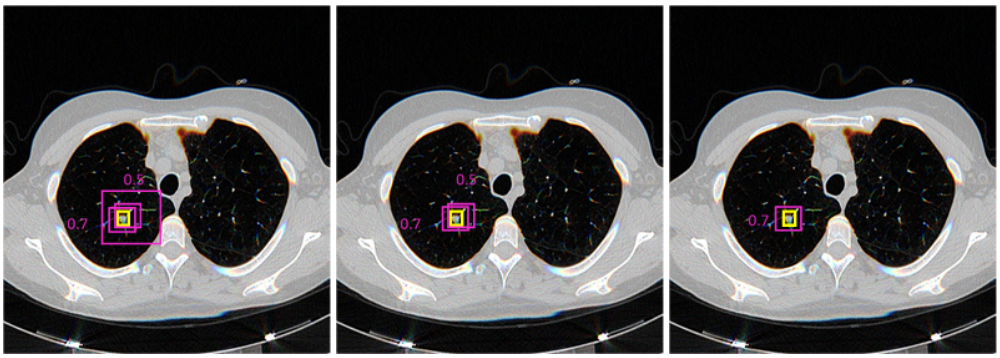


Figure 18: Pipeline transformations on LNDb-22 CT scan

Consider the following example in Figure 18. The ground truth is the yellow bounding box. There are 3 predicted bounding boxes. After applying the IoU technique, we will remove the predicted bounding box with the lowest IoU score – visually it is easy to see that this is the biggest square that gets removed in the second image. Now, we need to apply Non-max suppression rule, because we still have two predicted bounding boxes with the same IoU score. One of them has a probability score of 0.5 while the other one has 0.7 which is the one that we leave according to the Non-Max Suppression rule. In third image we can see the single prediction corresponding to the ground truth nodule.

When analyzing the results of the different methods, the most important evaluation metrics are the sensitivity and the number of false positives. The goal is to achieve high sensitivity, i.e. detecting as many of the nodules present in a scan while keeping low number of false positives, i.e. incorrectly detecting a spot as a nodule, when it is not. False Positive happens when the algorithm determines the existence of the disease when there is no disease which is highly undesirable scenario. There should be a trade-off between high sensitivity and high number of false positives, this is the reason why, the performance of the CAD systems is evaluated based on sensitivity and number of false positives. The ability of the model to correctly identify the nodules is measured by the sensitivity, using Formula 4.

$$Sensitivity = \frac{TruePositive}{(TruePositive + FalseNegative)} \tag{4}$$

Except for narrowing down the number of predicted boxes as discussed earlier in this section, the Intersection over Union score of the detected bounding box is also the parameter used for setting the threshold to classify the findings as true positive and false positive. IoU is widely used to evaluate custom object detectors, because, it is extremely unlikely that the (x,y) - coordinates of a predicted bounding box are going to exactly match the (x,y) - coordinates of a ground-truth bounding box due to the many parameters of the model. In this work, in order to designate a predicted bounding box as true positive it must pair with the ground truth bounding box, such that their IoU exceeds 0.5. In addition, in the results chapter, the sensitivity and the total number of

false positives will be calculated for two more IoU thresholds (0.3 and 0.7) to show the effect which the IoU score has on the model's performance.

The total number of false positives divided by the total number of test images (170) gives the false positive rate (false positives per image) which will also be shown in the results chapter.

4. Results

This chapter provides an overview of the obtained results from the eight experiments which were evaluated on the test set of 170 images (84 positive and 86 negative). The best performing model was selected based on the sensitivity score and the number of false positives. Among the eight experiments, YOLO v4 with pipeline 3 (normalization and median filtering) achieved the highest sensitivity of 76% and the second lowest number of FPs when compared to the other YOLO v4 models, i.e. 59 FPs or 0.35 false positives per image. From Table 3 we can see that YOLO v4 is performing better than YOLO v3 in terms of sensitivity, while v3, in general, has a lower number of false positives.

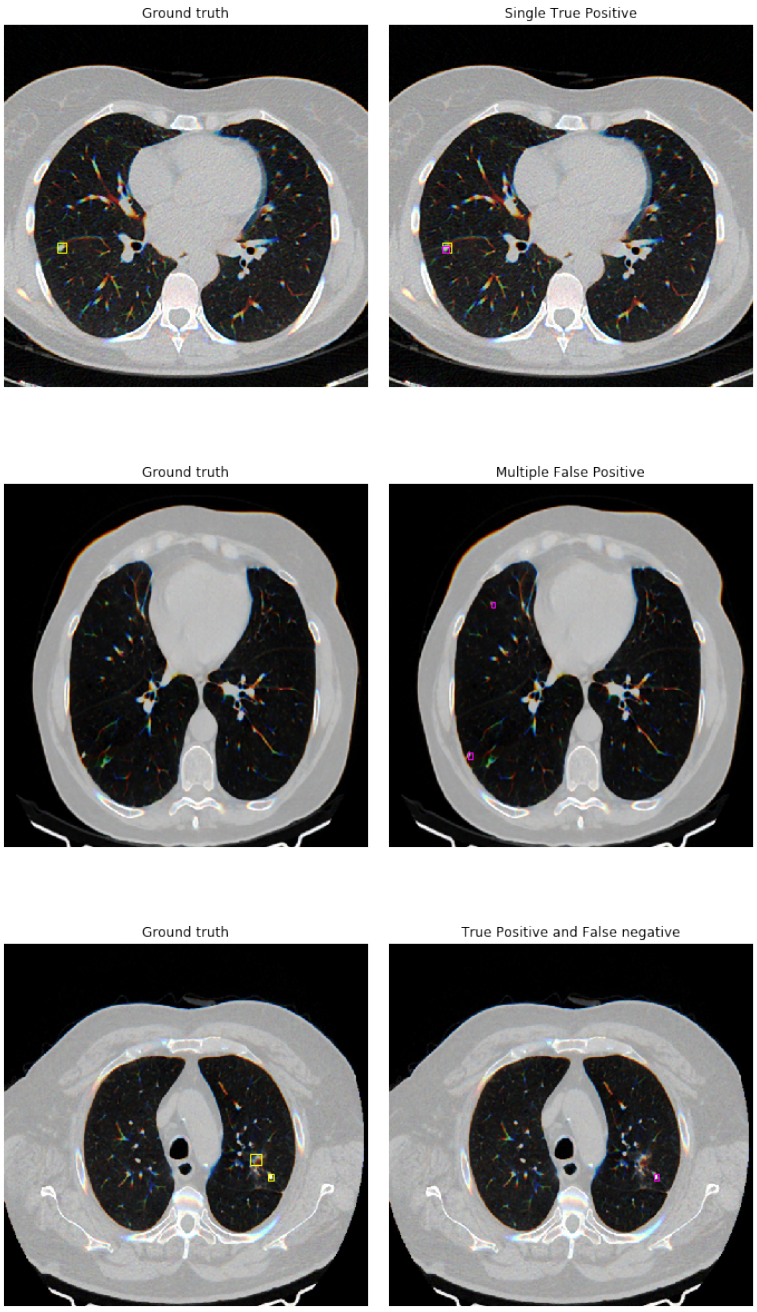
In both architectures the model generalization capability is lower when the model is trained on the original images vs. when trained on the pre-processed images. The ratio of images with no nodule (negative) where one was detected (FP) is given in Table 3 as well as the false positive rate (false positive per image). Even though the cost of a false positive in the same image where there is also a true positive is lower than the cost of a false positive where there is no nodule at all, both FP ratios are low and give satisfactory results.

The results in Table 3 are calculated based on The Intersection over Union (IoU) score of 0.5.

Experiments		IoU score 0.5					
Architecture	***Pipeline***	***Sensitivity***	***TP***	***FP***	***FN***	***FP / all test images***	***FP / negative test images***
YOLO v4	original img	0.65	55	55	30	0.32	0.64
	pipeline 1	0.66	56	64	29	0.38	0.74
	pipeline 2	0.68	58	69	27	0.41	0.80
	pipeline 3	**0.76**	65	59	20	**0.35**	0.69
YOLO v3	original img	0.56	49	50	39	0.29	0.58
	pipeline 1	0.65	55	49	30	0.29	0.57
	pipeline 2	0.61	52	33	33	0.19	0.38
	pipeline 3	0.64	54	57	31	0.34	0.66

Table 3: Results of the conducted experiments

The detection results along with the bounding boxes from the proposed detection system (YOLO v4 pipeline 3) are shown in Figure 19.



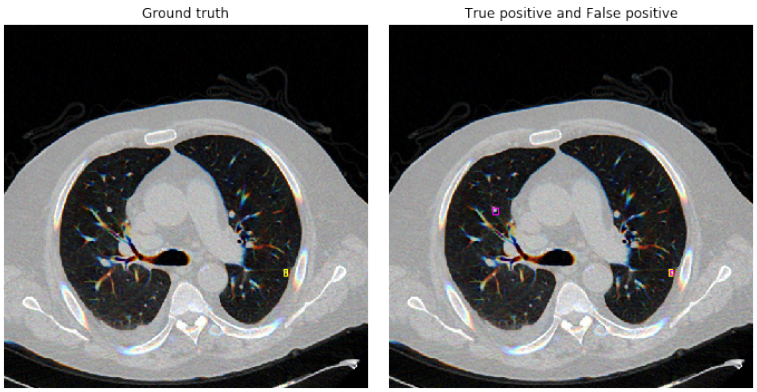


Figure 19: Results of the proposed CAD system

The chart in Figure 20 gives information on how the sensitivity and number of false positives get affected by a change in the IoU metric. For example, if we consider that a predicted bounding box is a true positive when the IoU with the ground truth bounding box exceeds 0.3, the sensitivity for YOLO v4 pipeline 3 increases to 85% vs 76% for IoU score of 0.5. On the other hand, if we are interested in IoU higher than 0.7 the sensitivity significantly drops to 33%, and the number of false positives increases to 96 vs. 59 when IoU is 0.5.
The chart helps us also to visualize the results from Table 3 at the point where the x-axis equals 0.5. It is clear that YOLO v3 pipeline 2 achieves the least number of false positives, but it also has the lowest sensitivity. Since both metrics are important, there should be a trade-off between good sensitivity results and high FPs. From the conducted experiments YOLO v4 with pipeline 3 is the one that achieves that balance.

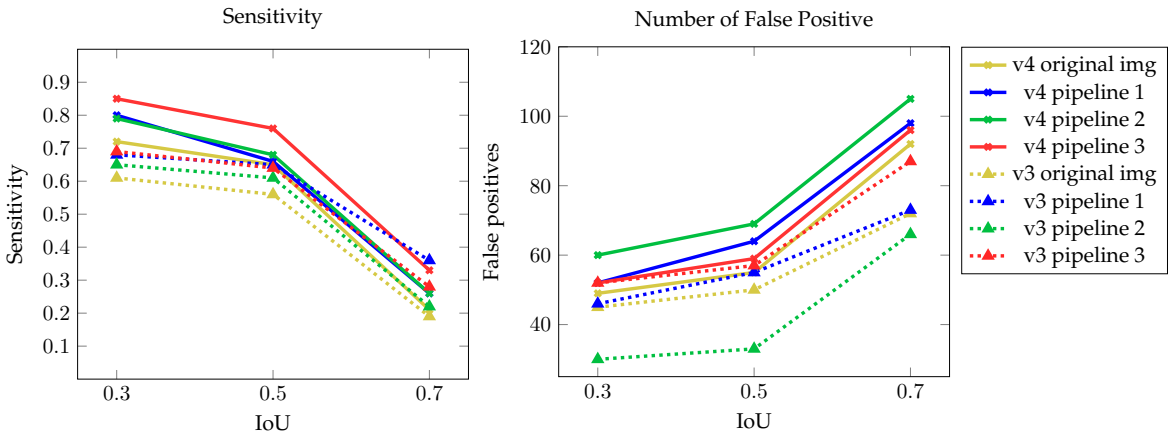


Figure 20: Summary on Sensitivity and FPs at different IoU scores

The example in Figure 21 shows how the prediction looks like for the different IoU scores, where the ground truth bounding box is yellow and the predicted bounding box is purple. IoU of 0.3 gives a high sensitivity, but as we can see in the first image, the localization is relatively poor. IoU of 0.5 is considered to be a good prediction.

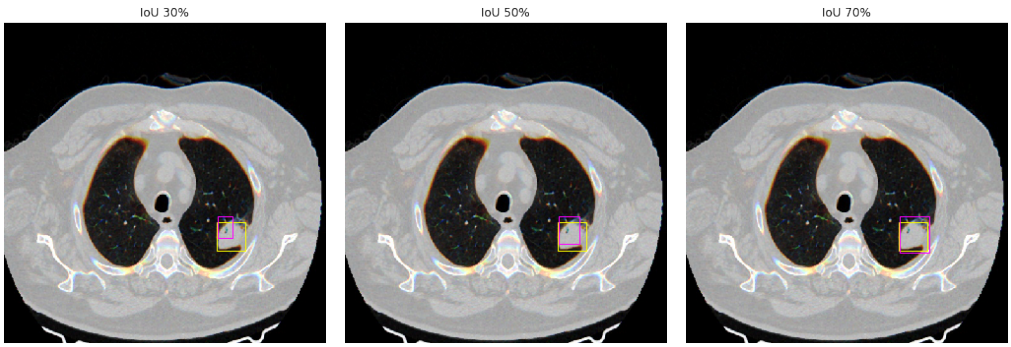


Figure 21: Purple bounding box is the prediction box and the yellow one is the ground truth bounding box

Since the LNDb dataset significantly differs in terms of size and variation from the datasets used for the lung nodules detection approaches published in the recent years, it would be misleading to make comparison of the results achieved by the proposed method and those of other detection systems. The fact that I did not exclude the smallest nodules (<3mm) leads to a decrease in the sensitivity metric which was expected but also confirmed by the post-processing analysis which showed that the average diameter of the false negative nodules is 3 mm.

5. Discussion

5.1 Answering research questions

The research question this thesis aimed to answer is **"To what extent can YOLO models automatically detect lung nodules from CT scans"** with the following two sub research questions:

1. How do YOLO algorithms trained on images pre-processed with various image enhancement techniques compare with training on the original images?
2. Which YOLO architecture and image enhancement technique achieve fewer false positive detections?

To answer these research questions, two model architectures (YOLO v3 and v4) were used. In order to prove that interactions may exist between image enhancement techniques and YOLO algorithms, both models were trained on the original images and on images pre-processed with 3 different image enhancement techniques. The image enhancement pipelines include: 1) normalization and zero centering, 2) normalization and logarithmic transformation, 3) normalization and median filtering. This approach

resulted in 8 models that were compared to answer the research question and the two sub research questions.

The sensitivity score of 76%, achieved with YOLO v4 architecture and pipeline 3 (normalization and median filtering), proves that lung nodules can be automatically detected by learning the discriminative features of the nodules and predict the bounding box coordinates by using only the images and no handcrafted features like shape, size or texture. Furthermore, when we compare the results of the models trained on pre-processed images with the results achieved by the models trained on the original images, we saw that the pre-processing of the raw CT scans had a positive impact on the overall performance of the model in terms of sensitivity and number of false positives. The sensitivity of YOLO v4 trained on original images was 65% while the same model trained on the pre-processed images generated sensitivity of 66%, 68% and 76%, respectively for pipeline 1, 2 and 3. The second sub research question required the comparison of the two architectures – YOLO v3 and v4 to distinguish the one achieving fewer false positives. And we saw that with all of the three pre-processing pipelines YOLO v3 had fewer FPs than YOLO v4.

To verify the effectiveness of the proposed method, its results were compared with the nodule detection results achieved by the teams which participated in the "Grand Challenge on automatic lung cancer patient management" (INESCTEC). They were working with the same dataset and also did not filter the nodules by their size which is one of the challenge's requirements. The best performing team reached a sensitivity score of 67%. The proposed method in this work showed a sensitivity of 76% suggesting that the YOLO models could have a promising contribution to the further development of CAD systems for lung nodules detection and most importantly address the challenges of small nodules detection.

5.2 Limitations

While the results of this study are promising, there are important limitations to be considered. The 3D contextual information of the CT scans was not fully utilized since only image slices of the scan were used for training and not the whole CT scan. On the other hand, in order for YOLO to be able to successfully predict a certain nodule from the test set, it requires a nodule similar in size and shape to be presented in the training dataset. This made the detection task more challenging since the dataset that was used is small and with high variability. Even though one of the dataset's strengths is its variability, as annotations were conducted solely in a single-blinded manner, which replicates more closely the clinical reality where images are analyzed by a single radiologist, this is also a limitation because there is no absolute ground truth. The drawback of the dataset variability revealed when working with the whole dataset versus the dataset limited to nodules independently confirmed by at least two radiologists. There was a huge difference in the models' generalization capabilities - the sensitivity that the proposed model (YOLO v4 pipeline 3) achieved with the full dataset was 40%, which improved to 76% only by including solely the nodules annotated by two or more radiologists.

I would like to pay a special attention to the computational power that is needed for training an object detection algorithm. Even with the minimum number of interactions (4000) needed to achieve a good result, a small dataset, and the usage of GPU, the training was taking days with many interruptions. Therefore, the need for good computer infrastructure should not be underestimated.

5.3 Future work

What makes the screening tests essential for the higher survival rate is the fact that lung cancer does not usually cause noticeable symptoms until it is spread through the lungs or into other parts of the body. However, the screening programs are directly impacting the workload of radiologists who need to analyze an increasing number of screening tests. This workload can result in errors in detection (failure to detect) or misinterpretation (inability to properly diagnose a tumor). Therefore, providing a reliable tool which can assist radiologists in the interpretation of CT scans can have a substantial impact for the society. This research contributed to the existing work by showing that the small nodules detection is an achievable task. Even though, the results of YOLO v4 and pipeline 3 (normalization and media filtering) were promising, in order to make the proposed method highly suited to be exploited in clinical practice, a higher sensitivity is required. A higher sensitivity will make the model more reliable when interpreting the CT scans and improve the diagnosis accuracy. The first step towards improvement of the proposed method, would be to ensure more computational power in order to train the model for more interactions. Furthermore, using the LNDb dataset as a complimentary to another publicly available dataset could give a better chance to account for the diversity of the nodule's size, shape, texture, and location and improve the proposed method's generalization ability. An experiment with a different view of the CT scans instead of working only with the axial plane might reveal an opportunity to further increase the dataset and improve the performance of the proposed method in future research.
YOLO is a very powerful tool. It is an instrument that can help us find solutions for the many challenges in the medical domain. Being able to implement a model trained on natural images into the medical field was an exciting and challenging task. I believe that the usage of YOLO and other deep learning models could have a positive impact not only for doctors and patients but for the society as well.

6. Conclusion

In this work, I propose an application of an object detection algorithm – YOLO, for detection of lung nodules from CT scan images. In the proposed work 2 different YOLO versions and 3 pre-processing pipelines were used to answer the question to what extent YOLO models can automatically detect lung nodules. This work shows that the presence of nodules can be detected along with bounding boxes with a sensitivity of 76%, using YOLO v4 with the pre-processing pipeline 3 which includes normalization and median filtering. The entire detection process takes place within the single neural network architecture which makes the method simpler as well as more effective, when compared to a multistage detection process. YOLO sees the entire image during the training and testing process and implicitly learns the contextual information about the classes and their appearance in the images. This gives the method an edge over the detection methods which are based on sliding window and region proposals. By incorporating transfer learning in CNN and by preserving the neighborhood image information during training, the proposed detection system exhibits promising sensitivity with a low number of false positives.

References

Anirudh, Thiagarajan J. J. et al., R. 2016. Lung nodule detection in ct using 3d convolutional neural networks trained on weakly labeled data. *SPIE Medical Imaging*.

Armato, Samuel G. 2011. The lung image database consortium (lidc) and image database resource initiative (idri): A completed reference database of lung nodules on ct scans. *Med Phys. 38(2): 915–931*.

Bochkovskiy, Chien-Yao Wang Hong-Yuan Mark Liao, A. 2020. Yolov4: Optimal speed and accuracy of object detection. Technical Report arXiv:2004.10934v1 [cs.CV].

Bradski, G. 2000. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*.

Brandman, Ko JP., S. 2011. Pulmonary nodule detection, characterization, and management with multidetector computed tomography. *J Thorac Imaging*.

Broder, Joshua MD. 2011. Diagnostic imaging for the emergency physician.

Ding, A.; Hu Z.; Wang L., J.; Li. 2017. Accurate pulmonary nodule detection in computed tomography images using deep convolutional neural networks. *In Medical Image Computing and Computer-Assisted Intervention—MICCAI*.

van Ginneken, Armato III S.G. et al., B. 2010. Comparing and combining algorithms for computer-aided detection of pulmonary nodules in computed tomography scans: the anode09 study. *Medical image analysis*, 14(6):707–722.

Ginneken, B.V. 2017. Fifty years of computer analysis in chest imaging: rule-based, machine learning, deep learning. *Radiol Phys Technol*, 10:23–32.

Hamidian, B.; Petrick N.; Pezeshk A., S.; Sahiner. 2017. 3d convolutional neural network for automatic detection of lung nodules in chest ct. *Proc. SPIE- Int. Soc. Opt. Eng.*, page 10134.

Howard Lee, Yi-Ping Phoebe Chen. 2015. Review image based computer aided diagnosis system for cancer detection. *Expert Systems with Applications*, 42(12):5356–5365.

INESCTEC. Lndb grand challenge.

Jinimole, C. G.and Harsha A. 2017. Comparative study of different enhancement techniques for computed tomography images. *International Journal of Biomedical and Biological Engineering*, 11(9).

Kathuria. What is new in yolo v3?

Kazerooni, Ella A. and Gross Barry H. 2004. *Cardiopulmonary Imaging*. Lippincott Williams Wilkins.

Krizhevsky, I.;, A.; Sutskever and G. E. Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, pages 1097–1105.

Messay, Hardie R. C. et al., T. 2010. A new computationally efficient cad system for pulmonary nodule detection in ct imagery. *Medical image analysis*, 14(3):390–406.

Nasrullah, N1.; Sang J1; Alam MS2; Mateen M1; Cai B1; Hu H1. 2019. Automated lung nodule detection and classification using deep learning combined with multiple strategies. *Europe PMC*, page PMID: 31466261.

Pedrosa, Aresta G. Ferreira C. et al., J. 2019. Lndb: A lung nodule database on computed tomography. *Grand Challenge on automatic lung cancer patient management*.

Pinsky, Paul F. 2015. Assessing the benefits and harms of low-dose computed tomography screening for lung cancer. *US PMC*, page PMID: 26617677.

Ramachandran, J.; Skaria S.; VarunV., S.; George. 2018. Using yolo based deep learning network for real time detection and localization of lung nodules from low dose ct scans. *Proceedings of the Medical Imaging 2018: Computer-Aided Diagnosis, International Society for Optics and Photonics, Houston, TX, USA*, 10575.

Redmon, Divvala S. Girshick R. Farhadi A., J. 2016. You only look once, unified real time object detection. Technical Report arXiv:1506.02640v5 [cs.CV].

Riquelme, D. and A.M. Akhloufi. 2020. Deep learning for lung cancer nodules detection and classification in ct scans. *AI*, 1:28–67.

Ross, G. 2015a. Fast r-cnn. *The IEEE International Conference on Computer Vision (ICCV)*.

Ross, Jeff D. Trevor D. Jitendra M., G. 2015b. Rich feature hierarchies for accurate object detection and semantic segmentation. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Shaoqing, Kaiming H. Ross G. Jian S., R. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in Neural Information Processing Systems*, pages 91–99.

Szegedy, C.; Liu W. Jia; Y. Sermanet P.; Reed S.; Anguelov D.; Erhan D.; Rabinovich A. 2015. Going deeper with convolutions. *IEEE Conf. on CVPR*, pages 1097–1105.

Tan, Lizhe. Jean Jiang. 2019. Image processing basics. *Digital Signal Processing (Third Edition)*.
Trajanovski, D.; Swisher C.L.; Gebre B.G.; Veeling B.; Wiemker R.; Klinder T.; Tahmasebi A.; Regis S.M.; Wald C.; et al., S.; Mavroeidis. 2018. Towards radiologist-level cancer risk assessment in ct lung screening using deep learning. *arXiv:1804.01901*.
Victor, Victor V. Mikla, I. Mikla. 2014. *Medical Imaging Technology*. Elsevier Inc.
Wei L, Dumitru E Christian S Scott R Cheng Yang Fu et al., Dragomir A. 2016. Ssd: Single shot multibox detector. *Advances in Neural European Conference on Computer Vision*, pages 21–37.
Zheng, S. et al. 2020. Efficient convolutional neural networks for multi-planar lung nodule detection: improvement on small nodule identification.

Appendix A: YOLO v3 architecture

	YOLO v3				
	Layer	***Filters***	***Size***	***Stride***	***Output shape***
	Convolutional	32	3 x 3	1	512 x 512 x 32
	Convolutional	64	3 x 3	2	256 x 256 x 64
	Convolutional	32	1 x 1	1	256 x 256 x 32
	Convolutional	64	3 x 3	1	256 x 256 x 64
	Residual				256 x 256 x 64
	Convolutional	128	3 x 3	2	128 x 128 x 128
2x	Convolutional	64	1 x 1	1	128 x 128 x 64
	Convolutional	128	3 x 3	1	128 x 128 x 128
	Residual				128 x 128 x 128
	Convolutional	256	3 x 3	2	64 x 64 x 256
8x	Convolutional	128	1 x 1	1	64 x 64 x 128
	Convolutional	256	3 x 3	1	64 x 64 x 256
	Residual				64 x 64 x 256
	Convolutional	512	3 x 3	2	32 x 32 x 512
8x	Convolutional	256	1 x 1	1	32 x 32 x 256
	Convolutional	512	3 x 3	1	32 x 32 x 512
	Residual				32 x 32 x 512
	Convolutional	1024	3 x 3	2	16 x 16 x1024
4x	Convolutional	512	1 x 1	1	16 x 16 x 512
	Convolutional	1024	3 x 3	1	16 x 16 x1024
	Residual				16 x 16 x1024
3x	Convolutional	512	1 x 1	1	16 x 16 x 512
	Convolutional	1024	3 x 3	1	16 x 16 x1024
	Convolutional	6	1 x 1	1	16 x 16 x 6
	Yolo				
	Concatenate				16 x 16 x 512
	Convolutional	256	1 x 1	1	16 x 16 x 256
	Upsample		2x		32 x 32 x 256
	Concatenate				32 x 32 x 768
3x	Convolutional	256	1 x 1	1	32 x 32 x 256
	Convolutional	512	3 x 3	1	32 x 32 x 512
	Convolutional	6	1 x 1	1	32 x 32 x 6
	Yolo				
	Concatenate				32 x 32 x 256
	Convolutional	128	1 x 1	1	32 x 32 x 128
	Upsample		4x		128 x 128 x 128
	Concatenate				128 x 128 x 256
3x	Convolutional	128	1 x 1	1	128 x 128 x 128
	Convolutional	256	3 x 3	1	128 x 128 x 256
	Convolutional	54	1 x 1	1	128 x 128 x 54
	Yolo				

Appendix B: YOLO v4 architecture

YOLO v4 - part 1/2

	Layer	Filters	Size	Stride	Output shape
	Convolutional	32	3x3	1	608 x 608 x 32
	Convolutional	64	3x3	2	304 x 304 x 64
	Convolutional	64	1x1	1	304 x 304 x 64
	Concatenate				304 x 304 x 64
	Convolutional	64	1x1	1	304 x 304 x 64
	Convolutional	32	1x1	1	304 x 304 x 32
	Convolutional	64	3x3	1	304 x 304 x 64
	Residual				304 x 304 x 64
	Convolutional	64	1x1	1	304 x 304 x 64
	Concatenate				304 x 304 x 128
	Convolutional	64	1x1	1	304 x 304 x 64
	Convolutional	128	3x3	2	152 x 152 x 128
	Convolutional	64	1x1	1	152 x 152 x 64
	Concatenate				152 x 152 x 128
	Convolutional	64	1x1	1	152 x 152 x 64
	Convolutional	64	1x1	1	152 x 152 x 64
	Convolutional	64	3x3	1	152 x 152 x 64
	Residual				152 x 152 x 64
	Convolutional	64	1x1	1	152 x 152 x 64
	Convolutional	64	3x3	1	152 x 152 x 64
	Residual				152 x 152 x 64
	Convolutional	64	1x1	1	152 x 152 x 64
	Concatenate				152 x 152 x 128
	Convolutional	128	1x1	1	152 x 152 x 128
	Convolutional	256	3x3	2	76 x 76 x 256
	Convolutional	128	1x1	1	76 x 76 x 128
	Concatenate				76 x 76 x 256
	Convolutional	128	1x1	1	76 x 76 x 128
8x	Convolutional	128	1x1	1	76 x 76 x 128
8x	Convolutional	128	3x3	1	76 x 76 x 128
8x	Residual				76 x 76 x 128
	Convolutional	128	1x1	1	76 x 76 x 128
	Concatenate				76 x 76 x 256
	Convolutional	256	1x1	1	76 x 76 x 256
	Convolutional	512	3x3	2	38 x 38 x 512
	Convolutional	256	1x1	1	38 x 38 x 256
	Concatenate				38 x 38 x 512
	Convolutional	256	1x1	1	38 x 38 x 256
8x	Convolutional	256	1x1	1	38 x 38 x 256
8x	Convolutional	256	3x3	1	38 x 38 x 256
8x	Residual				38 x 38 x 256
	Convolutional	256	1x1	1	38 x 38 x 256
	Concatenate				38 x 38 x 512
	Convolutional	512	1x1	1	38 x 38 x 512
	Convolutional	1024	3x3	2	19 x 19 x 1024
	Convolutional	512	1x1	1	19 x 19 x 512
	Concatenate				19 x 19 x 1024
	Convolutional	512	1x1	1	19 x 19 x 512

YOLO v4 - part 2/2

	Layer	Filters	Size	Stride	Output shape
	Convolutional	512	1x1	1	19 x 19 x 512
4x	Convolutional	512	3x3	1	19 x 19 x 512
4x	Residual				19 x 19 x 512
	Convolutional	512	1x1	1	19 x 19 x 512
	Concatenate				19 x 19 x 1024
	Convolutional	1024	1x1	1	19 x 19 x 1024
	Convolutional	512	1x1	1	19 x 19 x 512
	Convolutional	1024	3x3	1	19 x 19 x 1024
	Convolutional	512	1x1	1	19 x 19 x 512
	Max		5 x 5	1	19 x 19 x 512
	Concatenate				19 x 19 x 512
	Max		9 x 9	1	19 x 19 x 512
	Concatenate				9 x 19 x 512
	Max		13 x 13	1	19 x 19 x 512
	Concatenate				19 x 19 x 2048
	Convolutional	512	1x1	1	19 x 19 x 512
	Convolutional	1024	3x3	1	19 x 19 x 1024
	Convolutional	512	1x1	1	19 x 19 x 512
	Convolutional	256	1x1	1	19 x 19 x 256
	Upsample		2x		38 x 38 x 256
	Concatenate				38 x 38 x 512
	Convolutional	256	1x1	1	38 x 38 x 256
	Concatenate				38 x 38 x 512
2x	Convolutional	256	1x1	1	38 x 38 x 256
2x	Convolutional	512	3x3	1	38 x 38 x 512
	Convolutional	256	1x1	1	38 x 38 x 256
	Convolutional	128	1x1	1	38 x 38 x 128
	Upsample		4x		152 x 152 x 128
	Concatenate				152 x 152 x 128
	Convolutional	128	1x1	1	152 x 152 x 128
	Concatenate				152 x 152 x 256
3x	Convolutional	128	1x1	1	152 x 152 x 128
3x	Convolutional	256	3x3	1	152 x 152 x 256
	Convolutional	6	1x1	1	152 x 152 x 6
	Yolo				
	Concatenate				152 x 152 x 128
	Convolutional	256	3x3	4	38 x 38 x 256
	Concatenate				38 x 38 x 512
3x	Convolutional	256	1x1	1	38 x 38 x 256
3x	Convolutional	512	3x3	1	38 x 38 x 512
	Convolutional	6	1x1	1	38 x 38 x 6
	Yolo				
	Concatenate				38 x 38 x 256
	Convolutional	512	3x3	2	19 x 19 x 512
	Concatenate				19 x 19 x 1024
3x	Convolutional	512	1x1	1	19 x 19 x 512
3x	Convolutional	1024	3x3	1	19 x 19 x 1024
	Convolutional	54	1x1	1	19 x 19 x 54
	Yolo				